Project Number: **770299**

NewsEye:

A Digital Investigator for Historical Newspapers

Research and Innovation Action
Call H2020-SC-CULT-COOP-2016-2017

D4.5: Analysis of data in a given context (c) (final)

Due date of deliverable: M45 (31 January 2022)

Actual submission date: 10 December 2021

Start date of project: 1 May 2018 **Duration:** 45 months

Partner organization name in charge of deliverable: UH-CS

Project co-funded by the European Commission within Horizon 2020		
Dissemination Level		
PU	Public	PU
PP	Restricted to other programme participants (including the Commission Services)	-
RE	Restricted to a group specified by the Consortium (including the Commission Services)	-
CO	Confidential, only for members of the Consortium (including the Commission Services)	-

Revision History

Document administrative information		
Project acronym:	NewsEye	
Project number:	770299	
Deliverable number:	D4.5	
Deliverable full title:	Analysis of data in a given context (c) (final)	
Deliverable short title:	Analysis of data in context (final)	
Document identifier:	NewsEye-T41-D45-AnalysisOfContentInContext-c-Submitted-v3.1	
Lead partner short name:	UH-CS	
Report version:	V3.1	
Report preparation date:	10.12.2021	
Dissemination level:	PU	
Nature:	Report	
Lead author:	Mark Granroth-Wilding (UH-CS)	
Co-authors:	Elaine Zosa (UH-CS), Lidia Pivovarova (UH-CS)	
Internal reviewers:	Axel Jean-Caurant (ULR), Emanuela Boros (ULR)	
Status:		Draft
		Final
	x	Submitted

The NewsEye Consortium partner responsible for this deliverable has addressed all comments received, making changes as necessary. Changes to this document are detailed in the change log table below.

Change Log

Date	Version	Editor	Summary of changes made
5/3/2020	0.1	Mark Granroth-Wilding (UH-CS)	First draft
15/3/2021	0.2	Elaine Zosa (UH-CS)	Updated sections
19/3/2021	1.0	Mark Granroth-Wilding (UH-CS)	Further updates towards final draft
10/4/2021	1.1	Elaine Zosa (UH-CS)	Updates based on internal reviews
20/4/2021	2.0	Mark Granroth-Wilding (UH-CS)	Prepared final version for quality management (QM)
29/4/2021	2.1	Mark Granroth-Wilding (UH-CS)	Further updates following QM
30/04/2021	3.0	A. Doucet (ULR)	Minor changes and finalisation
10/12/2021	3.1	A. Doucet (ULR)	Submission

Executive summary

The NewsEye project addresses challenges relating to the exploration of historical news corpora. It makes contributions in text recognition, text analysis, natural language processing (NLP) and generation (NLG); in digital newspaper research; in digital humanities; and in history, in terms of analyzing historical assets with new methods.

WP4, entitled 'Dynamic Text Analysis' aims to develop and implement methods for *contextualized* and *contrastive content analysis*, carried out *dynamically*, both for use directly by a *Demonstrator* component, which allows the user to access the tools and collections, and by an *Investigator* component, which performs autonomous analysis and presents its results to the user. In this task, we are developing methods and tools for performing this analysis, primarily using *topic models* (TMs).

We report on a collection of tools that have been developed for use in content analysis and their integration into the NewsEye pipeline, taking input from Work Package 2. The tools are being used in the NewsEye Demonstrator and the Investigator. These provide **multilingual topic models** and **dynamic topic models** of various sorts, as well as subsequent analyses based on these models. The goals of these methods are to analyse textual content to support interactive analysis and to make tools available both for the end users and for the automated **Personal Research Assistant**.

A key challenge is to analyse a **multilingual** collection. We address this by investigating *multilingual topic modelling*. This has involved implementing existing work and developing new methods that extend existing models. This has included experiments on the application of recent topic modelling techniques that exploit *word embeddings*. The ultimate goal of this line of investigation – to produce a new type of *multilingual* topic model by combination with *multilingual word embeddings* – is still at the time of writing under development. We plan to report on further results in this area in an updated version of this deliverable.

To account for the large time span of the NewsEye data, we have investigated existing work on *dynamic TMs*. These models have not previously been adapted for *multilingual* corpora. We have developed a **novel multilingual, dynamic** TM and present the experimental results.

Significant work has been devoted also to producing a full working pipeline, analyses, and tools specifically focused on subsets of the NewsEye collection. We report on the completion of this pipeline. The tools provided by WP4 have been made available to both the Demonstrator and the Investigator via an API, which we describe in this deliverable.

We also report on work performed together with digital humanities collaborators to gain insight into their foreseen uses of the tools and how they may make productive contributions to historical research.

Contents

1. Introduction

In this section, we set the work of WP4 in the broader context of NewsEye, describe the goals of Task 4.1 and summarise the work carried out in this task during the second year.

This document is an updated version of *D4.1: Analysis of Content in a Given Context (a)* and *D4.2: Analysis of Content in a Given Context (b)*, which described the work on this task in the first two years of the project. However, this document is intended to be read in isolation, so it provides all necessary background.

1.1. Context within NewsEye

The NewsEye project addresses a number of challenges relating to the exploration of historical news corpora. These involve contributions in several directions:

- in text recognition, text analysis, natural language processing, computational creativity, and natural language generation, particularly with regard to historical newspapers;
- in digital newspaper research, addressing a number of editorial issues like OCR and article separation (AS);
- in digital humanities, dealing with huge amounts of text material, availability of useful tools and possibilities of searching and browsing; and
- in history, in terms of analyzing historical assets with new methods across different language corpora.

Central to the project are the *Demonstrator*, a means for a user to explore large collections, and the *Personal Research Assistant* (PRA), a tool to perform an autonomous exploratory search of collections to help a user identify some contents of interest. The PRA consists of the *Investigator*, carrying out an autonomous analysis, the *Reporter*, delivering reports on the results to the user, and the *Explainer*, explaining how the results were arrived at and why they may be of interest. The interactions between these components are described by Figure 1.

At the heart of both the Demonstrator and the Investigator component of the PRA lies a collection of tools for analysing historical newspaper data, made available in textual form by WP2 and enhanced with semantic annotations by WP3. WP4 provides a set of tools for broad-scale analysis of the collection and analysis of smaller groups of articles in the context of the whole collection. These tools can be used both by the user directly (through the Demonstrator) and by the autonomous Investigator. In years 2 and 3, we have focused on integration between the tools of WP4 and the Demonstrator, ensuring that as much as possible of the work carried out on basic methods in T4.1 can be exploited by the user in the Demonstrator. We also have a rich level of integration with the PRA (WP5), allowing it to explore analyses that build on the methods developed here.

1.2. Work package 4: Dynamic text analysis

The main objective of WP4 is to develop and implement methods for contextualized and contrastive content analysis, carried out dynamically. In this task, we are developed the methods and tools for performing this analysis, primarily using *topic models* (TMs).

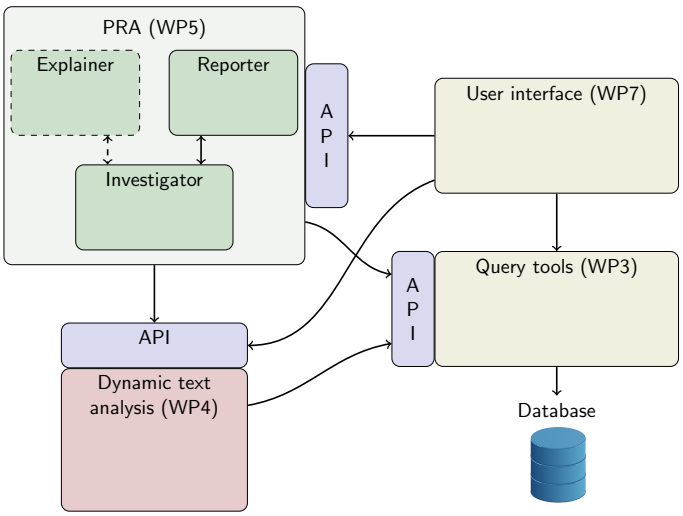


Figure 1: High-level architecture of component systems of NewsEye, showing how WP4 interacts with other WPs to acquire data and provide analyses.

The goals of these methods are:

- to provide detailed analysis of textual content given a context;
- to support interactive analysis of the content by discovering patterns, topics, trends, and viewpoints in given contexts;
- to make tools available both for the end-users and for the automated Personal Research Assistant, in both cases via an API.

Contexts for queries can concern a specific time range, topic, event, named entity and other aspects supported by the enriched data and static analyses and indices built in WP2 and WP3, as well as corpus-level textual analysis from the tools of this WP themselves. Since all of these types of contexts can, by means of the metadata query tools provided in other WPs, be reduced to a set of documents corresponding to the context, we define a context as either:

1. a set of query parameters defined by the user using the NewsEye demonstrator; or
2. a list of document IDs that are the result of the query.

One of the key challenges of this WP is to perform corpus-level analysis of a **multilingual** collection. We have addressed this by investigating a variety of types of *multilingual topic modelling*. This work has involved implementing the existing model of [1], as well as extending the model to make it *dynamic* (see below). Additionally, we have experimented with novel developments and applications of several existing methods for combining Bayesian topic modelling with *word embeddings* as an alternative way to make the topic models multilingual. So far, this work has explored monolingual uses of the modelling methods, with monolingual embeddings, but we are in the process of experimenting with their combination with *multilingual embeddings* and hope to report on this in an updated version of this deliverable.

Since the newspaper collection spans a large time period, it is advantageous, or even essential for some purposes, to account for changes in both the languages themselves (e.g., changing spelling conventions or word meanings) and the way the languages are used to discuss particular topics (e.g., changes in specialist vocabulary). This issue is addressed by existing work on *dynamic topic models*, but these

models have not previously been adapted to account for multilingual corpora. We have developed with a type of **multilingual, dynamic** TM. This work has been published in a major natural language processing (NLP) conference [2]. In Section 2.3, we present the model details and experimental evaluations.

We provide a detailed report here on the various avenues of topic modelling work that we have pursued during the project: multilingual TMs (Section 2.1); dynamic TMs (Section 2.2), including the novel multilingual adaptation and extensive experiments carried out on it (Section 2.3); experiments with TMs that use word embeddings, *Gaussian LDA*, *Latent Concept LDA* and the *Embedded Topic Model* (Section 2.4).

In Section 3, we give an overview of the processes established for training these TMs on the data from the collection, including preprocessing of text, which have allowed us to easily re-train TMs as new data became available from WP2 and WP3. Particular work went into establishing a full working pipeline, analyses, and tools, specifically focused on the NewsEye collection.

A further goal of T4.1 is to provide tools to discover links between documents within the NewsEye collection, including across language boundaries. In Section 4 we discuss work we have done on cross-lingual document linking.

In Section 5 we describe work on methods to address the issues arising from noisy input received from earlier tools in the pipeline due to errors in automatic text recognition (ATR). In particular, we have carried out experiments on the robustness of a variety of methods to such noise.

The tools provided by WP4 have been made available to both the Demonstrator and the Investigator via an API. Section 6 details the API available for the models described here. This is in active use by the Demonstrator, as well as being incorporated into the Investigator's armoury of analysis techniques using the same API.

In Section 7, we describe the ways in which we have made the source code for the tools developed in this WP available for further work outside and beyond the lifetime of the project.

Section 8 outlines our ongoing work together with digital humanities collaborators to gain insight into their foreseen uses of the tools and in what ways they can make productive contributions to historical research. Finally, in Section 9 we describe open questions in these lines of research.

Appendix E gives the full configuration file for training topic models for the working group work described in Section 3.

1.3. Key Performance Indicator

The NewsEye project description describes the following Key Performance Indicator (KPI):

Improvement of data analysis and exploration by dynamic text analysis	Demonstrate improvement in user satisfaction in dynamic text analysis capabilities of NewsEye. Success measured in terms of number of queries made using the dynamic features. **KPI Goal**: more than 75% of queries made by users use dynamic text analysis features.

Call type	Logged calls
Time series prominent splits	16
Word embeddings	39
Document linking with topic models	25
Topic model analyses	213
Document set comparisons with topic models	837
Other calls (not dynamic text analysis)	7604

Table 1: Number of calls to the different tools and analysis methods made by the autonomous Investigator.

The specific goal of 75% of queries was set when a quite different mode of interaction for the user of the Demonstrator was envisaged. The system was developed in conjunction with users and the needs and expectations of digital humanities scholars and other user groups were better understood, and in the light of the result, this goal made no sense. Every user begins by making a number of keyword searches at least before they can consider using, for example, the topic modelling tools described in this deliverable. Similarly, the Investigator always starts its investigation with queries on simple facets of the articles before applying more advanced tools. Here we report a number of statistics that demonstrate a similar level of active in a manner more appropriate to the final tools.

Every experiment carried out by the Investigator uses the dynamic text analysis tools. The developed methods and tools have been found to be sufficiently effective and useful that they are always used in at least one part of the pipelines of processing put together by the Investigator.

An examination of the logs of tools used by the Investigator shows that **13%** of calls to the available tools are to one of the dynamic text analysis methods. Considering that a call is made for every operation to, for example, extract the distribution of words used in a sub-collection, this is a surprisingly high figure. Table 1 shows the distribution of calls within this set.

Aside from their use in the Investigator, these tools are used elsewhere in the Demonstrator as well. The main keyword search interface makes calls to the word embeddings provided as part of the dynamic text analysis methods to assist with editing queries. **The tools are therefore used as a part of every user query in the Demonstrator.**

Through the Demonstrator, the user is also able to build a pipeline of tasks processing a subcollection, in the same way that the Investigator does autonomously. It is therefore of interest to look at how often users choose to make use the tools provided by WP4. Logs tell us that **topic models are included manually in 20% of pipelines built by users**. This figure initially appears low compared to the 75% target of the KPI. However, taking into account the substantially different approach to interaction with the tools taken compared with the envisaged approach that motivated the KPI, we consider this to be a **surprisingly high figure** and to reflect a success in application of these dynamic analysis tools.

2. Modelling methods

In this section, we introduce the various statistical models that we have adopted or developed for performing dynamic analysis in WP4.

In Year 1, we presented a method for learning hierarchies of topics, *HLDA*. After further work on the integration of TMs into the NewsEye workflow and discussions and feedback from DH collaborators, we have chosen not to work further on this line of topic modelling. Instead, we have prioritized making TMs that produce flat sets of topics which are more useful for analysis of historical data, as well as new techniques for making comparisons or contrasts on the basis of their analyses.

In Year 2, we developed the API interface of WP4 to enable other work packages to access topic model results and functionalities such as document linking. We also developed the WP4 document processing and training pipeline for processing the NewsEye collection and traininng our models.

In Year 3, we focused on the application of *multilingual* TMs for linking documents from different languages and using *dynamic* TMs for exploring the dynamics of discourses in historical newspapers. We also investigated the robustness of embedding-based topic models to handle noisy text data.

In this section, we present the various topic modelling methods that we have investigated and developed in this task.

2.1. Multilingual Topic Models

Most previous work on topic modelling has made the assumption that all documents in the collection being analysed are in a single language. For NewsEye's purposes, we need to provide analyses of a multilingual document collection, drawing connections between articles with common topics across different languages. The NewsEye collection will include four languages. Our aim is to develop methods that can be applied to model the topics discussed in a collection containing all four languages.

In Year 1, we investigated the limited existing work on multilingual topic modelling, including polylingual topic modeling (PLTM, [1]) and bilingual topic modelling using common word matching [3]. We chose PLTM as the more appropriate method for our purposes and compared it to an alternative involving aligning monolingual TMs. As the latter line of work produced inferior results, we have not developed it further, instead, we worked on an extension of PLTM (see Section 2.3, [2]). We then explored the application of Gaussian LDA [4] to multilingual data using cross-lingual word embeddings (see Section 2.4).

In Year 2, we explored the application of PLTMs to cross-lingual document retrieval. We found that combining PLTMs with cross-lingual word embeddings performs better than word embeddings or topic models alone. We have also started working on using PLTM on historical data to compare discourses across national boundaries. A significant issue we have encountered in this line of work is the need for PLTM to have aligned training data, which is another reason for us to investigate Gaussian LDA since it could allow us to avoid this issue.

2.1.1. Related work

Multilingual TMs are developed to capture cross-lingual topics from multilingual datasets. The topics they learn are *cross-lingual* in the sense that a single set of topics is used to describe the entire multilingual dataset, which each topic being related to (topically related) documents in multiple of the corpus' languages. Some existing models include Polylingual Topic Model [1], Multilingual Topic Model for Unaligned Text [3] and JointLDA [5].

Polylingual topic model [1][1] is an extension of LDA that infers topics from an aligned multilingual corpus composed of document tuples. Tuples are composed of documents in one or more languages that are thematically aligned. Another multilingual TM, suitable for unaligned corpora, is the Multilingual Topic Model for Unaligned Text [3]. MuTo attempts to match words from one language to the other language in the corpus and samples topic assignments for these matchings. JointLDA [5] is another multilingual model that does not require an aligned corpus but requires a bilingual dictionary. This model uses LDA to infer topics based on concepts rather than words, where concepts can be entries in the bilingual dictionary.

We chose to focus on PLTM because it can be easily extended to any number of languages and the simplicity of the model makes it suitable for combination with other types of TMs. In Year 1 we showed how it is possible to derive a weak alignment between documents from the type of metadata available for NewsEye articles.

2.1.2. Polylingual Topic Model (PLTM)

PLTM extends LDA to learn from an aligned aligned corpus. Instead of running topic inference on one document after another as in LDA, PLTM infers topics from tuples of documents, where each document in the tuple are made up of words drawn from different vocabularies (as they would be if they are of different languages). PLTM assumes that every document in a tuple discusses the same subject broadly and therefore shares the same document-topic distribution. It does not, therefore, rely on a *parallel* corpus, where documents are paired with direct translations, of the sort often used for machine translation, but assumes a weaker type of alignment. For example, pairs of documents published close in time and sharing prominent keywords can be used, on the assumption that they are likely to be discussing the same or a closely related event or issue.

We applied PLTM to the task of *ad hoc* cross-lingual document retrieval (CLDR). In this task, we are given a query document in one language and the goal is to rank the set of candidate documents in another language according to how related they are to the query document. This is in contrast to *known item* retrieval where only one of the candidate documents is relevant to the query document [6]. Previous work on cross-lingual known item retrieval such as matching Wikipedia pages used topic models, cross-lingual embeddings and cross-lingual distance measures [7, 8, 9, 10]. In these works, cross-lingual embeddings perform better than topic models however we find in our experiments that in fact, these methods are complementary to each other and that ensembling them gives a better performance than each method on their own.

Our manuscript on this work is attached to this deliverable (see Appendix **??**). We have also made improvements to the implementation of PLTM resulting in faster inference times by vectorising operations when possible.

2.2. Dynamic Topic Model (DTM)

Static TMs such as LDA and multilingual TMs learn static topics, meaning that each topic has a single distribution over the vocabulary. In the case of datasets that cover time spans, such as news articles,

[1] The authors used the term *polylingual*, but in our extension of the model (ML-DTM, below) we use *multilingual* for consistency with other work, such as [3]. There is no meaning distinction between the two terms.

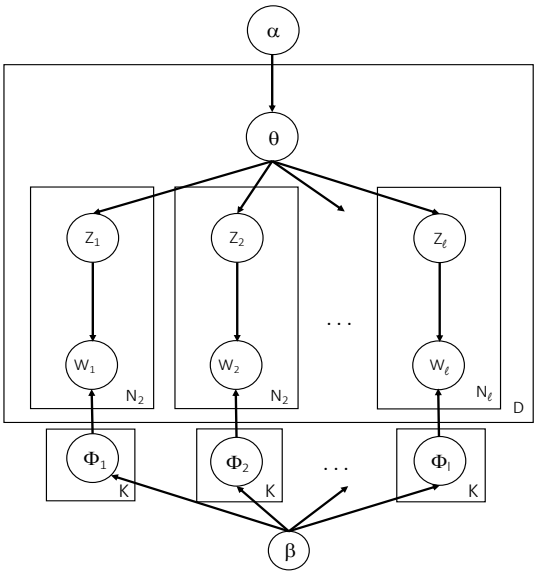


Figure 2: PLTM for l languages of Mimno et al. [1]. α and β are hyperparameters controlling Bayesian priors. θ is a document-specific topic distribution over the K topics. ϕ_l contains the distributions over the vocabulary of language l defining each topic.

we also want to capture dynamic co-occurrence patterns that evolve through time. Dynamic topic models [11] capture themes or topics discussed in a set of time-stamped documents and how the words related to these topics change in prominence with time. Several other models have been developed for capturing the same time-related phenomena [12, 13, 14]. We chose to focus on DTM because we want to capture topic evolution and language change.

DTM [11] explicitly models how words related to the topics change in prominence over time. It divides time into discrete slices and chains parameters from each slice in order to infer topics that are aligned across time – it assumes that a given topic in time slice t is closely related to the same topic at time $t+1$. The model employs a Markovian assumption where the state of the model at time $t+1$ is dependent only on the state at time t. Each topic will then have T different distributions, where T is the number of time slices. Figure 3 shows the plate diagram of this model.

DTM is designed for monolingual data. As such, it is not directly suitable for use in NewsEye, since it would only be able to provide analyses of documents in one language in the collection at a time. We use it here as the basis for a multilingual extension in the next section and for comparison to the extended model.

Implementations already exist of this model, and we use the Gensim implementation for our experiments. [2]

2.3. Multilingual Dynamic Topic Model (ML-DTM)

We present a novel TM that combines DTM and PLTM, the *multilingual dynamic topic model* (ML-DTM). ML-DTM is a novel topic model that captures dynamic topics from broadly topically aligned multilingual

[2] https://radimrehurek.com/gensim/models/ldaseqmodel.html

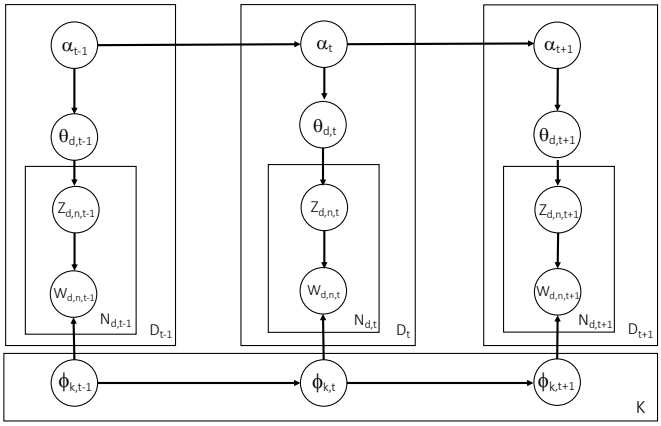


Figure 3: Dynamic topic model of Blei & Lafferty [11]

datasets, making the same assumption regarding the alignment of the input data as DTM (see above). We extend the DTM inference method of [15] to train this model.

In this section, we first give a description of our combined ML-DTM model. In the remainder of the section, we then demonstrate usage of this model on a parallel dataset and a comparable dataset of news articles and present our results. We show that this novel topic model learns aligned bilingual topics as demonstrated by the cosine similarities of learned vector representations of named entities. We have released code from training the model at https://github.com/NewsEye/Multilingual-Dynamic-Topic-Modeling. Further details can be found in [2].

2.3.1. Model

Figure 4 shows the diagram of ML-DTM for two languages and three time slices. Although we show only the bilingual case here for brevity, the model is applicable to any number of languages.

The inference method of [15] was originally motivated by the need to speed up DTM inference for very large datasets. We apply it here to the combined ML-DTM model. We propose the following posterior conditional distribution for $\theta_{x,t}$ where x is a tuple index in the dataset:

$$p(\theta_{x,t}|\alpha_t, Z_{x,t}) \propto \mathcal{N}(\theta_{x,t}|\alpha_t, \psi^2 I) \times \prod_{l=1}^{L} \prod_{n=1}^{N_{d_l,t}} Mult(Z_{d_l,n,t}|\pi(\theta_{x,t}))$$

Following [15], the update equation to evaluate the gradient of $\theta^k_{x,t}$ becomes:

$$\nabla_{\theta^k_{x,t}} \log p(\theta_{x,t}|\alpha_t, Z_{x,t}) = \frac{-1}{\psi^2}(\theta^k_{x,t} - \alpha^k_t) + \sum_{l=1}^{L} C^k_{d_l,t} - \left(N_{d_l,t} \times \frac{\exp(\theta^k_{x,t})}{\sum_j \exp(\theta^j_{x,t})} \right) \quad (1)$$

where $Z_{x,t}$ are the topic assignments for the words in the documents in tuple x at time slice t; $C^k_{d_l,t}$ is the number of times topic k has been assigned to a word in document d_l at time t; and $N_{d_l,t}$ is the length of document d_l at time t.

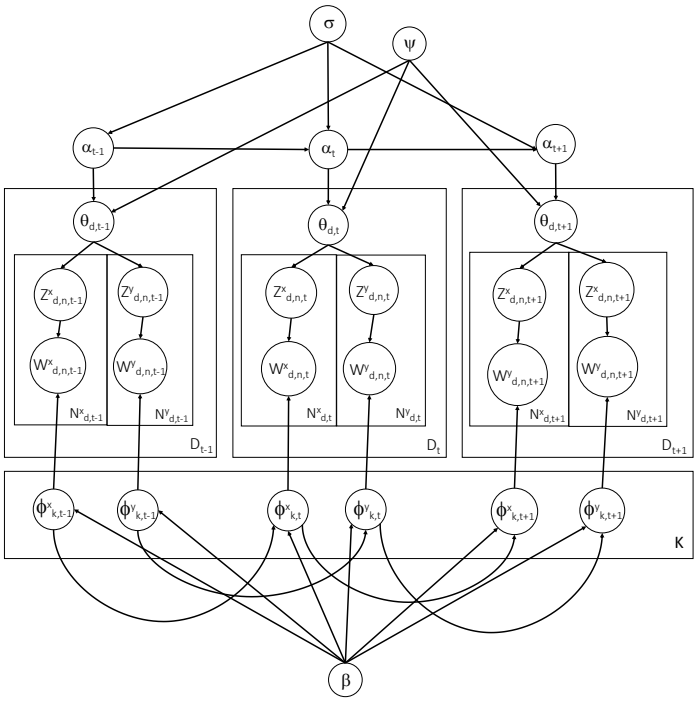


Figure 4: ML-DTM for two languages and three time slices.

Table 2: Dimensions of the sampled parameters in the multilingual dynamic topic model (ML-DTM). D^t is the number of document tuples in a dataset.

Parameter	Dimension
α	$K \times T$
θ	$D^t \times K \times T$
ϕ	$\lvert V^l \rvert \times L \times K \times T$

Instead of evaluating $\theta_{d,t}$ for a single document as in monolingual DTM, we compute $\theta_{x,t}$ for a document *tuple*. The second term in Eq 1 links the languages together by summing up the counts of each document in the tuple.

The equation for evaluating the gradient of the topic-term distributions $\phi_{k,t}$ is the same as in the original paper except that we compute separate distributions for each language since every language has a different vocabulary. This means that for each time slice, instead of updating K different ϕs (one for each topic), we will need to update $K \cdot L$ ϕs. Table 2 shows the dimensions of the parameters to be estimated.

Finally, the topic assignment $Z_{d_l,n,t}$ is sampled as in the original paper:

$$P(Z_{d_l,n,t} = k|\theta_{x,t}, \phi_{k,t}^{w_l}) \propto exp(\theta_{x,t}^k) exp(\phi_{k,t}^{w_l})$$

where w_l is a word from the vocabulary of language l.

Table 3: Average cosine similarity of topic vectors for NEs over three time slices in DE-NEWS.

Time slice	# of NEs	PLTM	ML-DTM
Aug 1996	53	**0.880**	0.692
Sept 1996	65	0.876	**0.908**
Oct 1996	64	0.840	**0.885**

Table 4: Average cosine similarity of the vectors of NEs for three time slices in the YLE dataset.

Time slice	# of NEs	PLTM	ML-DTM
Jan 2012	79	0.800	**0.896**
Feb 2012	71	**0.810**	0.796
Mar 2012	72	0.722	**0.745**

2.3.2. Datasets

We ran experiments on ML-DTM with two kinds of data: a parallel dataset and a thematically-comparable one.

The DE-NEWS parallel dataset consists of German news articles from August 1996 to January 2000 with English translations done by human volunteers[3]. This dataset covers 42 months with an average of 200 articles per month. Since this is a parallel corpus there is no need to align the articles.

For the comparable dataset, we use the YLE news dataset which consists of Finnish and Swedish articles from the Finnish broadcaster YLE, covering news in Finland from January 2012 to December 2018[4]. The Finnish and Swedish articles are written separately and are not direct translations of each other. We use existing methods for aligning comparable news articles [16, 17]. Specifically, we create an aligned corpus by pairing a Finnish article with a Swedish article published within a two-day window and sharing three or more named entities. We want to have a one-to-one alignment in our dataset such that no article is duplicated, so we pair a Finnish article with the first Swedish article encountered in the dataset that fits the above criteria, and then remove the paired articles from the unaligned dataset. The unaligned dataset has a total of 604,297 Finnish articles and 228,473 Swedish articles and the final aligned dataset consists of 123,818 articles covering 84 months. A script for aligning articles using the method described is provided in the Github project associated with this work: https://github.com/NewsEye/Multilingual-Dynamic-Topic-Modeling.

We tokenized, lemmatized (using WordNetLemmatizer for German and English and LAS [18] for Finnish and Swedish), we removed stopwords for these two datasets and then used the 5,000 most frequent words of each language as the vocabulary for that language.

[3] http://homepages.inf.ed.ac.uk/pkoehn/publications/de-news/
[4] https://www.kielipankki.fi/corpora/

Table 5: Topic diversity comparison between DTM and ML-DTM: average JS divergences of each topic pair for five months of the DE-NEWS dataset for English and German.

Time slice	DTM English	ML-DTM English
Aug 1996	0.372	**0.655**
Sep 1996	0.368	**0.660**
Oct 1996	0.366	**0.657**
Nov 1996	0.365	**0.664**
Dec 1996	0.363	**0.650**
	DTM German	ML-DTM German
Aug 1996	0.315	**0.661**
Sep 1996	0.312	**0.670**
Oct 1996	0.310	**0.665**
Nov 1996	0.308	**0.638**
Dec 1996	0.306	**0.666**

2.3.3. Cross-lingual alignment

We compare the cross-lingual alignment of topics of ML-DTM and PLTM by evaluating the similarity of the learned vector representations of named entities (NEs) that appear in both languages of the same dataset. This method is suggested by [19] on the basis that NEs tend to be spelled in the same way in different languages and can be expected to have a similar association with topics across languages. The K-dimensional vector of a NE w for language s is thus:

$$vec(w_s) = [P(z_1|w_s), P(z_2|w_s), ..., P(z_K|w_s)] \tag{2}$$

Under an assumption of a uniform prior over topics, this vector can be computed as:

$$\begin{aligned} P(z_k|w_s) &\propto \frac{P(w_s|z_k)}{P(w_s)} \\ &= \frac{\phi_{l,z_k,w_s}}{Norm_{\phi_{s,.,w_s}}} \end{aligned} \tag{3}$$

$$Norm_{\phi_{s,.,w_s}} = \sum_{k=1}^{K} \phi_{s,z_k,w_s} \tag{4}$$

$$vec(w_s) = \frac{[\phi_{l,z_1,w_s}, \phi_{l,z_2,w_s}, ..., \phi_{l,z_K,w_s}]}{Norm_{\phi_{s,.,w_s}}} \tag{5}$$

We then take the cosine similarities between the L different vector representations of the NE (for both datasets, $L = 2$).

We evaluate the cosine similarities of NEs that occur five or more times in each time slice. To make the comparison between PLTM and ML-DTM, we train one ML-DTM model on three time slices for 10 topics and three separate PLTM models for each time slice, also capturing 10 topics. We set $\alpha = 1.0$ and $\beta = 0.08$ for PLTM and $\alpha = 0.5$ and $\beta = 0.5$ for ML-DTM for both datasets, which achieved the best results of a small range of values tried.

Table 6: Topic diversity comparison between DTM and ML-DTM: average JS divergences of each topic pair for five months of the YLE dataset for Finnish and Swedish.

Time slice	DTM Finnish	ML-DTM Finnish
Jan 2012	0.332	**0.445**
Feb 2012	0.324	**0.465**
Mar 2012	0.322	**0.470**
Apr 2012	0.353	**0.498**
May 2012	0.357	**0.495**
	DTM Swedish	ML-DTM Swedish
Jan 2012	0.365	**0.480**
Feb 2012	0.360	**0.491**
Mar 2012	0.354	**0.497**
Apr 2012	0.388	**0.535**
May 2012	0.393	**0.537**

2.3.4. Topic diversity

We also measure the *diversity* of the topics ML-DTM finds by computing the Jensen-Shannon (JS) divergence of every topic pair for each time slice for each language and averaging the divergences. [12] used this method, though with KL divergence. It is desirable for the model to find topics that are as distinct as possible from each other.

We compare the diversity of the topics found by ML-DTM, trained as in the previous section, with the topics found by DTM. To make this comparison we train separate DTM models for each language in our two datasets, giving us four different models and compare the divergences of the topics found by these models with their ML-DTM counterparts. We use the Gensim implementation of DTM[5] where we set the chain variance to 0.1 and leave other parameters to be inferred during training. We train both ML-DTM and DTM on 10 time slices for 10 topics.

2.3.5. Results and discussion

Tables 3 and 4 show the average cosine similarity between NEs for each language in the DE-NEWS and YLE datasets, respectively. In the DE-NEWS data (Table 3), PLTM outperforms ML-DTM in the first time slice but ML-DTM performs better on the succeeding time slices. This is an encouraging result, considering that the parameters of ML-DTM at time slice t are estimated from adjacent time slices, adding a large degree of complexity to the model, whereas PLTM estimates parameters based on the current time slice only (PLTM has no concept of time).

For the YLE dataset (Table 4), ML-DTM shows an improvement in the first and third time slices and comparable performance in the second. The comparable nature of this dataset makes aligning NEs a more challenging task for both models. One way to improve performance on this task might be to use

[5] https://radimrehurek.com/gensim/models/ldaseqmodel.html

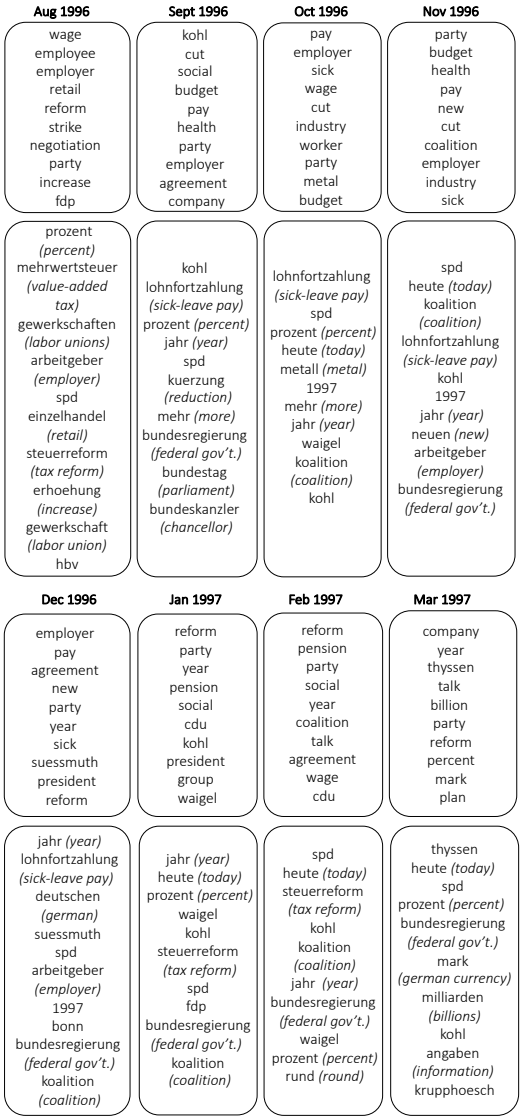


Figure 5: Top words of a topic concerning news about labor unions from the DE-NEWS dataset for English (top) and German (bottom) from Aug 1996 to March 1997. English translations of the German words excluding named entities are enclosed in parentheses.

stricter criteria in aligning the dataset, such as pairing articles only if they were published on the same day or if they share more named entities.

We compare topic diversity of the topics found by DTM and ML-DTM. Tables 5 and 6 show the average JS divergence of every topic pair for five time slices in the DE-NEWS and YLE datasets, respectively. ML-DTM consistently learns more diverse topics than DTM for both datasets.

In Figure 5, we show the evolution of one topic found by ML-DTM trained on DE-NEWS. We show the top words of a topic about labour unions for the first eight months of the dataset. The English and German words are not exact translations of each other, but we see similar or related words and NEs in each time slice. For instance, in August 1996 'employer' and 'arbeitgeber' both appear, as does 'einzelhandel' and 'retail'. In Sept 1996, 'kohl' is the top term for both languages (referring to former German chancellor Helmut Kohl). There are cases where German terms have no direct translation in English but an equivalent concept appears in the English topic. This is the case with 'lohnfortzahlung'

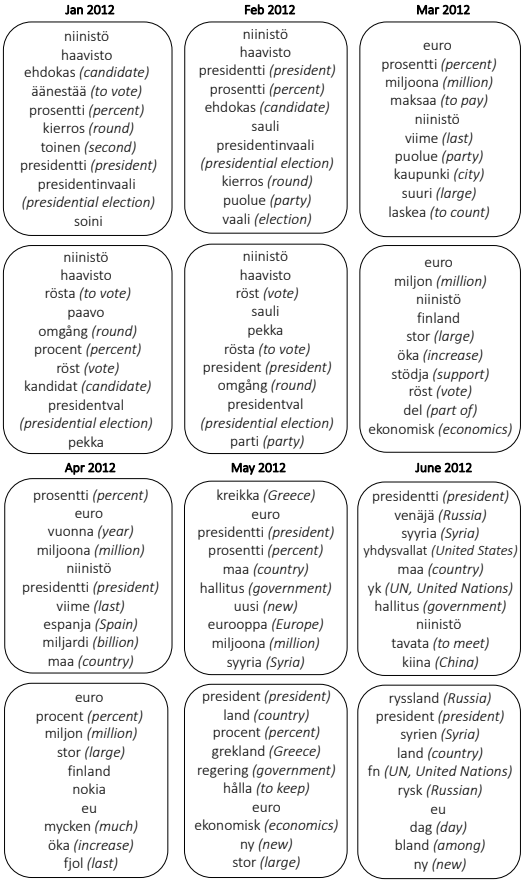


Figure 6: Top words of a topic on political news in Finland from the YLE dataset for Finnish (top) and Swedish (bottom) from January to June 2012. English translations of the words excluding named entities are enclosed in parentheses.

(sick-leave pay) where the terms 'sick' and 'pay' appear on the English side; and 'steuerreform' (tax reform) where 'reform' appears on the English side as well.

A named entity, 'thyssen', appears in March 1997 in both languages but not in other months. This is because of an event that happened around mid-March where the German steel company Thyssen was being bought by competitor Krupp-Hoesch (also a top term in the German topic) prompting concerns about job losses[6].

Figure 6 shows the first six months of a topic about political news from the YLE dataset. The first two months have terms related to presidential elections. These terms refer to the Finnish presidential election in 2012, where rounds of voting took place in January and February 2012[7]. These time slices also mention the two candidates in the runoff election, Sauli Niinistö and Pekka Haavisto. Sauli Niinistö eventually won the election which explains why the next time slices cease to mention Pekka Haavisto while 'niinistö' is still a prominent term. After March 2012, the topic stops talking about presidential elections and moves on to other political news. This gives us an insight into how the model can track significant events, such as high-profile elections, related to a topic. Another example is May 2012, where Greece ('kreikka' in Finnish, 'grekland' in Swedish) suddenly becomes a prominent term for both languages due to the Greek legislative elections which took place on 6 May 2012. The term

[6] https://www.nytimes.com/1997/03/19/business/krupp-hoesch-confirms-bid-of-8-billion-for-thyssen.html
[7] https://en.wikipedia.org/wiki/2012_Finnish_presidential_election

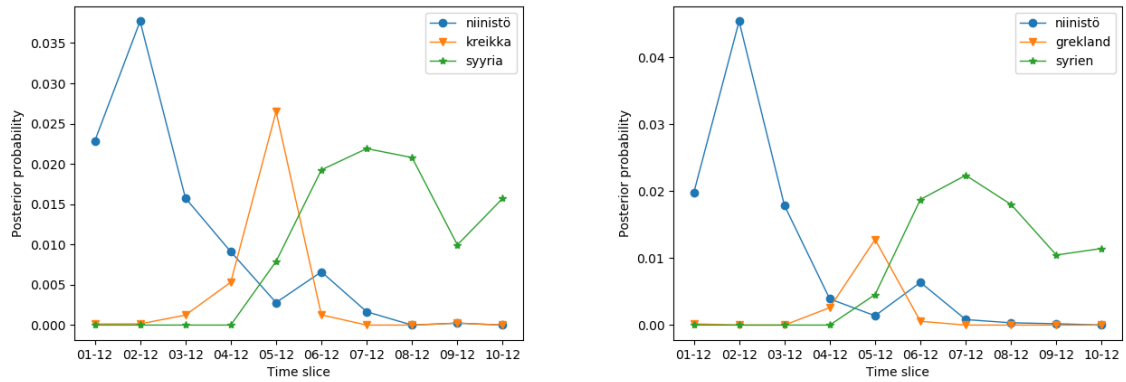


Figure 7: Posterior probabilities of salient terms in Finnish (left) and Swedish (right) related to events in the political news topic captured by ML-DTM from the YLE dataset.

'syyria'/'syrien' appears in May and June, corresponding to the beginning of the Syrian Civil War.

Figure 7 shows the posterior probabilities of some terms related to the presidential elections ('niinistö'), Greece ('kreikka' or 'grekland') and Syria ('syyria' or 'syrien') in the political news topic for both languages. We see the rise and fall of the prominence of the terms according to their relevance in the news.

2.4. Gaussian LDA

LDA is a generative model of the words of documents and topics are learned as categorical distributions over a fixed set of words in the vocabulary, derived from the words used in the training data. In the case of PLTM, the model contains such a distribution for each of the languages being modelled.

This approach to modelling words has a number of disadvantages. The set of words that the model can take into account when inferring topics for a document is limited to the words that occurred sufficiently frequently in the training corpus that the training inference routine could learn reliable probabilities for them. Low-frequency words, or those not seen at all in the training data, must be ignored. Furthermore, to learn meaningful connections between topics and words, each word (or, in fact, the precise form of each word) must have been seen numerous times, in order to be able to exploit its statistical co-occurrence with other words that typically appear in the same documents.

Gaussian LDA (GLDA, [4]) attempts to overcome these problems by making use of *word embeddings*, vector representations of the meaning of words, typically derived from observations of the contexts in which a word is used. Embeddings can be derived efficiently from large corpora in an unsupervised manner [20]. They have the general property that words that reside close to one another in the vector space have a similar or closely related meaning. Instead of a categorical distribution over the known words of the language, GLDA has a Gaussian distribution over the vector space, parameterized as a mean vector and a covariance matrix.

Each topic is now effectively a region of the vector space, whose position and size are updated during training. It is therefore able to exploit the similarity properties of the vector space. If the word *granite* does not occur frequently in the training collection of documents, but has been placed close to *rock*

in the vector space, topics that assign a high probability to *rock* will automatically also assign a high probability to *granite*.

In section 5, we report experiments to compare GLDA to other topic models in their application to noisy data, of the sort encountered in the NewsEye dataset. Using standard word embedding training methods, we can train embeddings on the NewsEye collection, learning representations appropriate to the historical data. They can have the effect of taking into account common ATR errors: for example, *serial* may be close to *ferial* in the space, if this corruption is common in the data. It is this ability of the model to smooth such noise in the input data that we test in our experiments.

In Year 2, we have produced a new Python implementation of GLDA, including the efficiency technique described by [4] using the Cholesky decomposition. The implementation is complete, but we have not yet applied the training to NewsEye data. The next step is to train word embeddings on the NewsEye collection and use these to train a GLDA model.

In Year 3, we also built on GLDA in a manner similar to [21] to create a new type of TM. A limitation of GLDA as it stands is that each topic consists of a single Gaussian over the vector space. Words that are not close in the original embeddings, but would be grouped by LDA on the basis that they often co-occur at the document level, cannot be included in a single topic, since the Gaussian distribution is unimodal. We will explore a multimodal extension of GLDA, where a topic is a categorical distribution over a fixed number of 'concept' components (shared across all topics) and each concept is represented by a (unimodal) Gaussian, learned as in GLDA. The *concept* is a distribution over the embedding space and thus groups together a region of closely related words. Compared to Gaussian LDA, this involves introducing an additional level of latent variables (the concept variables) into GLDA. We experimented with a number of different ways of defining such a model, but a full experimental comparison to the other embedding-based topic models is still ongoing.

3. Training models

To start running the topic modelling training infrastructure described in Section 3.1, we receive a dump of the dataset from the team in ULR that is in charge of the Demonstrator. This data dump contains the articles and their respective metadata. Once we have this data stored in our servers, we move to the steps of the **pre-processing phase**:

1. **Tokenization**: Tokenize the articles with a custom tokenizer to remove characters that might be OCR errors.
2. **Filtering of short tokens**: Remove tokens less than a specified number of characters, currently a small token is one which has only 1 character. This is primarily to remove punctuation marks and characters that will not be useful in topic modelling.
3. **Filtering of short documents**: remove documents less than a specified length (number of tokens) from the corpus. Currently, the threshold is 50 tokens for all datasets however we plan to make this threshold language-specific. Documents removed in this step will no longer be used in the rest of the pipeline. This reduces the corpus size which improves processing times for the rest of the pipeline and also prevents us from analyzing short documents. Topic distributions for such documents will not be reliable anyway.
4. **Lemmatization**: we lemmatize the corpus with language-specific lemmatizers provided by the

LAS toolkit. [8]

5. **Yearly sampling**: due to the high volume of documents, we need to sample our data for training such that it is reasonably representative of the corpus. Currently, we randomly select some number of articles per year (300 for LDA training data and 100 for DTM training data).

Once we have finished the steps above, we move on to the **training phase**:

1. **Vocabulary building**: we build the vocabulary for each dataset by scoring the terms in the corpusm using TF-IDF (term frequency-inverse document frequency) and keeping the terms with the top scores, currently we keep the top 20,000 terms as the vocabulary for the dataset.
2. **Topic model training**: this is the main topic model training step where topics are inferred from the sampled dataset that has passed through the pre-processing phase. Currently, we have training modules for LDA and DTM.

After the training phase, we proceed to the **post-processing phase**. Unlike the pre-processing and training phases, some of the steps in the post-processing phase are not in sequential order (they can be run in parallel).

1. **Corpus analysis**: This is the step where we infer the topic distributions of each of the documents in the corpus using the topic models we have trained in the training step. The topic models are frozen at this stage and are no longer updated from the unseen documents we pass to it.
2. **Visualization**: We generate several visualizations from the trained model such as word clouds for each topic, PyLDAVis visualization for LDA models [9] and interactive bar charts for DTM models. The visualizations generated here can be in the form of PNG files for images or HTML files for interactive charts. This can be done in parallel with the other steps in this phase.
3. **Database updates**: We store the document topic vectors from the previous step in the Solr index used by the Demonstrator so that other work packages can make use of them through API interfaces we provide (API specifications are detailed in Section 6.1).

3.1. Training infrastructure

For NewsEye, it is important that we are able to train new TMs on any given dataset, including the datasets we handle here, the full initial collection and updated versions as new ATR methods are applied. As new datasets are produced during the project, it will be necessary to update the TMs that we make available for use by other NewsEye components. Re-training the models involves not just re-running the parameter inference process, but also a pipeline of pre-processing steps, such as tokenization, lemmatization and, vocabulary filtering.

To develop a clearly documented, robust pipeline of preprocessing and model training that can be straightforwardly run on new datasets as we receive them, we use the **Pimlico processing toolkit**[10] [22]. Pimlico is a Python-based toolkit for defining complex pipelines of data-processing tools, running the pipelines on large datasets and managing input and output data between the different components. It has a particular focus on NLP tasks and text processing but is applicable to any pipelines processing large datasets. It is developed and maintained by the UH-CS team and a substantial amount of development has been carried out as part of this task to support topic model training and analysis.

[8] https://github.com/jiemakel/las-ws
[9] https://pyldavis.readthedocs.io/en/latest/readme.html
[10] https://github.com/markgw/pimlico

As we develop implementations of the various models, including the TMs described above, we also develope Pimlico pipelines to train the models on the datasets we have available. When new data becomes available, it is trivial to create a new version of any existing pipeline, read in the new dataset and train new models, following exactly the same procedure as previously.

Pre-processing steps described in the previous section have been implemented as Pimlico pipelines, so are easy to apply to new datasets and to share between different model types. Pimlico pipelines can be run on any personal computer or server.

Trained models are stored external to the training pipeline and transferred to a separate server which hosts the tools' API (see Section 6.1). The final steps of the pipeline analyse all the articles in the collection using the trained TM and store the analyses in the local mirror of the Demonstrator's database, ready for use by the tools in the Demonstrator and the PRA.

3.2. Input data acquisition

In order to train TMs on the data from the Demonstrator, produced by earlier WPs, we create a JSON dump of all articles in the given language set, directly from the mirror of the Demonstrator's database. These are read into the Pimlico pipeline for further processing, together with the year of publication of each article, provided as metadata.

3.3. Pre-processing

Typical pipelines for training TMs include preprocessing steps such as stopword removal and text normalization. We apply the same preprocessing steps to training the different types of TMs.

First, the articles are tokenized using a generic tokenizer. This applies to a few regular expressions to, for example, separate words from punctuation. It is specially designed to accommodate the high levels of noise from ATR, accounting, for example, for the fact that some characters may get spuriously replaced by punctuation. For now, a language-general set of rules is applied. We do not use a pre-existing language-specific tokenizer, as these would typically be misled by the ATR noise.

We filter out very short words, which typically are either noise resulting from ATR (e.g., spots on the page interpreted as punctuation) or otherwise carry little information about the content (e.g., page numbers or short function words). We also filter out any short whole documents (we used 20 tokens as the minimum but the threshold can be adjusted), since these will generally not contain enough data to distinguish any coherent topical information. (Note that this is applied to *training* documents – short documents can still be *analyzed* using the trained TM).

We apply lemmatization, using the LAS lemmatization tool [18]. This loads a lemmatizer specific to the known language and is designed to be more robust than other similar tools to noise from historical data, as well as historical spelling variants.

Next, we extract a vocabulary from the training corpora to use for the model during training. We exclude any terms that appear in more than 10% of documents, since these are uninformative to discerning topics, and exclude any terms that appear fewer than 30 times in the whole corpus since these will not have enough occurrences to learn reliable distributions from. Finally, if there are more than 20,000 remaining terms, we limit the vocabulary to the most frequent 20,000.

Since the training corpus is large, we randomly subsample documents in order to train the TMs. We sample a fixed number of articles from each year. In the first round of training, we have subsampled 300 articles per year for training LDA and 100 for DTM (since the training process is slower).

3.4. Trained models

The datasets for training these topic models are described below:

1. **German**: Articles from 23 non-consecutive years for 1895-1900, 1911-1922 and 1922-1937 from various German-language newspapers courtesy of the ÖNB.
2. **Finnish**: Articles from various Finnish-language newspapers from the NLF spanning 48 years from 1869-1917.
3. **French**: Articles from various French-language newspaper from the BnF spanning 30 years, from 1915 to 1944.

The following topic models are available for inspection in the NewsEye platform. This means that through the NewsEye platform, the user can inspect visualizations of the topic model, see textual topic descriptions and use topic distributions of documents for analysis.

1. **LDA-DE**: LDA trained on 100 topics from the German dataset.
2. **LDA-FI**: trained on the Finnish dataset.
3. **LDA-FR**: trained on the French dataset.
4. **DTM-DE**: DTM trained for 23 time slices and 50 topics from the German dataset
5. **DTM-FI**: trained for 50 topics and 48 time slices from the Finnish dataset.
6. **DTM-FR**: trained for 50 topics and 30 time slices from the French dataset.

4. Document linking

4.1. Monolingual article linking

We use trained monolingual LDA topic models to link articles from the same language. Given the topic distribution of a query document, we go through each of the topic distributions of the other documents that have been trained on the same model and compute their Jensen-Shanon divergence (JSD). We then rank the candidate documents in ascending order where the high-ranking documents have smaller divergence (more similar to the query document) and the low-ranking ones have higher divergences.

In cases where the query is a set of documents rather than an individual document, we take the *mean topic distribution* of the set and do the same procedure to rank the candidate documents.

Since the topic distributions of all the documents in the corpus are already stored in a database (except the ones we have filtered out due to their short length), this procedure is a simple matter of running database queries and computing divergences. Although this procedure is straightforward, the high volume of documents makes lookup very time-consuming even for just a single query. We improved this process using some hashing algorithms to make lookups more efficient (Section 4.2). These basic comparison methods are already available through the API described in Section 6.1.

4.2. Scalable article linking

The NewsEye collection is composed of tens of millions of documents even if we consider only a single language. This makes the document linking method outlined above unfeasible especially when multiple queries are run at the same time. The bottleneck in the process comes from loading the matrices that contain the document-topic distributions and then computing the document similarities (as quantified by JSD) between the query document and *all* the candidate documents and ranking the documents according to their similarity.

To reduce the time and space complexity of this process, we first compute a hash table of all the documents in a dataset using locality-sensitive hashing (LSH). LSH is a hashing technique that places similar items in the same bucket [23]. This can be thought of as a kind of clustering of similar items. Given a query Q, we can return the N most similar items quickly based on the computed hash table. Below we outline our method for scalable topic-based article linking.

During development time We infer the document-topic distributions of all the documents in our dataset with a trained topic model. This results in a matrix $M \in \mathbb{R}^{D \times K}$ where D is the number of documents in the dataset (this would be in the order of 1×10^6) and K is the number of topics (in the NewsEye platform, we use $K = 100$). $M_{i,k} \in \mathbb{R}$ is the probability of topic k in document i.

1. For each row i in M:
   - for each topic probability p in $M[i]$, if $p > thresh$ then set $p = 1$ else $p = 0$.
2. Compute hash table using the binary matrix from step (1). This hash table is stored in the server where it will be used during query time.

During query time Queries are passed to WP4 in the form of document-topic distributions. This is the normalized document-topic distribution of a single document or a collection of documents.

1. During query time, given a query distribution Q, transform Q to Q_b s.t. $Q[k] > thresh$ is set to 1 else 0.
2. Query the hash table to get the c items closest to Q_b. This returns article IDs of the c closest articles to the query. This is our shortlisted articles
3. For each article ID in the shortlist, query the article's document-topic distribution from the Solr index and compute JSD between this and the query Q.
4. Rank the shortlisted articles according to the JSD computed in the previous step. Return the n top-ranking articles. We set c to be $2n$ where n is the number of articles requested.

Using this method, we can speed up queries for similar documents from approximately 10 minutes to under a minute. This speed-up comes from two factors: (1) we are no longer loading huge matrices in memory and, (2) we are no longer computing JSD for all documents in matrix M, instead we are only computing it for the c candidate articles where $c << D$.

4.3. Cross-lingual article linking

We investigate the use of PLTMs in linking articles across languages and compared the performance of these models with methods that make use of cross-lingual embeddings.

In particular, we compare the performance of three methods in the task of ad hoc cross-lingual document retrieval: PLTM, cross-lingual document embeddings, and cross-lingual word embeddings with cross-lingual distance measures.

4.3.1. Polylingual topic model

One of the main advantages of PLTM is that it can extend across any number of languages, not just two, as long as there is a topically aligned corpus covering these languages. This can be difficult because aligning corpora is not a trivial task, especially as the number of languages gets larger. For this reason, Wikipedia, currently in more than 200 languages, is a popular source of training data for PLTM.

Another issue facing topic models is that the choice of hyperparameters can significantly affect the quality and nature of topics extracted from the corpus and, consequently, its performance in the downstream task we want to use it for. There are three main hyperparameters in LDA-based models: the number of topics to extract, K; the document concentration parameter, α, that controls the sparsity of the topics associated with each document; and the topic concentration parameter, β, which controls the sparsity of the topic-specific distribution over the vocabulary.

4.3.2. Cross-lingual document embeddings

Cross-lingual reduced-rank ridge regression (Cr5) was introduced as a novel method of obtaining cross-lingual document embeddings [8]. The authors formulate the problem of inducing a shared document embedding space as a linear classification problem. Documents in a multilingual corpus are assigned language-independent concepts. The linear classifier is trained to assign the concepts to documents, learning a matrix of weights W that embeds documents in a concept space close to other documents labelled with the same concept and far from documents expressing different concepts.

They show that the method outperforms the state-of-the-art cross-lingual document embedding method from previous literature. Cr5 is trained to produce document embeddings, but can also be used to obtain embeddings for smaller units, such as sentences and words.

4.3.3. Wasserstein distance for documents

Wasserstein distance is a distance metric between probability distributions and has been previously used to compute distances between text documents in the same language (*Word Mover's Distance* [24]). In [7] the authors propose the Wasserstein distance to compute distances between documents from different languages. Each document is a set of cross-lingual word embeddings [25] and each word is associated with some weight, such as its term frequency-inverse document frequency (tf-idf). The Wasserstein distance is then the minimum cost of transforming all the words in a query document to the words in a target document.

4.3.4. Ensembled models

We also create ensemble models from these methods by averaging the document distances from the stand-alone models and ranking candidate documents according to this score. We construct four ensemble models by combining each pair of models, as well as all three: **PLTM_Wass**; **Cr5_Wass**; **PLTM_Cr5**; and **PLTM_Cr5_Wass**.

4.3.5. Experimental setup

We evaluated the models using a dataset of Finnish and Swedish news articles published by the Finnish broadcaster YLE and available for download from the Finnish Language Bank[11]. The articles are from 2012-18 and are written separately in the two languages (not translations and not parallel). Each article is tagged with a set of keywords describing the subject of the article. These keywords were assigned to the articles by a combination of automated methods and manual curation. The keywords vary in specificity, from named entities, such as *Sauli Niinisto* (the Finnish president), to general subjects, such as *talous* (sv: *ekonomi*, en: economy). On average, Swedish articles are tagged with five keywords and 15 keywords for Finnish articles. Keywords are provided in Finnish and Swedish regardless of the article language so no additional mapping is required.

To build a corpus of related news articles for testing, we associate one Finnish article with one or more Swedish articles if they share three or more keywords and if the articles are published in the same month. From this, we created three separate test sets: 2013, 2014, and 2015. For each month, we take 100 Finnish articles to use as queries, providing all the related Swedish articles as a candidate set visible to the models.

To build a topically aligned corpus for training PLTM, we match a Finnish article with a Swedish article if they were published within two days of each other and share three or more keywords. To train MLTM we use a year which is preceding the testing year: e.g., we train a model using articles from 2012 and test it on articles from 2013.

4.3.6. Results and Discussion

Table 7 shows the results for each model and ensemble on each of the three test sets, reporting the precision of the top-ranked k results and mean reciprocal rank (MRR). Cr5 is the best-performing stand-alone model by a large margin. Cr5 was originally designed for creating cross-lingual document embeddings by classifying Wikipedia documents according to concepts. We did not retrain it for our particular task. Nevertheless, using these pre-trained word embeddings we were able to retrieve articles that discuss similar subjects in a different domain.

However, it is worth noting that Cr5 can only be trained on languages for which labels are available for *some* similarly transferable training domain.

[11] https://www.kielipankki.fi/corpora/

Table 7: Precision at k and MRR of cross-lingual linking of related news articles obtained by three stand-alone models and four ensemble models.

Test set:	2013				2014				2015			
Measure:	**P@1**	**P@5**	**P@10**	**MRR**	**P@1**	**P@5**	**P@10**	**MRR**	**P@1**	**P@5**	**P@10**	**MRR**
PLTM	21.8	18.2	16.3	31.6	24.1	22.4	20.6	34.8	30.8	29.0	27.1	41.6
Wass	21.1	13.7	11.3	30.8	21.0	16.9	14.7	31.9	25.1	20.6	17.9	37.2
Wass $\lambda = 0.01$	20.3	13.5	11.1	30.0	21.3	16.8	14.6	32.0	25.1	20.1	17.3	36.6
Cr5	32.5	24.5	21.2	41.7	38.3	30.2	26.0	48.0	43.1	37.1	33.5	53.8
PLTM_Wass	24.6	21.3	19.1	35.2	27.3	25.5	23.4	38.2	30.4	31.4	30.1	42.9
Cr5_Wass	35.4	27.4	23.2	45.2	38.1	32.2	28.2	49.2	41.2	37.7	34.9	52.9
PLTM_Cr5	36.4	28.2	24.4	46.6	**44.8**	34.3	30.1	53.6	42.7	40.1	36.9	54.5
PLTM_Cr5_Wass	**40.7**	**30.7**	**26.3**	**50.3**	43.0	**36.1**	**31.9**	**53.8**	**44.5**	**41.3**	**38.5**	**55.9**

Table 8: Mean Spearman correlation of the ranks of candidate documents for each pair of models.

Test set:	2013	2014	2015	AVG
MLTM, Wass	-0.039	-0.016	-0.022	-0.026
Cr5, Wass	0.128	0.027	0.026	0.060
MLTM, Cr5	0.156	0.164	0.178	0.166

PLTM, being a topic-based model, would seem like the obvious choice for a task like this because we want to find articles that share some broad characteristics with the query document, even if they do not discuss the same named entities or use similar words. However, Cr5 outperforms PLTM on its own. One reason may be that 100 topics are too few. We chose this number because it seemed to give topics that are specific enough for short articles but still broad enough that they could reasonably be used to describe similar articles.

Wasserstein distance is the worst-performing of the stand-alone models especially for the 2014 and 2015 test sets where it offers little improvement when ensembled with Cr5 (Cr5_Wass). A possible reason is that it attempts to transform one document to another and therefore favours documents that share a similar vocabulary to the query document. The technique might be suitable for matching Wikipedia articles, as shown in [7] because they talk about the same subject at a fine-grained level and use similar words, whilst in our task the goal is to make broader connections between documents.

For the ensemble models, combining all three models performs the best overall for all three test sets and all but one precision level — the only exception is P1 for 2014 where PLTM_Cr5 achieves roughly the same performance. This tells us that each model sometimes finds relevant documents not found by the other models. The correlation of candidate document rankings between the different methods is quite low (Table 8). We compute the correlation between the ranks for each of the 1200 query documents (100 queries for each month) for each year of our test set and average them. As can be seen in the table the correlations are rather low, which means that they retrieve documents based on different principles. The highest correlation is between PLTM and Cr5 while the correlation between MLTM and Wasserstein is the lowest.

This suggests that there are different ways of retrieving related documents across languages and that the three methods of cross-lingual embeddings, cross-lingual topic spaces, and cross-lingual distance measures capture complementary notions of similarity. A simple combination of their decisions is thus able to make better judgements than any can make on its own.

4.3.7. Summary and future work

We compare three different methods for cross-lingual ad-hoc document retrieval by applying them to the task of retrieving Swedish news articles that are related to a given Finnish article. We show that a word-embedding based model, Cr5, performs best followed by PLTM and the distance-based Wasserstein model has the worst results of the stand-alone models. We also demonstrate that combining at least two of these methods by averaging their distances yields better results than the models used on their own. Finally, we show that combining the three models yields the best results.

We plan to investigate the performance of Gaussian LDA with multilingual embeddings. Such model could potentially combine the benefits of the multilingual topic model with word embeddings for retrieving similar documents across languages.

5. Handling noisy input

5.1. Robustness to OCR noise

In this section, we present the results of our quantitative assessment of the performance of two embedding-based models, Gaussian LDA (GLDA) and the Embedded Topic Model (ETM), on datasets with OCR noise. GLDA is discussed more in Section 2.4. ETM uses word embeddings during topic inference by learning a topic embedding which is a point in the embedding space and training an inference network to learn the parameters of the logistic normal distribution from which the document-topic distributions are drawn [26].

The aim of our experiments is to test whether embedding-based models can be used to improve the analysis of digitised historical documents. Our results showed that these models, especially ETM, are more resilient than LDA in the presence of noise in terms of topic quality and classification accuracy.

5.1.1. Methodology

Following [27], we first evaluated the topic models on a corpus of historical documents with real OCR noise, with aligned gold standard (GS) text. And then we evaluated the models on a larger corpus where synthetic noise has been introduced at increasing levels.

5.1.2. Datasets

Real noise The Overproof NLA dataset [28] consists of 30,301 digitised news articles from the Sydney Morning Herald 1842–1954, from the archives of the National Library of Australia[12]. The OCRed articles have a word error rate (WER) of 25% [29]. The OCR and GS articles are aligned on a character level.

[12] http://overproof.projectcomputing.com/datasets/

Synthetic noise To generate data with synthetic noise, we start with a clean dataset and gradually corrupt the data by introducing noise at increasing levels. For our clean data, we use the Reuters RCV1 dataset, consisting of over 800K English newswire articles with pre-assigned categorical labels [30]. We use a reduced dataset of 50K articles sampled from the largest categories. We followed the procedure of [27] to generate synthetic noise based on a noise model constructed from a dataset with real noise.

5.1.3. Models

LDA We use LDA as a baseline model. We trained LDA models using the Gensim library, which uses variational inference to infer topics [31].

ETM For ETM, we used the authors' implementation[13]. We ran inference for 1000 epochs with default hyperparameters.

GLDA For GLDA, we used the `gaussianlda` Python package, which implements Gibbs sampling with Cholesky decomposition and alias sampling to reduce sampling complexity [4][14]. We ran the sampler for 20 iterations, based on initial experiments with the clean *20-Newsgroups* dataset.

5.1.4. Evaluation measures

Topic coherence Topic coherence quantifies the interpretability of a topic as represented by its most probable terms. Coherence measures based on pointwise mutual information (PMI) of word pairs, such that words that tend to appear together in the same documents have better scores, have been found to correlate well with human judgement. We use the C_v coherence measure proposed by [32].

Topic diversity Models that learn a wide variety of topics are preferable to models with redundant topics. We measure topic diversity as the proportion of unique words out of all the top words representing all the topics in the model [26].

Classification accuracy We evaluate the quality of the per-document topic proportions inferred by the models through a supervised document classification task. We train a classifier on a portion of the data using the inferred topic proportions as features and pre-assigned categories as labels, then test the classifier on unseen documents [27].

5.1.5. Experimental Results

Results on real noise data Figure 8 show the results of our experiments with real noise data. In terms of topic coherence, almost all the models perform better on the GS documents than the OCR documents, as would be expected (Figure 8a).

[13] https://github.com/adjidieng/ETM
[14] https://pypi.org/project/gaussianlda/

ETM with Overproof embeddings has similar topic coherences to LDA–both have a coherence of 0.57 for OCR while for GS, ETM is only a little better with a mean coherence of 0.62 and LDA has 0.6.

Results on synthetic data Figure 9 shows our experimental results on the corrupted Reuters data for increasing levels of noise. Figure 9a indicates that both ETM and LDA degrade linearly in coherence as noise increases, though the former degrades more slowly than the latter. At a CER of 0% (no noise), coherences for both models are quite similar (0.60 for LDA and 0.61 for ETM), while at the highest noise setting (35% CER), LDA coherence is 0.27 while ETM is 0.36, a difference of almost ten percentage points.

5.1.6. Conclusions from the study

We found that using embeddings trained on the same data as the topic models produces more coherent topics than embeddings trained on Wikipedia although the latter is a larger and cleaner dataset. We reasoned that this is due to the difference in time periods between Wikipedia and the Overproof data. Therefore when using these embedding-based models on historical corpora, it is important to also use word embeddings matching the time period and genre of the target corpus. This area is worth further investigation in future work. We also noted the qualitatively dissimilar nature of ETM and GLDA topics and the high correspondence of ETM topics from OCR data with topics from the aligned GS data.

We are currently preparing a manuscript for submission. A draft is attached to the Appendix.

5.2. Impact of noisy article segmentation on topic modelling

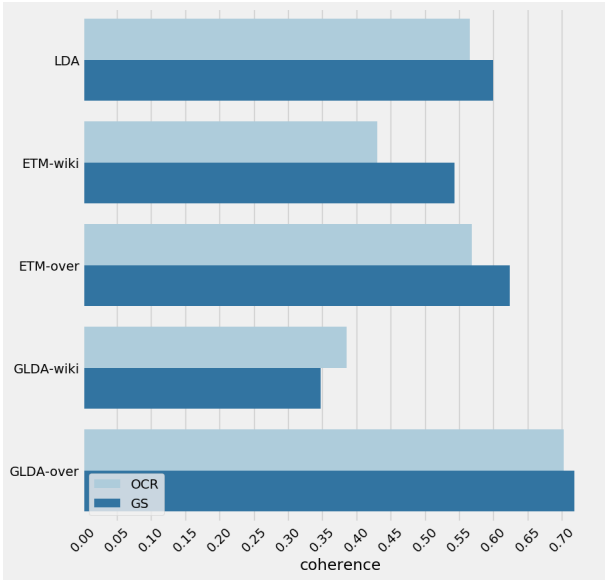
There are several existing work that use topic models to segment text into coherent articles. To our knowledge, however, there is no existing study that systematically quantifies the impact of incorrect text segmentation on the results of topic models. Here we present the results of a preliminary study we performed to evaluate the impact of noisy text segmentation on topics.

5.2.1. Datasets

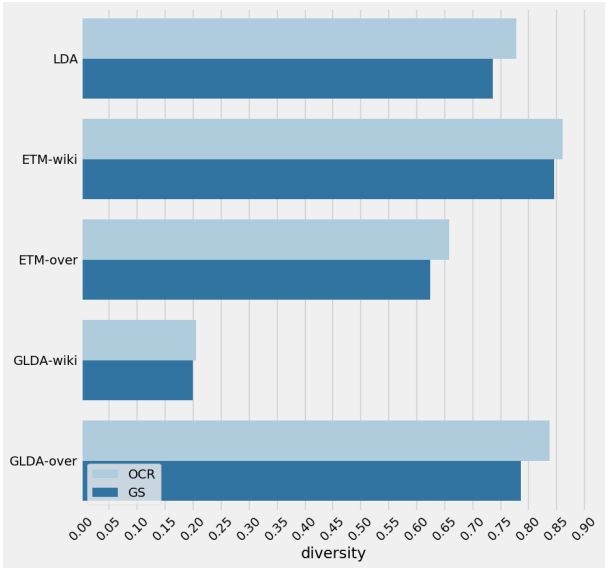
We construct a dataset of artificially segmented text following [33]. We use the Reuters RCV1 corpus for out experiments. From this corpus we created three data subsets: (1) 3-5 segments; (2) 6-8 segments and, (3) full (unsegmented). The 3-5 subset divides articles into n approximately equal segments where n is a randomly sampled value ranging from $[3, 5]$. Thhe 6-8 subset divides articles in 6-8 segments and the full subset is the original article (ground truth segmentation).

5.2.2. Experiments

We extracted 20 and 50 topics using LDA from the three subsets described above. To measure the resulting topic quality, we use normalized pointwise mututal information (nPMI) which has been shown to correlate well with human judgment [32].

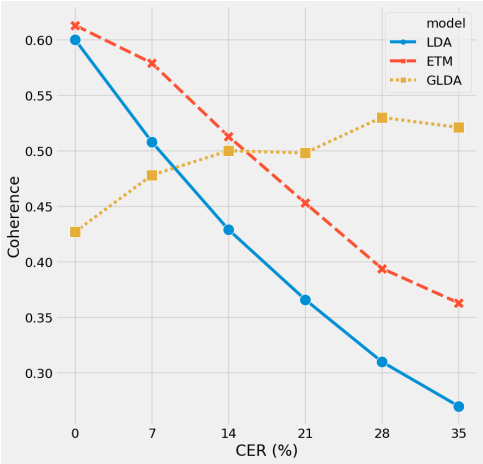


(a) Topic coherence

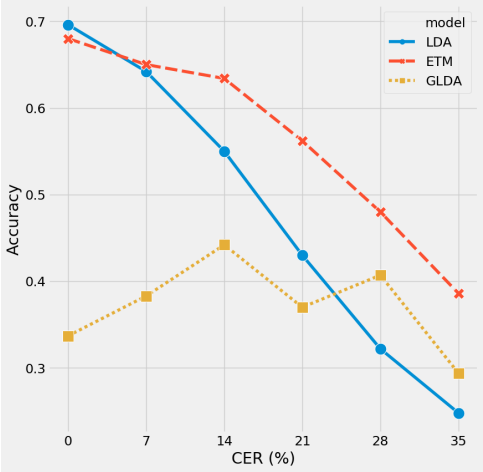


(b) Topic diversity

Figure 8: Performance on real noise data. *wiki* models use word embeddings trained on Wikipedia while *over* models use embeddings trained on the Overproof data. Higher values are better.



(a) Topic coherence on the training corpus



(c) Mean classification accuracy

Figure 9: Performance on synthetic noise data. Higher values are better.

5.3. Results

Table 9 shows the coherence scores from different methods of artificially segmenting articles. For 20 topics, segmenting the text in 3-5 segments does not degrade the resulting topics too much with nPMI score close to the full (unsegmented) model. There is more degradation though when the segments get shorter, as in the 6-8 segments. It is in the models trained for 50 topics that we begin to see the negative impacts of incorrect segmentation with the 3-5 and 6-8 segments having negative nPMI scores. The coherence of the full model, in contrast, remains close to the its coherence for 20 topics.

	3-5	**6-8**	**full**
20 topics	0.044	0.025	**0.050**
50 topics	-0.029	-0.050	**0.042**

Table 9: normalized PMI coherence scores for models trained on different text segmentations. Higher is better.

5.4. Conclusions

We have shown that incorrect text segmentation negatively impacts the outputs of topic modelling and in the previous section we have also shown that text with OCR noise also does the same. We would recommend a quantitative study that investigates the combined impact of both incorrect text segmentations and noisy OCR-ed text on the results of topic modelling algorithms.

6. Querying TMs

Access to analyses from trained TMs is made available to other components of NewsEye via an API described below (Section 6.1). This is used both by the Demonstrator, for direct incorporation of TM analysis into the user interface, and by the autonomous Investigator, to allow it to analyse small document collections and discover potentially interesting trends and contrasts.

The main result of TM analysis, the document-topic distributions of articles, are stored in the same Solr index used by the Demonstrator where it can be used by the Personal Research Assistant (PRA) without going through the WP4 API. For this reason, the methods available in the WP4 API concentrate mainly on document linking, topic description, and topic visualization. Most of the methods available through the API described below are already integrated into the Demonstrator and work is ongoing to provide access to those remaining.

For any given type of TM (DTM, for example), multiple models may be trained, differing in:

- model hyperparameters (e.g., number of topics);
- training dataset (e.g., different subcorpora);
- version of the data (e.g., incorporating new ATR versions from WP2);
- language(s) (e.g., monolingual TMs trained on different languages).

The input to an analysis is a set of documents to analyse. Since the whole NewsEye project uses a single collection, contained in the Demonstrator's database, this set of documents can be specified by giving a list of IDs that can be used to retrieve the documents from that database (a JSON list of strings). The tools provided here have access to a local mirror of the Demonstrator's database for faster access.

This mirror is regularly synchronized so that the document content corresponding to the given IDs is up to date.

6.1. REST API

The following is a specification for the current version of the REST API. This is now available to project partners and most of the calls described are integrated in some form into the Demonstrator.

`/lda/list-models`

GET: List available trained LDA models

This call lists trained **LDA models**. Other similar `list-models` calls will be provided for the other model types, with identical inputs and outputs.

- `/dtm/list-models`

Query parameters:
(None)

Returned status codes:
`200`: The results are included in the response.

Returned body (*application/json*):
`models` *list of objects*: List of available models. Each is an object containing:

- `name` *string*: Name used to refer to model
- `description` *string*: Human-readable description of the model, containing details like what training set was used
- `lang` *string*: Language code of the trained model
- `num_topics` *int*: Number of topics in the model

`/dtm/valid-years`

POST: List of years covered by a trained DTM model

Body parameters:
`model_name` *string*: Name of trained DTM mode

Returned status codes:
`200`: The results are included in the response.
`404`: The specified model does not exist or is not an LDA model

Returned body (*application/json*):
`years` *list*: List of years

/doc-linking-by-distribution

POST: Find similar documents according to topic similarity given a topic distribution.

This call returns a task ID which will be used to retrieve the result of this task (see `/doc-linking-results/`)

Body parameters:
`lang` *string*: Language code of the input documents
`model_type` *string*: Type of topic model used to infer topic distribution. Options are 'lda' or 'dtm'.
`num_docs` *int*: Number of document IDs to return
`topics_distrib` *list of floats*: Query vector representing the topic distribution of the query.

Returned status codes:
`200`: The task was accepted. Task UUID is included in the response.

Returned body (*application/json*):
`task_uuid` *string*: The UUID of the task that will be triggered by this query. This can be used to retrieve the results later with another `POST` request (see `/doc-linking-results/`).

/doc-linking-results

POST: Retrieve the results of a document linking task

Body parameters:
`task_uuid` *string*: UUID of the task whose results we want to retrieve

Returned status codes:
`200`: The results are included in the response.
`202`: The task was accepted, but is still running at the time the response is sent.

Returned body (*application/json*):
`similar_docs` *list of strings*: List of document IDs most similar to the input documents.
`distance` *list of floats*: Distance of document (computed by some distance measure) from the input documents.

/lda/pyldavis

POST: PyLDAVis visualization of a trained LDA model

This call returns an HTML file containing the PyLDAVis visualization of the trained LDA model.

Body parameters:
`model_name` *string*: Name of trained LDA model to visualize

Returned status codes:
`200`: The results are included in the response.
`404`: The specified model does not exist or is not an LDA model

Returned body (*application/json*):
`pyldavis` *html*: The HTML file output of the PyLDAVis visualization library.

`/dtm/bar_chart`

POST: Interactive bar chart visualization of a trained DTM model

This call returns an HTML file containing the visualization of the trained DTM model.

Body parameters:
`model_name` *string*: Name of trained DTM model to visualize

Returned status codes:
`200`: The results are included in the response.
`404`: The specified model does not exist or is not an LDA model

Returned body (*application/json*):
`bar_chart` *html*: The HTML file output with the bar chart and interactive features.

`/lda/top-words`

POST: Top words of a given topic in a trained LDA model

This call returns the top words of a given topic in a **LDA model**.
Other similar `top-words` calls will be provided for the other model types. These will have identical inputs and outputs, except that the non-dynamic models will not accept a `time_slice` parameter:

- `/dtm/top-words`

Body parameters:
`model_name` *string*: Name of trained model to describe
`topic_id` *string*: Topic ID starting with 1
`time_slice` *int*: Time slice starting with 1 or a year (see `/dtm/valid-years`). This is only for DTM models.

Returned status codes:
`200`: The results are included in the response.
`404`: The specified topic does not exist.

Returned body (*application/json*):
`topic_desc` *string*: Human-readable description of the topic in the specificed language

`/lda/word-cloud`

POST: Word cloud from a trained LDA model

This call returns a word cloud of a given topic in a **LDA model**. Other similar `word-cloud` calls will be provided for the other model types. These will have identical inputs and outputs, except that the non-dynamic models will not accept a `time_slice` parameter.

- `/dtm/word-cloud`

Body parameters:
`model_name` *string*: Name of trained model
`topic_id` *string*: topic ID starting with 1
`time_slice` *int*: time slice starting with 1 or year (see `/dtm/valid-years`)
`lang` *string*: language code

Returned status codes:
`200`: The results are included in the response.
`404`: The specified topic does not exist.

Returned body (*application/json*):
`topic_cloud` *image*: Word cloud of the specified topic in the specified language

7. Code

The tools developed in WP4 are provided to other WPs via the API described in Section 6. This is intended to provide all access to these tools required by other components in NewsEye.

The code for training and using the novel topic modelling techniques described above is written in Python and uses a number of standard toolkits for Bayesian modelling, topic modelling and statistical inference: Numpy, Tensorflow, Gensim, and Keras.

We released the source code for our training and analysis pipelines in Github as public code repositories:

- Training LDA and DTM models with Gensim: https://github.com/NewsEye/Training-Topic-Models
- Cross-lingual document linking with topic models and word embeddings: https://github.com/NewsEye/cross-lingual-linking
- Analysis of discourse dynamics: https://github.com/COMHIS/article_2020_disappearing-discourses
- Training of PLTM models: https://github.com/NewsEye/Multilingual-Topic-Model
- Implementation of ML-DTM: https://github.com/NewsEye/Multilingual-Dynamic-Topic-Modeling
- Gaussian LDA implementation: We will soon release code for our implementation of Gaussian LDA in Python, including the efficiency improvements described by [4] (code was originally released in Java).

Other code bases for analysis tools and processing pipelines is currently available to project partners and will be released publicly before the end of the project.

8. Use by Digital Humanities collaborators

We have ongoing research collaborations with digital humanities scholars in the NewsEye consortium. We presented work done with the University of Helsinki DH group (UH-DH) on investigating the dynamics of discourses in nineteenth-century Finnish newspapers. This work was presented in the digital humanities in the Nordic Countries Conference (DHN 2020, online due to travel restrictions). A full paper has also been accepted in the DHN 2020 post-conference proceedings (a copy of the manuscript is attached to this deliverable).

9. Conclusion

In this deliverable, we reported on the progress of T4.1 in the following areas:

Multilingual TMs. We investigated and implemented several methods for modelling multilingual collections, in particular, PLTM and used the trained model to link modern news articles in Finnish and Swedish.

Dynamic TMs. We implemented existing methods in this area and applied them to NewsEye data. In particular, we have applied a complete training and analysis pipeline for DTM to NewsEye subcorpora and made the results available via the Demonstrator and partially integrated into the Investigator.

We have developed our own novel method for multilingual, dynamic topic modelling, ML-DTM. We performed extensive experiments, published the results [2] and made the code publicly available.

Word embedding-based TMs. We investigated TMs that exploit word embeddings and some relevant extensions. We also studied the robustness of these models on noisy text data. We described here existing work in this area. We hope that this will provide further modelling benefits to the NewsEye pipeline.

NewsEye training and analysis pipeline. Together with other partners, we integrated some of the developed topic modelling methods into a complete pipeline of model training and document analysis for three monolingual subsets of the NewsEye collection.

Document linking. We use topic models to search for documents related to a query document by computing the divergences between their topic distributions. We developed a scalable document linking method that is suitable for very large document collections such as the NewsEye collections. This functionality is now available through the REST API of WP4.

Collaboration. We worked with digital humanities scholars in the NewsEye consortium to apply our methods to their research questions. We published work on exploring discourses that have disappeared or declined over time in the Finnish newspapers from the mid-nineteenth century until 1918. These collaborations have been extremely fruitful during the project, leading to valuable insights into the techniques themselves, as well as contributions to DH research. These efforts will continue and should lead to further contributions in DH.

References

[1] David Mimno, Hanna M Wallach, Jason Naradowsky, David A Smith, and Andrew McCallum. “Polylingual topic models”. In: *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*. Association for Computational Linguistics. 2009, pp. 880–889.

[2] Elaine Zosa and Mark Granroth-Wilding. “Multilingual Dynamic Topic Model”. English. In: *RANLP 2019 - Natural Language Processing a Deep Learning World*. Ed. by Galia Angelova, Ruslan Mitkov, Ivelina Nikolova, and Irina Temnikova. International Conference on Recent Advances in Natural Language Processing. Bulgaria: INCOMA, 2019, pp. 1388–1396. ISBN: 978-954-452-055-7.

[3] Jordan Boyd-Graber and David M Blei. “Multilingual topic models for unaligned text”. In: *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*. AUAI Press. 2009, pp. 75–82.

[4] Rajarshi Das, Manzil Zaheer, and Chris Dyer. “Gaussian LDA for Topic Models with Word Embeddings”. In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Beijing, China: Association for Computational Linguistics, July 2015, pp. 795–804.

[5] Jagadeesh Jagarlamudi and Hal Daumé. “Extracting multilingual topics from unaligned comparable corpora”. In: *European Conference on Information Retrieval*. Springer. 2010, pp. 444–456.

[6] Ellen M Voorhees and Donna Harman. “Overview of TREC 2003.” In: *Trec*. 2003, pp. 1–13.

[7] Georgios Balikas, Charlotte Laclau, Ievgen Redko, and Massih-Reza Amini. “Cross-lingual document retrieval using regularized wasserstein distance”. In: *European Conference on Information Retrieval*. Springer. 2018, pp. 398–410.

[8] Martin Josifoski, Ivan S Paskov, Hristo S Paskov, Martin Jaggi, and Robert West. “Crosslingual document embedding as reduced-rank ridge regression”. In: *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*. 2019, pp. 744–752.

[9] Robert Litschko, Goran Glavaš, Ivan Vulic, and Laura Dietz. “Evaluating resource-lean cross-lingual embedding models in unsupervised retrieval”. In: *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2019, pp. 1109–1112.

[10] Robert Litschko, Goran Glavaš, Simone Paolo Ponzetto, and Ivan Vulić. “Unsupervised cross-lingual information retrieval using monolingual data only”. In: *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 2018, pp. 1253–1256.

[11] David M Blei and John D Lafferty. “Dynamic topic models”. In: *Proceedings of the 23rd international conference on Machine learning*. ACM. 2006, pp. 113–120.

[12] Xuerui Wang and Andrew McCallum. “Topics over time: a non-Markov continuous-time model of topical trends”. In: *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2006, pp. 424–433.

[13] Xing Wei, Jimeng Sun, and Xuerui Wang. “Dynamic Mixture Models for Multiple Time-Series.” In: *Ijcai*. Vol. 7. 2007, pp. 2909–2914.

[14] Chong Wang, David Blei, and David Heckerman. “Continuous time dynamic topic models”. In: *arXiv preprint arXiv:1206.3298* (2012).

[15] Arnab Bhadury, Jianfei Chen, Jun Zhu, and Shixia Liu. “Scaling up dynamic topic models”. In: *Proceedings of the 25th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee. 2016, pp. 381–390.

[16] Masao Utiyama and Hitoshi Isahara. “Reliable measures for aligning Japanese-English news articles and sentences”. In: *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*. Association for Computational Linguistics. 2003, pp. 72–79.

[17] Thuy Vu, Ai Ti Aw, and Min Zhang. “Feature-based method for document alignment in comparable news corpora”. In: *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics. 2009, pp. 843–851.

[18] Eetu Mäkelä. “LAS: an integrated language analysis tool for multiple languages”. In: *The Journal of Open Source Software* 1 (2016).

[19] Ivan Vulić, Wim De Smet, Jie Tang, and Marie-Francine Moens. “Probabilistic topic modeling in multilingual settings: An overview of its methodology and applications”. In: *Information Processing & Management* 51.1 (2015), pp. 111–147.

[20] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. “Efficient Estimation of Word Representations in Vector Space”. In: *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2013.

[21] Weihua Hu and Jun’ichi Tsujii. “A Latent Concept Topic Model for Robust Topic Inference Using Word Embeddings”. In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*. 2016.

[22] Mark Granroth-Wilding. “Pimlico: A Toolkit for Corpus-Processing Pipelines and Reproducible Experiments”. In: *Proceedings of the 2nd Workshop on Natural Language Processing Open Source Software (NLP-OSS) at EMNLP*. 2020.

[23] Anand Rajaraman and Jeffrey David Ullman. *Mining of massive datasets*. Cambridge University Press, 2011.

[24] Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. “From word embeddings to document distances”. In: *International conference on machine learning*. 2015, pp. 957–966.

[25] Robyn Speer, Joshua Chin, and Catherine Havasi. “ConceptNet 5.5: An Open Multilingual Graph of General Knowledge”. In: *CoRR* abs/1612.03975 (2016).

[26] Adji B Dieng, Francisco JR Ruiz, and David M Blei. “Topic modeling in embedding spaces”. In: *Transactions of the Association for Computational Linguistics* 8 (2020), pp. 439–453.

[27] Daniel Walker, William B Lund, and Eric Ringger. “Evaluating models of latent document semantics in the presence of OCR errors”. In: *Proceedings of the 2010 conference on empirical methods in natural language processing*. 2010, pp. 240–250.

[28] John Evershed and Kent Fitch. “Correcting noisy OCR: Context beats confusion”. In: *Proceedings of the First International Conference on Digital Access to Textual Cultural Heritage*. 2014, pp. 45–51.

[29] Thi-Tuyet-Hai Nguyen, Adam Jatowt, Mickael Coustaty, Nhu-Van Nguyen, and Antoine Doucet. “Deep statistical analysis of OCR errors for effective post-OCR processing”. In: *2019 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*. IEEE. 2019, pp. 29–38.

[30] David D Lewis, Yiming Yang, Tony G Rose, and Fan Li. “Rcv1: A new benchmark collection for text categorization research”. In: *Journal of machine learning research* 5.Apr (2004), pp. 361–397.

[31] Radim Řehůřek and Petr Sojka. “Software Framework for Topic Modelling with Large Corpora”. English. In: *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. http://is.muni.cz/publication/884893/en. Valletta, Malta: ELRA, May 2010, pp. 45–50.

[32] Michael Röder, Andreas Both, and Alexander Hinneburg. “Exploring the space of topic coherence measures”. In: *Proceedings of the eighth ACM international conference on Web search and data mining*. 2015, pp. 399–408.

[33] Hemant Misra, François Yvon, Olivier Cappé, and Joemon Jose. “Text segmentation: A topic modeling perspective”. In: *Information Processing & Management* 47.4 (2011), pp. 528–544.

A. Manuscript: Topic Modelling Discourse Dynamics in Historical Newspapers

Topic Modelling Discourse Dynamics in Historical Newspapers

Jani Marjanen[1*][0000−0002−3085−4862], Elaine Zosa[2*][0000−0003−2482−0663], Simon Hengchen[3][0000−0002−8453−7221], Lidia Pivovarova[2][0000−0002−0026−9902], and Mikko Tolonen[1][0000−0003−2892−8911]

[1] Helsinki Computational History Group, University of Helsinki
[2] Department of Computer Science, University of Helsinki
[3] Språkbanken, University of Gothenburg[‡]
firstname.lastname@{helsinki.fi,gu.se}

Abstract. This paper addresses methodological issues in diachronic data analysis for historical research. We apply two families of topic models (LDA and DTM) on a relatively large set of historical newspapers, with the aim of capturing and understanding discourse dynamics. Our case study focuses on newspapers and periodicals published in Finland between 1854 and 1917, but our method can easily be transposed to any diachronic data. Our main contributions are a) a combined sampling, training and inference procedure for applying topic models to huge and imbalanced diachronic text collections; b) a discussion on the differences between two topic models for this type of data; c) quantifying topic prominence for a period and thus a generalization of document-wise topic assignment to a discourse level; and d) a discussion of the role of humanistic interpretation with regard to analysing discourse dynamics through topic models.

Keywords: discourse dynamics, Finland, historical newspapers, nineteenth century, topic modeling, topic modelling

1 Introduction

This paper reports our experience on studying discursive change in Finnish newspapers from the second half of the nineteenth century. We are interested in grasping broad societal topics, discourses that cannot be reduced to mere words, isolated events or particular people. Our long-lasting goal is to investigate a global change in the presence of such topics and especially finding discourses that have disappeared or declined and thus could easily slip away in modern research. We believe that these research questions are better approached in a data-driven way without deciding what we are looking for beforehand, though the choice of the most suitable techniques for such research is still an open problem.

In this paper we focus on developing methodology. Choosing available algorithms for analysis guides possible outcomes as they are designed to be operationalised in

[‡] SH was affiliated with the University of Helsinki for most of this work.
[*] Equal contribution.

certain ways. Approaching our goal with mere word counts is counterproductive due to the sparseness of the language and the variety of discourse realisations in a given text. Further, word counts are unreliable with historical data due to never ending language change, spelling variations and text recognition errors.

Thus, as many other papers in the area of digital humanities, we utilize topic modelling as a proxy to discourses. In particular, we apply the "standard" Latent Dirichlet Allocation model [3, LDA] and its extension the Dynamic Topic Model [2, DTM], which is developed specifically to tackle temporal dynamics in data. However, any model has its limitations and tends to exaggerate certain phenomena while missing other ones. We focus on the difference between models and try to reveal their limitations in historical data analysis from the point of view that is relevant for historical scholarship.

Our main contributions are the following:

- We propose a **combined sampling, training and inference procedure** for applying topic models to large and imbalanced diachronic text collections.
- We discuss differences between two topic models, paying special attention to how they **can be used to trace discourse dynamics**.
- We propose a method to quantify **topic prominence for a period** and thus to generalize document-wise topic assignment to a discourse level.
- We **acknowledge and discuss the drawbacks of topic stretching,** which is typical for DTM. It is commonly known that DTM sometimes represents topics beyond the time period, but thus far there is no discussion in how researchers should tackle this for humanities questions.

In order to illustrate the appropriateness of the proposed methodology we discuss two use cases, one relating to discourses on church and religion and one that relates to education. The role of religion and education has been studied extensively in historical scholarship but there are no studies that deal with these topics through text mining of large-scale historical data. These two topics were chosen due to the the fact that the former was in general a discourse in decline relating to the process of secularization in Finnish society, whereas the latter increased in the second half of the nineteenth century and relates to the modernization of Finnish society and the inclusion of a larger share of the population in the sphere of basic education. In addition to these two interlinked discursive trends, we also use other examples to illustrate the strengths and weaknesses of LDA and DTM for this type of historical research.

2 Data

Our dataset is from the digitised newspaper collection of the National Library of Finland (NLF). This dataset contains articles from *all* newspapers and most periodicals that have been published in Finland from 1771 to 1917. Several studies have used parts of this dataset to investigate such issues as the development of the public sphere in Finland, the evolution of ideological terms in nineteenth-century Finland and the changing vocabulary of Finnish newspapers [36, 17, 16, 11, 21, 22, 25, 29, 12].

The full collection includes articles in Finnish, Swedish, Russian, and German. In this work we focus only on the Finnish portion starting from 1854 because this is the point where we determined we have sufficient yearly data to train topic models. The resulting subset has over 3.6 million articles and is composed of over 2.2 billion tokens. Figure 1a shows that the number of tokens published per year in Finnish-language papers increased steadily. The average article has 526 tokens but article length varies widely from year to year, as seen in Figures 1b and 1c which show the average article length and the number of articles per year. As made clear by these figures, there is a noticeable difference in the number of articles and average article length after 1910. This shift does not reflect the actual articles in the newspapers, but is the result of a change of OCR engine used to digitise the collection [20]. While the raw data is publicly available, we used the lemmatised version of the newspaper archive produced by Eetu Mäkelä, whom we thank.

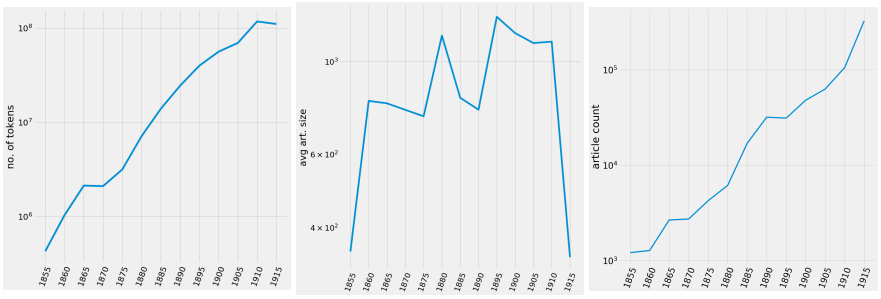
Still, even if the article segmentation differs in the latter period, Fig. 1a shows that there is steady increase in the vocabulary used in the Finnish-language newspapers published in the second half of the nineteenth century. They also covered more themes and regions. This entailed a process of diversification and modernization of the Finnish press, which has been widely discussed in historiography. As a collection, the newspapers vary a lot in style and focus. Some larger newspapers mainly contain political content, whereas others are rather specialised, and yet others thrived by giving a voice to the local public [35, 22, 16, 32]. This means that any analysis done on the entirety of the newspapers, like topic models, tend to balance out some of the differences between newspapers. This variety in the content, is also something that make newspapers such an interesting source material for historical research that is interesting in an overview of society. Although some issues were obviously not discussed because of taboo, courtesy or censorship, most of the themes present in public discourse are recorded in the newspapers and thus accessible to us in the present. Hence, we believe newspapers are an especially good source of assessing how the role of particular discourses changed over time.

2.1 Preprocessing the data

Given the size of the data and its inherent nature, notoriously the OCR quality and the unbalanced data from different time slices, we performed a series of pre-processing steps on the data.[1]

Despite prior work (albeit on English), showing that stemming has no real advantage for likelihood and topic coherence and can actually degrade topic stability [30], we follow [40, 10, 13] and use a lemmatised version of the corpus. Indeed, the work in [10] hints at the fact that Finnish, being much more inflected than English, would benefit from lemmatisation, whereas in [40, 13] the authors stem so as to reduce the huge number of token types due to OCR issues which impacts the performance of topic

[1]The more apt phrase "purposeful data modification", coined by [34], advocates that our material is not mere data that can go through a standardised "pre-processing" pipeline. Rather, the data is modified and altered only for the specific purposes of this study, and following this study's technical and scientific requirements only.



(a) Corpus size (b) Average article size (c) No. of articles

Fig. 1: Characteristics of the NLF dataset

modelling [38]. After lemmatisation, we remove tokens that occur less than 40 times in the collection, stopwords, punctuation marks and tokens with less than 3 characters. These are additional measures to further reduce the vocabulary size and mitigate the impact of OCR noise.

3 Topic Models

3.1 LDA

Topic modelling is an unsupervised method to extract topics from a collection of documents. Typically, a topic is a probability-weighted list of words that together express a theme or idea of what the topic is about. One of the most popular topic modelling methods currently in use is Latent Dirichlet Allocation (LDA), which is "a generative probabilistic model for collections of discrete data such as text corpora" [3]. It has been extensively used in the digital humanities to extract certain themes from a collection of texts [4]. In this model, a document is a mixture of topics and a topic is a probability distribution over a vocabulary. A limitation of LDA for historical research, in its vanilla form, is that it does not account for the temporal aspect of the data: every document in the collection is "considered synchronic", as time is simply not a variable in the model. Many document collections such as news archives, however, are diachronic—the documents are from different points in time, and scholars wish to study the evolution of topics.

There are different ways to overcome this limitation. One possibility is to split the data into time slices and train LDA separately on each slice. However, in this case LDA models for each slice would be independent of each other and there is no straightforward approach of matching topics from independent models trained on disjoint data. Another possibility, which we explore in this paper, is to train a single model for a subset of the whole data set over the entire time period and then use *topic prominence* as proxy for the dynamics of discourses over time.

To do this, we compute the prominence of a topic in a given year by summing up the topic contribution for each document in that year and then normalise this number by the sum of all topic contributions from all topics for that year, as in Equation 1.

$$P(z_k|y) = \frac{\sum_{j=1}^{|D_y|} P(z_k|d_j)}{\sum_{i=1}^{T} \sum_{j=1}^{|D_y|} P(z_i|d_j)} \quad (1)$$

where y is a year in the dataset, k is a topic index, D_y is the number of documents in year y, d_j is the j^{th} document in year y and T is the number of topics in the model.

The large size of the collection and its unbalanced nature is a problem for training topic models. It is computationally expensive to train a model with millions of articles and the resulting model would be heavily biased towards the latter years of newspaper collection because it has far more data. To overcome these issues, we sampled the collection such that we have a roughly similar data size for each year of the collection and as a result, we also get a vastly reduced dataset. However, to have a model of discourse dynamics that reflects the collection more closely, we compute topic prominence using the entire collection and not just the sampled portion. We do this by inferring the topic proportions of all the documents in the collection and using these inferred distributions to compute topic prominence.

3.2 DTM

As mentioned above, there are topic models that explicitly take into account the temporal dynamics of the data. One such model is the dynamic topic model (DTM). DTM is an extension of LDA that is designed to capture dynamic co-occurence patterns in diachronic data. In this model, the document collection is divided into discrete time slices and the model learns topics in each time slice with a contribution from the previous time slice. This results in topics that evolve slightly–words changing in saliency in relation to a topic–from one time step to the next.

However, DTM also has its own limitations. It is based on an assumption that each topic should be to some extent present in each time slice, which is not always the case with real-world data such as news archives where events and themes can sometimes disappear and then re-appear at some point in the future.

Perhaps more importantly for historical research, a weakness of DTM lies in its design: to accomplish alignment across time the topic model is fit across the whole vocabulary and thus smoothing between time slices is applied. As a result, events end up being "spread out" before and after they are known to happen. This problem only becomes evident after a thorough analysis: similar models in different fields such as lexical semantic change present the same issue – the dynamic topic model SCAN [7] generates a "plane" top word for the year 1700 (two centuries ahead of the Wright Flyer, and well before the word's first attested sense of "aeroplane"), while similar model GASC [26, 23] encounters the same weakness when modelling Ancient Greek. There is unfortunately no easy way to bypass this obstacle, which is particularly problematic when studying historical themes.

For both the LDA and DTM models, we use the Gensim implementation [28] with default model hyperparameters.

4 Related Work

Topic models are widely used in the digital humanities and social sciences to draw insights from large-scale collections [4] ranging from newspaper archives to academic journals. In this section, which we do not claim to be exhaustive, we discuss some of the previous works that aimed to capture historical trends in large data collections or used such collections to study discourses using topic models. All in all, these examples highlight that there is a need to discuss how topic models can be used to capture discursive change.

In [24] the authors use Latent Semantic Analysis, another topic modelling method, to study historical trends in eighteenth-century colonial America with articles from the *Pennsylvania Gazette*. Their work also used topic prominence to show, for instance, an increased interest in political issues as the country was heading towards revolution. The authors of [40] fit several topic models on Texan newspapers from 1829 to 2008. To discover interesting historical trends, the authors slice their data into four time bins, each corresponding to historically relevant periods. Such a slicing is also carried out in [9], where the author fits LDA models on Dutch-language Belgian socialist newspapers for three time slices that are historically relevant to the evolution of workers rights, with the aim of generating candidates for lexical semantic change.

Topic modelling has also been used in discourse analysis of newspaper data. In [37] the authors applied LDA to a selection of Italian ethnic newspapers published in the United States from 1898 to 1920 to examine the changing discourse around the Italian immigrant community, as told by the immigrants themselves, over time. They proposed a methodology combining topic modelling with close reading called discourse-driven topic modelling (DDTM). Another study examined anti-modern discourse in Europe from a collection of French-language newspapers [5]. In this case, however, the authors primarily use LDA as a tool to construct a sub-corpus of relevant articles that was then used for further analysis. Modernization was also an issue in the study of Indukaev [14], who uses LDA and word embeddings to study changing ideas of technology and modernization in Russian newspapers during the Medvedev and Putin presidencies.

LDA was not designed for capturing trends in diachronic data and so several methods have been developed to address this, such as DTM, Topics over Time [39, TOT], and the more recent Dynamic Embedded Topic Model [6, DETM], an extension of DTM that incorporates information from word embeddings during training. As far as we are aware, DTM and TOT have not been used for historical discourse analysis or applied to large-scale data collections. In the original papers presenting these methods, DTM was applied to 30,000 articles from the journal *Science* covering 120 years and TOT was applied to 208 State of the Union Presidential addresses covering more than 200 years. This was to demonstrate the evolution of scientific trends for the former and the localisation of significant historical events for the latter. Recently DETM was applied on a dataset of modern news articles about the COVID-19 pandemic where the authors observed differences between countries in how the pandemic and the reactions to it were framed [19].

In the mentioned cases researchers tackle the interpretative part of using topic models for humanistic research in different ways. Like Pääkkönen and Ylikoski [27] state, they toggle between some sort of topic realism, that is, using topic models to grasp

something that exists in the data, and topic instrumentalism, that is, using topic models to find something that can be further studied. Only Bunout [5] is a clear case of topic instrumentalism. All the other studies depart from some sort of realist position, and attempt to grasp policy shifts, ideas, discourses or framings of topics through topic models, but end up with correctives of some kind by highlighting the interpretative element [24, 37], by deploying formal evaluation by historians [9] or by using other quantitative methods to fine tune the results [14]. The interpretative aspect seems especially important when it comes to deciding on what researchers use the topics to study as they can reasonably relate to historical discourses, the semantics of related words, or simply ideas. How the topics are seen to represent these or, more likely, how the researchers use the topics to make an interpretation about these based on the topics, requires a strong element of interpretation [27]. Studies show that interpreters prefer to be able to go back to actual texts in order to make sense of topics [18], which is more than reasonable, but it also seems that there is a further need for researchers to understand how different topic-modelling methods represent diachronic data. Without this knowledge it is difficult to assess to which degree and for which time periods researchers need to manually assess individual documents.

5 Use Cases

What a discourse is, has been heavily theorised within the different strands of discourse analysis [1], but the advent of digital methods that can handle large textual data sets require quite some adjustment of discourse analysis as we know it. Like this article, others have turned to topic models to grasp changes in discourse [37, 5], but this article seeks specifically to discuss the interpretation that is required when we use topic models to study discourse dynamics. The probabilistic topic models set clear boundaries between topics and in doing so might merge or separate things that historians might regard as coherent topics. However, where the probabilistic model enforces boundaries, human interpretation in general is very bad at setting those boundaries and usually just identifies the core of a discourse or topic, but cannot say where it ends.

To get at the tension between topics and discourses, we approached the material without a predefined idea about which topics we wanted to study in order to keep the study as data-driven as possible. Our interest was to use topic modelling to capture topics that could in a meaningful way be related to societal discourses, that is themes that cannot be narrowed down to individual words, but still are reasonably coherent and form at least loose topics. To this end, we trained topic models with $k \in \{30; 50\}$, inferred topic distributions for the whole collection and inspected models by carefully going through the top words in each topic and using PyLDAVis[2] [31] to study overlap between topics and salience of terms per topic in LDA and heatmap visualizations for DTM. All topics were annotated and evaluated from the point of view of historical interpretation. We then opted to use the 50-topic model to study discourse changes over time. As is common, a portion of the topics seemed incoherent or were clearly the result of the layout in newspapers (e.g. boilerplate articles about prices etc.) and

[2] `https://github.com/bmabey/pyLDAvis`

did not produce interesting information about societal discourses. Further, some of the topics clearly overlap, so that a cluster of 2-5 topics can reasonably be seen as related to a particular societal discourse. The advantage of choosing 50 topics over 30 lies precisely in the possibility of merging topics later on in interpretation, while splitting them is more difficult.

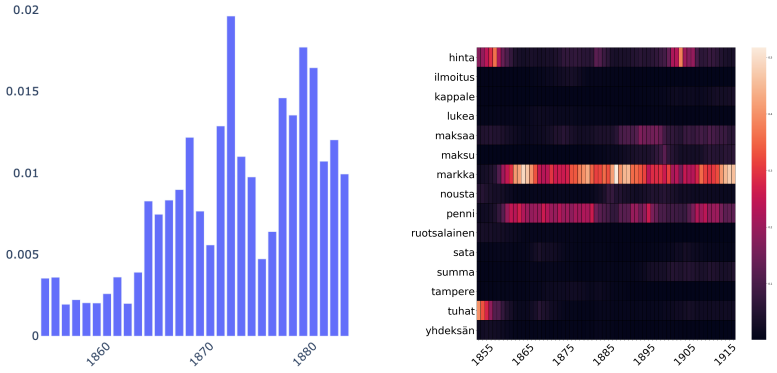
To discuss the benefits of LDA and DTM, we chose to focus on two specific themes, the discourse relating to religion and religious offices, and education. They are both rather neatly identifiable in the data, but display different trends. The former is in decline over the period of interest, whereas the latter increases in topic prominence. They can also be related to large scale processes in Finland, religious discourse to the secularization of society and education to the modernization of civic engagement.

5.1 DTM and Stretching of Topics

The two topic modelling methods perform in somewhat different ways. As mentioned, DTM is designed to incorporate temporal change in the topics, which means it includes a stronger sense of continuity in its representations of data. Whether or not this is desirable, depends on the research question, but our contention is that for studies interested in discursive change, this is either a problem or at least it is something that needs to be factored in making the historical interpretation. If we want to understand when certain discourses became dominant, declined, or even disappeared, this type of stretching cannot be allowed.

An exceptionally illustrative example of stretching among our fifty topics, is an introduction of the Finnish mark as a currency (Fig. 2a). With top words such as “mark”, “penny”, “price”, “thousand”, “pay” etc. the topic comes across as one with high internal coherence. We also see that the topic grows in prominence over time, from being relatively modest in the 1850s to gradually increased prominence after 1860. This makes sense, as the mark was adopted as currency in the year 1860 and after that self-evidently figured in public discourse. However, when we look at a heatmap visualization of the topic (Fig. 2b), we see how the topic stretches from the period 1854–1859 to the period 1860–1917, that is, from the period before the introduction of the mark to the period it was in use. After 1860 the words “mark” and “penny” are by far the most dominant terms in the topic, but for the period before 1860, the dominant terms are “price” and “thousand.” It is clear that “mark”, “penny”, “price”, and “thousand” are words that can belong to the same topic, but the heatmap representation clearly shows that the focus in the topic shifts. It is almost as if two related topics are merged as to represent one topic over the whole time period. In a situation where a historical interpretation highlights a change in past discourse, DTM produces continuity.

While there is obviously no right answer as to when one topic is stretched a bit or when different topics are simply merged together to provide a temporally continuous topic, it seems that DTM is especially problematic if one wants to study discourses that emerge or disappear in the middle of a time period studied. This means that any historical analysis using DTM requires a component of historical interpretation of not only topic coherence, but also topic coherence *over time*. Here, relying on word embeddings like in [14] can help, but this is primarily a task for evaluating the topics.



(a) Introduction of the Finnish mark in 1860 (y-axis indicates the topic probability)

(b) Heat map of terms linked to the introduction of the Finnish mark in 1860.

Fig. 2: Topic related to the introduction of the Finnish mark in 1860 (DTM). The most prominent terms in the heatmap are are "Mark" = *markka*, "penny" = *penni*, "price" = *hinta*, "thousand" = *tuhat*, "pay" = *maksu* and *maksaa*.

The speed of topic evolution can be controlled by a parameter in the DTM model. However, the 'ideal' amount of stretching is difficult to assess. For analysing discourse, this might in some cases be productive as it can point at links between nearby discourses, but is largely problematic as it hides discontinuities in the data. It becomes even problematic when dealing with material factors, like the introduction of the Finnish mark, as the stretching effect is likely to produce anachronistic representations, that is, placing something in the wrong period of time. Dealing with anachronism can perhaps be seen as one of the cornerstones of the historian's profession, which makes DTM as an anachronism prone method a poor match for historical study. Avoiding anachronisms completely is impossible, most historians would agree, but knowing when to avoid them and how to communicate about anachronistic elements in historical interpretation is key to history as a discipline [33].

5.2 Religion and Secularization

Our model performed well in grasping topics that relate to religion. The initial expectation regarding the discourse dynamics was that religious topics would be in decline. We hoped that using a topic model would be a way of showing this quantitatively. Results obtained from both LDA and DTM, presented in Figures 3a and 3b respectively, harmonize with our initial hypothesis, but do so differently. The DTM and LDA outputs cannot be aligned in any other way than manual interpretation by domain experts. In doing this we simply regarded topics that included several words that denote religious practices or offices as religious. Thus, the definition of "religious" is is rather narrow, but it also seems to match the topics that emerged from our data.

In order to inspect the discourse dynamics of religious topics, we have combined several topics that related to religious themes in the LDA model, whereas in the latter, DTM model, we only chose one topic to be represented.[3]

To our knowledge, topic models have not been used to study discursive change regarding secularization. However, in line with some earlier qualitative assessments [15], we hypothesize that this decline in religious discourse entails two interrelated developments: 1) Religion did not disappear from public discourse, but instead changed and disappeared from certain *types* of discourses. In the early nineteenth century, religion had a much more holistic presence in public discourse, meaning that religious metaphors and religious expressions and topics were used at a much vaster scale. 2) Over the course of the nineteenth century, religious topics became more focused. This means a segmentation of public discourse so that religious topics were increasingly confined to particular journals or genres.

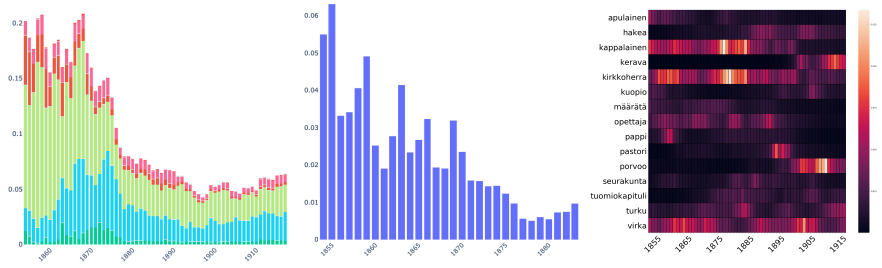
Keeping in mind the issue of stretching with DTM, we can look into the shifting saliency of words within the topic of religious offices and notice a shifting focus over time (Fig. 3c). In the early 1900s terms relating to "holding an office" and names of particular congregations become more dominant in the topic. This, again, suggests that DTM as a method does some stretching. There is a downside and an upside to this. On the one hand, the stretching distorts the topic prominence a bit by making it look like there is more continuity than in the LDA visualization. However, this may not be that crucial as the declining trends in Fig. 3a and Fig. 3b are rather similar. On the other hand, the stretching may be good for detecting conceptual links between different groups of words. In this particular case the stronger link between religious offices and some towns like Kerava and Porvoo, is probably indicative of a move of religious discourse from an overarching question to something that is more likely dealt with in conjunction to matters at local parishes. That is, religious offices were more often than before dealt with in connection to local congregations. This is in line with our above-mentioned assumption about religious discourse becoming more distinct.

5.3 Education and Modernity

While we expected religious themes to decline and become less central, we assumed there would be some themes that partly overlap with religion, but also would show an increasing trend. One example of this is the topic of education, which has historically been heavily interwoven with the church, but at the same time when basic education became available for a higher amount of people, it also became central in questioning the role of the church and religion. Education in nineteenth-century Finland was both central for ensuring conformity of the Lutheran faith, but paradoxically also was a vehicle of secularization. [8]

As in the case of religious discourse, alignment between DTM and LDA can only be made through human interpretation. It seems, that in this case DTM captures one topic

[3] We also experimented with more data-driven methods to cluster topics, including for example methods based on Jensen-Shannon Divergence. They unfortunately did not need to clusters that our domain experts would make sense of. Nonetheless, despite this, we still believe this is an interesting avenue to pursue which could help answer the common 'number of topics' question often brought up within the field.



(a) Topics related to religion on decline (LDA)

(b) Development of religious topic (chaplain, priest and office) over time

(c) Heatmap of terms linked to office of religion topic.

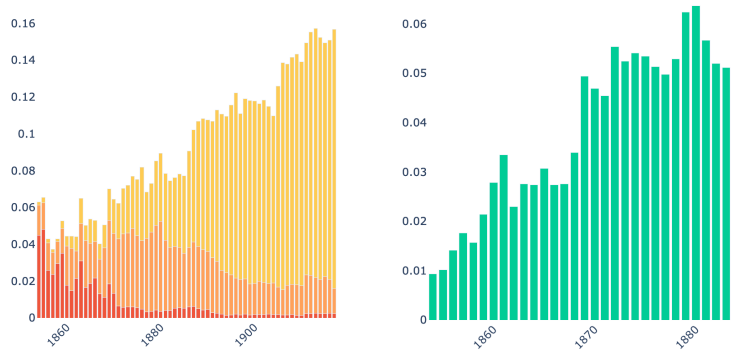
Fig. 3: Religious topics in LDA (a) and DTM (b,c); y-axis in (a, b) indicates the topics' probabilities. Most prominent terms in the heatmap are "chaplain" = *kappalainen*, "vicar" = *kirkkoherra*, "teacher" = *opettaja*, "priest" = *pappi*, "Porvoo" (a town), "parish" = *seurakunta*, "Turku" (a town), and "office" = *virka*.

that is fairly coherent, revolves around education and schooling, and is on the rise in the research period (Fig. 4b). For LDA, this is not the case, as an PyLDAVis inspection of most salient words across all fifty topics show that words like "school" and "folk school" appear mostly in three topics of which two are in decline and one heavily on the rise (Fig. 4a).

Interestingly, LDA and DTM seem to be pointing at a similar historical development. The two declining LDA topics are based on their most salient terms and are more focused on schools as buildings and institutions as well as teaching as a profession, whereas the topic on the rise includes salient vocabulary relating to, not only schools, but also meetings, civic engagements, and decision making. The DTM topic at hand shows a similar development which can be inspected in a heatmap of most salient terms over time. The terms "school", "child", and "teacher" dominate early in the period. By the end of the period the topic becomes broader, and terms like "municipality" and "meeting" have become more salient than the vocabulary relating to schools. Here the stretching of DTM creates the links that are also visible in the three LDA topics, and it shows a transformation in which educational issues are present in the whole topic, but focus shifts from concrete schools to civic engagement.

6 Conclusions

Our focus in this text has been on discourses that cannot be reduced to mere words, isolated events or particular people, but concern broader societal topics that either declined or gained in prominence. The interpretation of these topics and their contextualisation to nineteenth-century Finnish newspapers revealed clear topical cores that can be interpreted as an encouraging point of departure for further explorations based on topic models when aiming to understand Finnish public discourse through historical newspapers.



(a) Development of education topic over time (LDA)

(b) Development of education topic over time (DTM)

Fig. 4: Education topic in LDA and DTM; y-axis indicates the topics' probabilities

In this paper, we have learned that although it is difficult to pinpoint exactly where a discourse or topic ends, LDA and DTM can fairly reliably grasp many semi-coherent themes in past discourse and help us study the dynamics of discourses. However, our comparison of LDA and DTM as methods for getting at past discourse also shows that both methods require a very strong interpretative element in analysing historical discourses. DTM is much more prone to stretch or even merge topics, which requires an interpretative assessment of whether the stretching highlights interesting historical continuities or if it hides historical discontinuities that would require attention. We found that producing heatmaps of term saliency over time for each topic is a very useful way of doing this type of assessment. For LDA, stretching is not so much a problem, but often it seems interpretation is needed in seeing which topics logically relate to one another. While historical discourse analysis is traditionally tied strongly to a tradition of hermeneutic interpretation, the use of topic models to grasp discourse dynamics does not remove that need even if they allow for a quantification of discourse dynamics over time.

While we regard stretching in DTM as a predominantly negative feature, in some cases it can be useful. In the topics relating to education discussed above, the stretching in DTM actually points out links in discourses and is quite productive for the interpretative process of trying to figure out discourse dynamics. However, also in this case, the relevance of historical interpretation should be highlighted because it is very hard to tell whether the stretching of topics is an accurate reflection of the data or a shortcoming of the model. This can be addressed only by relating visualisations of topics to existing historical research and reading source texts. Humanities scholars are in general very good at making such interpretations, but it also needs to be noted that when we move further into the domain interpretative scholarship, we also lose some of the benefits of working with quantifying models. While it would be foolish to claim that a topic model represents data in a way that it provides simple facts about historical development, our use cases show that if we seek to find more reliable quantification LDA may

provide better results than DTM. Further, using LDA moves the interpretative stage further down in the research process, as it is likely to be about evaluating the connections between different topics over time. In DTM, the interpretation is likely moved forward to an evaluation of how well the algorithm did this merging topics. On this sense, our take on topic models harmonises with [27] who stress the role of humanistic interpretation, but for the sake of transparency suggest pushing the interpretation stage later in the research process.

Acknowledgements

This work has been supported by the European Union Horizon 2020 research and innovation programme under grants 770299 (NewsEye) and 825153 (EMBEDDIA). SH is funded by the project *Towards Computational Lexical Semantic Change Detection* supported by the Swedish Research Council (2019–2022; dnr 2018-01184).

References

1. Angermuller, J., Maingueneau, D., Wodak, R. (eds.): The discourse studies reader: Main currents in theory and analysis. John Benjamins Publishing, Amsterdam, the Netherlands ; Philadelphia PA (2014)
2. Blei, D.M., Lafferty, J.D.: Dynamic topic models. In: Proceedings of the 23rd international conference on Machine Learning. pp. 113–120 (2006)
3. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. Journal of Machine Learning Research **3**(Jan), 993–1022 (2003)
4. Brauer, R., Fridlund, M.: Historicizing topic models, a distant reading of topic modeling texts within historical studies. In: International Conference on Cultural Research in the context of Digital Humanities, St. Petersburg: Russian State Herzen University (2013)
5. Bunout, E.: Grasping the anti-modern discourse on Europe in the digitised press or can text mining help identify an ambiguous discourse? (2020)
6. Dieng, A.B., Ruiz, F.J., Blei, D.M.: The dynamic embedded topic model. arXiv preprint arXiv:1907.05545 (2019)
7. Frermann, L., Lapata, M.: A Bayesian model of diachronic meaning change. Transactions of the Association for Computational Linguistics **4**, 31–45 (2016)
8. Hanska, J., Vainio-Korhonen, K. (eds.): Huoneentaulun maailma: kasvatus ja koulutus Suomessa keskiajalta 1860-luvulle. Suomalaisen Kirjallisuuden Seuran toimituksia, 1266:1, Suomalaisen kirjallisuuden seura, Helsinki (2010), publication Title: Huoneentaulun maailma : kasvatus ja koulutus Suomessa keskiajalta 1860-luvulle
9. Hengchen, S.: When Does it Mean? Detecting Semantic Change in Historical Texts. Ph.D. thesis, Université libre de Bruxelles (2017)
10. Hengchen, S., Kanner, A.O., Marjanen, J.P., Mäkelä, E.: Comparing topic model stability between Finnish, Swedish, English and French. In: Digital Humanities in the Nordic Countries (2018)
11. Hengchen, S., Ros, R., Marjanen, J.: A data-driven approach to the changing vocabulary of the nation in English, Dutch, Swedish and Finnish newspapers, 1750-1950. In: Proceedings of the Digital Humanities (DH) conference (2019)
12. Hengchen, S., Ros, R., Marjanen, J., Tolonen, M.: A data-driven approach to studying changing vocabularies in historical newspaper collections. Digital Scholarship in the Humanities (2021)

13. Hill, M.J., Hengchen, S.: Quantifying the impact of dirty OCR on historical text analysis: Eighteenth century collections online as a case study. Digital Scholarship in the Humanities **34**(4), 825–843 (2019)
14. Indukaev, A.: Studying Ideational Change in Russian Politics with Topic Models and Word Embeddings. In: Gritsenko, D., Wijermars, M., Kopotev, M. (eds.) Palgrave Handbook of Digital Russia Studies. Palgrave Macmillan, Basingstoke (2021)
15. Juva, M.: Valtiokirkosta kansankirkoksi: Suomen kirkon vastaus kahdeksankymmentäluvun haasteeseen. WSOY, Porvoo (1960)
16. Kokko, H.: Suomenkielisen julkisuuden nousu 1850-luvulla ja sen yhteiskunnallinen merkitys. Historiallinen Aikakauskirja **117**(1), 5–21 (2019)
17. La Mela, M., Tamper, M., Kettunen, K.: Finding Nineteenth-century Berry Spots: Recognizing and Linking Place Names in a Historical Newspaper Berry-picking Corpus. In: Navarretta, C., Agirrezabal, M., Maegaard, B. (eds.) DHN 2019 - Digital Humanities in the Nordic Countries. pp. 295–307. CEUR Workshop Proceedings, CEUR (2019), https://cst.dk/DHN2019/DHN2019.html
18. Lee, T.Y., Smith, A., Seppi, K., Elmqvist, N., Boyd-Graber, J., Findlater, L.: The human touch: How non-expert users perceive, interpret, and fix topic models. International Journal of Human-Computer Studies **105**, 28–42 (Sep 2017). https://doi.org/10.1016/j.ijhcs.2017.03.007, https://linkinghub.elsevier.com/retrieve/pii/S1071581917300472
19. Li, Y., Nair, P., Wen, Z., Chafi, I., Okhmatovskaia, A., Powell, G., Shen, Y., Buckeridge, D.: Global surveillance of covid-19 by mining news media using a multi-source dynamic embedded topic model. In: Proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics. pp. 1–14 (2020)
20. Mäkelä, E., Tolonen, M., Marjanen, J., Kanner, A., Vaara, V., Lahti, L.: Interdisciplinary collaboration in studying newspaper materiality. In: Krauwer , S., Fišer, D. (eds.) Twin Talks Workshop at DHN 2019. pp. 55–66. CEUR Workshop Proceedings, CEUR-WS.org, Germany (2019)
21. Marjanen, J., Pivovarova, L., Zosa, E., Kurunmäki, J.: Clustering ideological terms in historical newspaper data with diachronic word embeddings. In: 5th International Workshop on Computational History, HistoInformatics 2019. CEUR-WS (2019)
22. Marjanen, J., Vaara, V., Kanner, A., Roivainen, H., Mäkelä, E., Lahti, L., Tolonen, M.: A national public sphere? Analyzing the language, location, and form of newspapers in Finland, 1771–1917. Journal of European Periodical Studies **4**(1), 54–77 (2019)
23. McGillivray, B., Hengchen, S., Lähteenoja, V., Palma, M., Vatri, A.: A computational approach to lexical polysemy in Ancient Greek. Digital Scholarship in the Humanities **34**(4), 893–907 (2019)
24. Newman, D.J., Block, S.: Probabilistic topic decomposition of an eighteenth-century american newspaper. Journal of the American Society for Information Science and Technology **57**(6), 753–767 (2006)
25. Oiva, M., Nivala, A., Salmi, H., Latva, O., Jalava, M., Keck, J., Domínguez, L.M., Parker, J.: Spreading News in 1904: The Media Coverage of Nikolay Bobrikov's Shooting. Media History **26**(4), 391–407 (Oct 2020). https://doi.org/10.1080/13688804.2019.1652090, https://www.tandfonline.com/doi/full/10.1080/13688804.2019.1652090
26. Perrone, V., Palma, M., Hengchen, S., Vatri, A., Smith, J.Q., McGillivray, B.: GASC: Genre-aware semantic change for Ancient Greek. In: Proceedings of the 1st International Workshop on Computational Approaches to Historical Language Change. pp. 56–66. Association for Computational Linguistics, Florence, Italy (Aug 2019). https://doi.org/10.18653/v1/W19-4707, https://www.aclweb.org/anthology/W19-4707

27. Pääkkönen, J., Ylikoski, P.: Humanistic interpretation and machine learning. Synthese (Sep 2020). https://doi.org/10.1007/s11229-020-02806-w, http://link.springer.com/10.1007/s11229-020-02806-w
28. Řehůřek, R., Sojka, P.: Software Framework for Topic Modelling with Large Corpora. In: Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks. pp. 45–50. ELRA, Valletta, Malta (May 2010), http://is.muni.cz/publication/884893/en
29. Salmi, H., Paju, P., Rantala, H., Nivala, A., Vesanto, A., Ginter, F.: The reuse of texts in Finnish newspapers and journals, 1771–1920: A digital humanities perspective. Historical Methods: A Journal of Quantitative and Interdisciplinary History pp. 1–15 (Sep 2020). https://doi.org/10.1080/01615440.2020.1803166, https://www.tandfonline.com/doi/full/10.1080/01615440.2020.1803166
30. Schofield, A., Mimno, D.: Comparing apples to apple: The effects of stemmers on topic models. Transactions of the Association for Computational Linguistics **4**, 287–300 (2016)
31. Sievert, C., Shirley, K.: Ldavis: A method for visualizing and interpreting topics. In: Proceedings of the workshop on interactive language learning, visualization, and interfaces. pp. 63–70 (2014)
32. Sorvali, S.: "Pyydän nöyrimmästi sijaa seuraavalle" – Yleisönosaston synty, vakiintuminen ja merkitys autonomian ajan Suomen lehdistössä. Historiallinen Aikakauskirja **118**(3), 324–339 (2020)
33. Syrjämäki, S.: Sins of a historian: Perspectives on the problem of anachronism. Ph.D. thesis, Tampere University Press, Tampere (2011), oCLC: 816367378
34. Thompson, L., Mimno, D.: Authorless topic models: Biasing models away from known structure. In: Proceedings of the 27th International Conference on Computational Linguistics. pp. 3903–3914. Association for Computational Linguistics, Santa Fe, New Mexico, USA (Aug 2018), https://www.aclweb.org/anthology/C18-1329
35. Tommila, P., Landgrén, L.F., Leino-Kaukiainen, P.: Suomen lehdistön historia 1. Sanomalehdistön vaiheet vuoteen 1905. Kustannuskiila, Kuopio (1988)
36. Vesanto, A., Nivala, A., Rantala, H., Salakoski, T., Salmi, H., Ginter, F.: Applying BLAST to text reuse detection in finnish newspapers and journals, 1771-1910. In: Proceedings of the NoDaLiDa 2017 Workshop on Processing Historical Language. pp. 54–58 (2017)
37. Viola, L., Verheul, J.: Mining ethnicity: Discourse-driven topic modelling of immigrant discourses in the USA, 1898–1920. Digital Scholarship in the Humanities (2019)
38. Walker, D., Lund, W.B., Ringger, E.: Evaluating models of latent document semantics in the presence of ocr errors. In: Proceedings of the 2010 conference on empirical methods in natural language processing. pp. 240–250 (2010)
39. Wang, X., McCallum, A.: Topics over time: a non-markov continuous-time model of topical trends. In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 424–433 (2006)
40. Yang, T.I., Torget, A., Mihalcea, R.: Topic modeling on historical newspapers. In: Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities. pp. 96–104 (2011)

B. Manuscript: Embedding-Based Topic Models are Robust to OCR Noise in Text

Evaluating the Robustness of Embedding-based Topic Models to OCR Noise

First Author[1 [0000−1111−2222−3333]], Second Author[2,3 [1111−2222−3333−4444]], and Third Author[3 [2222−−3333−4444−5555]]

Abstract. Unsupervised topic models such as Latent Dirichlet Allocation (LDA) are popular tools to analyse digitised corpora. However, these tools have been shown to degrade with OCR noise. Topic models that incorporate word embeddings during inference have been proposed to address the limitations of LDA, but these models have not seen much use in historical text analysis. In this paper we explore the impact of OCR noise on two embedding-based models, Gaussian LDA and the Embedded Topic Model (ETM) and compare their performance to LDA. Our results show that these models, especially ETM, are more resilient than LDA in the presence of noise in terms of topic quality and classification accuracy.

Keywords: Topic modelling · Word embeddings · OCR noise

1 Introduction

Large-scale collections of historical documents are becoming more accessible to researchers due to the efforts made to digitise these materials. Digitization pipelines commonly involve passing the material through an optical character recognition (OCR) engine which outputs text data that can then be used for downstream tasks. Due to various factors such as the printing quality of the original material, font, and layout styles, the output of OCR engines varies in quality. OCR errors stemming from this process can have a significant impact when downstream natural language processing (NLP) tools are used to analyse this data, such as to discover topics from the data.

Topic modelling is a method to extract latent topics in a collection of documents. It is a popular approach in Digital Humanities and data-driven historical research to analyse large historical collections such as newspaper archives [22, 20, 11], academic journals [13] and handwritten diaries [3]. Probabilistic topic models such as the Latent Dirichlet Allocation [2] model a topic as a distribution over a vocabulary and a document as a mixture of topics. Prior research quantifying the impact of OCR noise on topic modelling shows that the topics and topic mixtures deteriorate in quality as the level of noise increases [21, 14].

Word embeddings are distributed representations of words in a dense vector space that encode their usage in a corpus [12, 16]. They can capture both syntactic and semantic attributes of words such that words that typically occur in similar contexts are in close proximity to each other in the embedding space. Approaches that combine topic modelling with word embeddings to improve the semantic coherence of topics and address the challenge of scaling topic models to large vocabularies include Gaussian LDA (GLDA) [5], spherical Hierarchical Dirichlet Process (sHDP) [1], the Latent Concept

Topic Model [9], and the Embedded Topic Model (ETM) [6]. GLDA and ETM are LDA-like models that use word embeddings and have shown improved topic quality over LDA on clean datasets.

Classical topic models like LDA use word co-occurence patterns to discover latent topics in a corpus and the negative impact of OCR noise on topic modelling is due to the distortion of the word distribution when words are misspelled [21]. In embedding-based models, word identities are replaced with word *embeddings* that, in principle, can be more resilient to OCR noise, provided misspellings of the same word cluster together in the embedding space. There is, however, no existing work that investigates the robustness of these models on data with OCR noise and whether they show any improvement over LDA.

In this paper we conduct a quantitative assessment of the performance of two embedding-based models, GLDA and ETM, on datasets with OCR noise. Our aim is to test whether embedding-based models can be used to improve the analysis of digitised historical documents. In Section 2 we give an overview of the models used in this work and previous evaluations on the impact of OCR noise on topic modelling. In Section 3 we describe our experimental setup and evaluation measures. In Section 4 we present and analyse our experimental results, showing a clear benefit of using embeddings in this context, especially when they are trained on noisy data.

2 Related Work

Latent Dirichlet Allocation (LDA) [2] is a probabilistic topic modelling method for extracting topics from text corpora. It models a topic as a probability distribution over a fixed vocabulary of the given corpus and a document as a mixture of topics. LDA relies on the co-occurrence of the words in the documents to infer the latent topics and topic mixtures of the documents. LDA is widely used in digital humanities and social sciences to study large-scale collections ranging from newspaper archives to academic journals [4].

Models that use word embeddings have been proposed to improve topic quality and handle out-of-vocabulary words. Gaussian LDA (GLDA) [5] is the first LDA-based topic model that directly incorporates word embeddings during topic inference. Instead of treating topics as categorical distributions over the vocabulary, GLDA characterizes topics as multivariate Gaussian distributions over the word embedding space whose mean and variance are estimated during inference using a Gibbs sampler. Words are ranked according to their probability density under the posterior-predictive distribution given the training corpus.

In the Embedded Topic Model (ETM) [6], words are generated from a categorical distribution whose natural parameter is the inner product of the word embeddings associated with a topic. Topics and words share the same embedding space which means that a topic is a point in the embedding space called a topic embedding and the most probable words in the topic are those with embeddings that are close to the topic embedding.

Various studies have evaluated the impact of OCR errors on unsupervised topic modelling. A comparative study of document clustering and LDA on OCR-ed text

indicated that OCR noise had a greater performance impact on topic modelling than on document clustering [21]. Another evaluation revealed that while OCR noise resulted in lower topic coherence, it had little impact on model stability [14]. A more general study on the impact of noisy OCR on historical analysis using a large corpus of eighteenth-century texts found that topics extracted by a Structural Topic Model [18]–an LDA-based model that incorporates document structure and metadata information in the priors–from OCR-ed texts aligned well with topics from the gold standard texts although they hinted that the topic model had trouble with poetry-adjacent topics [8]. These previous evaluations, however, focused on well-established topic models based on word co-occurrence and as far as we are aware embedding-based models have not been tested to analyse OCRed data.

3 Methodology

Following walker2010evaluating, we first evaluate the topic models on a corpus of historical documents with real OCR noise that have aligned gold standard (GS) texts. Then we evaluate the models on a larger corpus where synthetic noise has been introduced at increasing levels. Generating synthetic noisy data allows us to control the level of noise to measure its impact in experiments.

3.1 Datasets

Real noise The Overproof dataset [7] consists of 30,301 digitised news articles from the Sydney Morning Herald 1842–1954, from the archives of the National Library of Australia.[1] The articles were processed using the ABBYY FineReader OCR tool and additional corrections were done using crowd-sourced annotations. The OCR-ed articles have a word error rate (WER) of 25% [15]. The OCR and GS articles are aligned on a character level.

Synthetic noise To generate data with synthetic noise, we start with a clean dataset and gradually corrupt the data by introducing noise at increasing levels. For our clean data, we use the Reuters RCV1 dataset, consisting of over 800K English newswire articles with pre-assigned categorical labels [10]. We use a reduced dataset of 50K articles sampled from the largest categories.

We follow the procedure of walker2010evaluating to generate synthetic noise based on a noise model constructed from a dataset with real noise. To build a noise model, we construct a contingency matrix $\mathbf{M}$ where $\mathbf{M}_{x,y}$ is the number of times character x in a GS article is confused with character y in the corresponding OCR article. We then normalise these counts by row to get a distribution $p(y|x)$.

To generate parameterised noise, we interpolate the matrix $\mathbf{M}$ such that $\mathbf{M}_\gamma = \gamma\mathbf{M} + (1-\gamma)\mathbf{I}$ where $\mathbf{I}$ is the identity and γ is the interpolation parameter. This means that for $\gamma = 0$, no noise is introduced, while at $\gamma = 1.0$, the interpolated matrix is equivalent to $\mathbf{M}$. We generated corrupted datasets from the Reuters corpus with γ ranging from 0 to 1 in increments of 0.2. This resulted in datasets with CERs of 0%, 7%, 14%, 21%, 28% and 35%. Table 1 summarizes the datasets used in our experiments.

[1] http://overproof.projectcomputing.com/datasets/

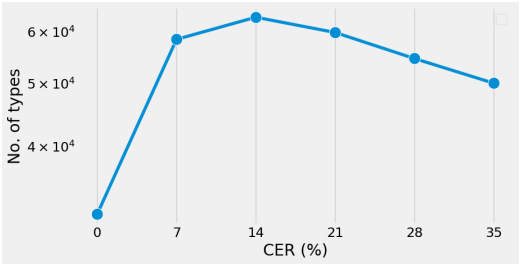


Fig. 1: Vocabulary size of the synthetic Reuters data after vocabulary reduction.

Vocabulary reduction We reduced the vocabularies using simple frequency cutoffs to reduce model training times. For the Overproof dataset, we used a document frequency cutoff of 10 (terms that occur in fewer than 10 documents are excluded), leading to a vocabulary size of 37,699 for the OCR articles and 27,174 for the GS articles. For the corrupted Reuters dataset, we also set the document frequency cutoff to 10. Figure 1 shows the resulting vocabulary sizes at each noise setting.

	#types	#tokens	#art.
Overproof OCR	1.3M	10M	30K
Overproof GS	414K	9.8M	30K
Reuters sampled	414K	12.4M	50K

Table 1: Datasets used in the experiments.

3.2 Model training and word embeddings

We use LDA as our baseline model. We trained LDA models using the Gensim library, which uses variational inference to infer topics [17]. For ETM, we used the authors' implementation[2]. For LDA and ETM, we ran inference for 1000 epochs with default hyperparameters. For GLDA, we used the `gaussianlda` Python package, which implements the algorithm using Gibbs sampling with Cholesky decomposition and alias sampling to reduce sampling complexity [5][3]. We ran the sampler for 20 iterations, based on initial experiments with the clean *20-Newsgroups* dataset.

In our experiments with real noise data, we experimented with two different types of word embeddings: (1) pre-trained GloVe embeddings trained on English Wikipedia and Gigaword [16][4]; and (2) Word2Vec embeddings [12] trained on the Overproof data using Gensim (separate embeddings for the OCR and GS portions of the data). This is to investigate whether word embeddings trained on a large amount of clean data would result in better topic models than embeddings trained on more limited and noisier data.

[2] https://github.com/adjidieng/ETM
[3] https://pypi.org/project/gaussianlda/
[4] https://nlp.stanford.edu/projects/glove/

On the experiments with synthetic data, we used Word2Vec embeddings trained on the corrupted Reuters data. We trained separate sets of embeddings for each noise setting.

We trained topic models with 50 topics on the OCR and GS portions of the Overproof data and 100 topics for each noise setting of the synthetic Reuters data. To account for the randomness inherent in the models we repeated each experiment ten times and report the averaged results.

3.3 Evaluation measures

We evaluate topic models using a variety of quantitative measures that account for different aspects of their usefulness for historical text analysis.

Topic coherence Topic coherence quantifies the interpretability of a topic as represented by its most probable terms. Coherence measures based on pointwise mutual information (PMI) of word pairs, such that words that tend to appear together in the same documents have better scores, have been found to correlate well with human judgement. We use the C_v coherence measure proposed by roder2015exploring and implemented in the Gensim package [5]. Coherence is measured with respect to a corpus, which can be the training corpus or an external corpus such as Wikipedia. Although Wikipedia is widely used as a reference corpus to measure coherence [19], it is not suitable in this case, because it will unfairly penalise words with OCR errors that do not appear in Wikipedia. Thus, we measure coherence only with respect to the training corpus.

Diversity of topics Models that learn a wide variety of topics are preferable to models with redundant topics. We measure topic diversity as the proportion of unique words out of all the top words representing all the topics in the model [6]. The impact of OCR noise on topic diversity has also not been quantified before and so we measure it in this study. For topic coherence and diversity we evaluate on the top 20 terms of each topic.

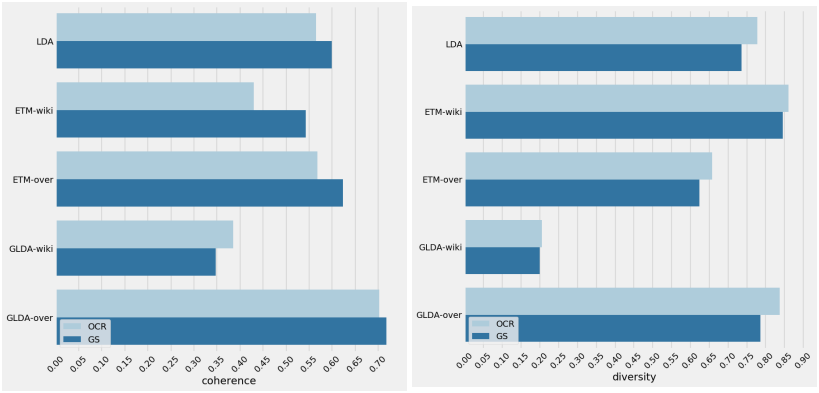
Classification accuracy We evaluate the quality of the per-document topic proportions inferred by the models through a supervised document classification task. We train a classifier on a portion of the data using the inferred topic proportions as features and pre-assigned categories as labels, then test the classifier on unseen documents [21]. As this evaluation requires GS labels, we only conduct this on the Reuters dataset with synthetic noise. We used a logistic regression classifier with ten-fold cross-validation in our evaluation.

4 Results and Discussion

4.1 Performance on Real Noise

Figure 2 shows the results of our experiments with real noise data. In terms of topic coherence, almost all the models perform better on the GS documents than the OCR

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



(a) Topic coherence (b) Topic diversity

Fig. 2: Performance on the Overproof dataset averaged over 10 runs. *wiki* models use word embeddings trained on Wikipedia while *over* models use embeddings trained on the Overproof data. Higher values are better.

documents, as would be expected (Figure 2a). Only GLDA with Wikipedia-trained embeddings performs better on OCR documents than the GS (0.39 for the former and 0.35 for the latter). On the other hand, GLDA with Overproof embeddings shows the best topic coherence for both OCR and GS (0.70 for OCR and 0.72 for GS).

ETM with Overproof embeddings has similar topic coherences to LDA–both have a coherence of 0.57 for OCR while for GS, ETM is only a little better with a mean coherence of 0.62 and LDA has 0.6. ETM with Wikipedia-trained embeddings performs worse than LDA, and is in fact the second-lowest performing model after GLDA with Wikipedia embeddings, with a coherence of 0.43 for OCR and 0.54 for GS–a difference of almost 10 percentage points.

These results indicate that for embedding-based topic models, it is preferable to use embeddings trained on the target corpus rather than on a broad-domain dataset like Wikipedia, despite the latter being larger in size and cleaner especially when the target corpus is a specialized document collection, such as historical documents. One reason for this could be that Wikipedia is a contemporary dataset written for a contemporary audience while the Overproof corpus is made up of articles from the mid-nineteenth to the mid-twentieth century. With a corpus from an earlier time period than Overproof or from a very different genre (e.g. novels and poetry) we predict that using embedding-based models with embeddings trained from standard datasets such as Wikipedia would perform even worse than what we see here.

Now we take a closer look at the characteristics of the topics produced by one run of each of the models (Table 2). We focus on ETM and GLDA with Overproof embeddings. We see that the most coherent ETM and LDA topics appear to be more coherent than the GLDA topics despite GLDA having the best mean topic coherence overall (mean topic coherences for OCR documents are 0.57, 0.57, and 0.70 for LDA, ETM, and GLDA, respectively while for GS, it is 0.60, 0.62, and 0.72). GLDA is known

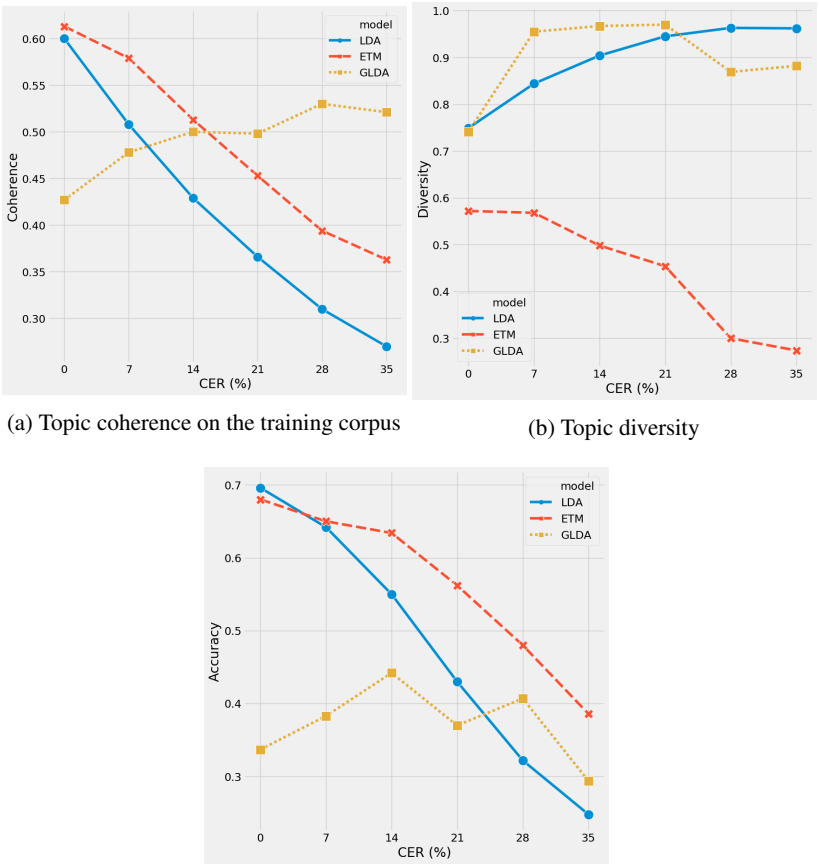
No.	*Words*	*Coherence*
LDA-OCR		
33	petitioner, respondent, nisi, decree, honor, formerly, appeared, ground, marriage, granted	0.95
8	club, match, team, cricket, played, play, runs, first, association, matches	0.83
7	john, william, james, thomas, george, charles, henry, following, robert, joseph	0.80
LDA-GS		
11	petitioner, marriage, decree, respondent, formerly, nisi, appeared, married, ground, granted	0.95
9	accused, prisoner, charged, guilty, charge, court, trial, stealing, months, sessions	0.82
40	fined, police, court, charged, days, costs, one, 10s, two, pay	0.79
ETM-OCR		
31	respondent, petitione, nisi, appeared, honor, formerly, decree, ground, issue, foi	0.91
9	charged, court, fined, john, police, prisoner, two, sentenced, months, guilty	0.81
50	john, william, james, thomas, henry, george, charles, pte, joseph, edward	0.80
ETM-GS		
21	petitioner, marriage, appeared, formerly, respondent, decree, ground, nisi, married, granted	0.95
41	match, cricket, team, played, wickets, runs, play, second, first, club	0.88
33	john, william, george, charles, james, henry, thomas, frederick, edward, arthur	0.84
GLDA-OCR		
12	managers, woiking, administrator, guidance, servlco, goneral, publicity, lenders, bown	0.73
38	accompanying, pipers, recoived, governors, alio, transmitted, photographs, btato, lag	0.73
24	labt, revived, tuna, succeeding, ast, thief, riot, casualty, lator, houbo	0.72
GLDA-GS		
47	parent, outset, sult, cardiff, terror, dawn, tha, alley, biggest, sweepin	0.72
1	discontinued, livered, forcibly, blacksmith, extracted, interrupted, reopened, sampson, tempted	0.72
42	curiosity, prominence, sult, repetition, notion, strangers, tha, birmingham, ity, lame	0.71

Table 2: Most coherent topics from LDA, ETM, and Gaussian LDA on the Overproof dataset.

to produce qualitatively different topics from LDA [5] and we notice that it also produces different topics from ETM. Another difference is that topics produced by ETM on the OCR documents show a high degree of correspondence with topics from the GS data, while the same cannot not be said of the GLDA topics. For instance, Topic 31 of ETM-OCR and Topic 21 of ETM-GS are topics on legal matters and show many overlapping terms (they share 17 of their top 20 terms), and Topic 50 of ETM-OCR and Topic 33 of ETM-GS are topics on first names (they overlap on 15 out of 20 terms). Further similar correspondences can be found by manual inspection (see Table 2). This correlation of topics between the clean and noisy data was also observed for LDA by walker2010evaluating, although they did not quantify the correlation.

OCR topics are more diverse than GS topics for all models (Figure 2b). We suggest this can be attributed primarily to the higher vocabulary size of the OCR documents. While the training data used for word embeddings has a high impact on the coherence of the embedding-based models, it does not seem have a significant influence on topic diversity. ETM with Wikipedia embeddings has the most diverse topics (0.86 for the OCR portion and 0.85 for GS) while GLDA with the same Wikipedia embeddings has the most redundant topics (0.21 for OCR and 0.20 for GS).

4.2 Performance on synthetic noise



(a) Topic coherence on the training corpus

(b) Topic diversity

(c) Mean classification accuracy

Fig. 3: Performance on the Reuters data with synthetic noise averaged across 10 runs. Higher values are better.

Figure 3 shows our experimental results on the synthetic data. Figure 3a indicates that both ETM and LDA degrade linearly in coherence as noise increases, though the former degrades more slowly than the latter. At a CER of 0% (no noise), coherences for both models are quite similar (0.60 for LDA and 0.61 for ETM), while at the highest noise setting (35% CER), LDA coherence is 0.27 while ETM is 0.36, a difference of almost ten percentage points.

Interestingly, GLDA improves in coherence as noise increases, it starts at 0.43 for 0% CER and at 35% CER, it is at 0.52. GLDA topics have the tendency to cluster misspelled

words together, an effect of the nature of GLDA topics which are unimodal distributions in the embedding space. For instance at 7% CER, the top terms of one topic are 'dollat', 'lollar', 'dollan', 'doluar', all misspellings of the same word. At the same noise level, the 'dollar topic' of ETM has the top terms 'dollar', 'bank', 'dealers', 'rates'.

In terms of topic diversity, our results show that ETM produces more homogeneous topics than LDA or GLDA at all noise levels (Figure 3b), corroborating our results in the real noise data (Figure 2b). As noise increases, ETM topics become even less diverse (at 35% CER, diversity is at 0.27, 0.88, and 0.96 for ETM, GLDA and LDA, respectively). It is surprising therefore to find that even though ETM has the lowest topic diversity, it performs better than LDA and GLDA in the document classification task (Figure 3c). We would suppose that homogeneous topics translate to homogeneous features and that classifiers trained with such features would not do a very job of differentiating between documents from different categories.

To investigate further, we look at which topics the classifier deemed the most useful for predicting the the article labels by checking the weights assigned by the logistic regression classifier to each feature/topic. This tells us which topics are most indicative of which classes and how those topics change with a change in the amount of noise in the data.

In Table 3, we show the top weighted topics for two article classes: GCAT (Government/Social) and M14 (Commodity Markets). For GCAT articles, the top topics for LDA and ETM are about government finances and law enforcement in the dataset without noise injected. When the CER is at 21%, we still see similar topics on law enforcement and politics for both models. For M14 articles with no noise, top LDA topics include topics on interest rates and court proceedings. ETM, however, has only a single topic that associated with this class (all other topics have a coefficient of zero). When noise is added (CER of 21%), both LDA and ETM now have only one topic associated with M14 articles. This tells us that as noise increases we get more topics that are not useful for document representation.

5 Conclusions

In this paper we assessed the impact of OCR noise on two embedding-based topic models, Gaussian LDA and ETM, on datasets with real noise and synthetic noise, with LDA as our baseline. We evaluated the models on the following measures: coherence, diversity, and classification accuracy. We also experimented with different word embeddings for GLDA and ETM.

We found that using embeddings trained on the same data as the topic models produces more coherent topics than embeddings trained on Wikipedia although the latter is a larger and cleaner dataset. We reasoned that this is due to the difference in time periods between Wikipedia and the Overproof data. Therefore when using these embedding-based models on historical corpora, it is important to also use word embeddings matching the time period and genre of the target corpus. This area is worth further investigation in future work. We also noted the qualitatively dissimilar nature of ETM and GLDA topics and the high correspondence of ETM topics from OCR data with topics from the aligned GS data.

GCAT (Gov't/Social)	**CER: 0%**	**CER: 21%**
LDA	47 : rights, people, government, political, human 44 : tax, budget, billion, government 9 : court, case, law, judge, legal	55 : court, police, law, group, people 10 : party, war, vote, home, since 49 : talks, told, minister, prime, peace
ETM	17 : workers, union, strike, government, pay 64 : police, people, two, killed, security 61 : court, case, judge, trial, charges	95 : police, people, two, men, bomb 82 : tax, budget, billion, cut, year 30 : party, vote, poll, election, prime
M14 (Commodity Markets)	**CER: 0%**	**CER: 21%**
LDA	45 : rates, rate, percent, interest, inflation 0 : settlement, lawsuit, suit, filed, ford 71 : inc, municipal, securities, desk, smith	87 : said, percent, market, tho, rate
ETM	91 : rate, inflation, rates, rise, interest	19 : rate, percent, rates, data, cut

Table 3: Top weighted topics for some article classes in the synthetic dataset. Topics are *not* aligned across noise levels and models.

Our experiments on synthetic data revealed that while ETM, like LDA, degraded in terms of topic coherence and classification accuracy as noise increased, it did so slower than LDA. But unlike LDA and GLDA, ETM topics became less diverse as noise increased. GLDA improved in topic coherence with increased noise and produced more varied topics but performed worse in document classification because its topics do not correlate with the gold standard labels in the dataset. We also showed that increasing noise causes LDA and GLDA to become less stable.

LDA is a popular method for analysing digitised historical collections but it is not without its shortcomings, especially when applied to documents with OCR errors. In our experiments, we have shown that topic models that incorporate information from word embeddings improve over LDA in the presence of OCR noise in terms of coherence, diversity, and classification performance.

References

1. Batmanghelich, K., Saeedi, A., Narasimhan, K., Gershman, S.: Nonparametric spherical topic modeling with word embeddings. In: Proceedings of the conference. Association for Computational Linguistics. Meeting. vol. 2016, p. 537. NIH Public Access (2016)
2. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of machine Learning research **3**(Jan), 993–1022 (2003)
3. Blevins, C.: Topic modeling martha ballard's diary. Cameron Blevins (2010)
4. Boyd-Graber, J.L., Hu, Y., Mimno, D., et al.: Applications of topic models, vol. 11. now Publishers Incorporated (2017)
5. Das, R., Zaheer, M., Dyer, C.: Gaussian lda for topic models with word embeddings. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). pp. 795–804 (2015)

6. Dieng, A.B., Ruiz, F.J., Blei, D.M.: Topic modeling in embedding spaces. Transactions of the Association for Computational Linguistics **8**, 439–453 (2020)
7. Evershed, J., Fitch, K.: Correcting noisy ocr: Context beats confusion. In: Proceedings of the First International Conference on Digital Access to Textual Cultural Heritage. pp. 45–51 (2014)
8. Hill, M.J., Hengchen, S.: Quantifying the impact of dirty OCR on historical text analysis: Eighteenth century collections online as a case study. Digital Scholarship in the Humanities **34**(4), 825–843 (2019)
9. Hu, W., Tsujii, J.: A latent concept topic model for robust topic inference using word embeddings. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). pp. 380–386 (2016)
10. Lewis, D.D., Yang, Y., Rose, T.G., Li, F.: Rcv1: A new benchmark collection for text categorization research. Journal of machine learning research **5**(Apr), 361–397 (2004)
11. Marjanen, J., Zosa, E., Hengchen, S., Pivovarova, L., Tolonen, M.: Topic modelling discourse dynamics in historical newspapers. arXiv preprint arXiv:2011.10428 (2020)
12. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
13. Mimno, D.: Computational historiography: Data mining in a century of classics journals. Journal on Computing and Cultural Heritage (JOCCH) **5**(1), 1–19 (2012)
14. Mutuvi, S., Doucet, A., Odeo, M., Jatowt, A.: Evaluating the impact of ocr errors on topic modeling. In: International Conference on Asian Digital Libraries. pp. 3–14. Springer (2018)
15. Nguyen, T.T.H., Jatowt, A., Coustaty, M., Nguyen, N.V., Doucet, A.: Deep statistical analysis of ocr errors for effective post-ocr processing. In: 2019 ACM/IEEE Joint Conference on Digital Libraries (JCDL). pp. 29–38. IEEE (2019)
16. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). pp. 1532–1543 (2014)
17. Řehůřek, R., Sojka, P.: Software Framework for Topic Modelling with Large Corpora. In: Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks. pp. 45–50. ELRA, Valletta, Malta (May 2010), http://is.muni.cz/publication/884893/en
18. Roberts, M.E., Stewart, B.M., Tingley, D., Airoldi, E.M., et al.: The structural topic model and applied social science. In: Advances in neural information processing systems workshop on topic models: computation, application, and evaluation. vol. 4. Harrahs and Harveys, Lake Tahoe (2013)
19. Röder, M., Both, A., Hinneburg, A.: Exploring the space of topic coherence measures. In: Proceedings of the eighth ACM international conference on Web search and data mining. pp. 399–408 (2015)
20. Viola, L., Verheul, J.: Mining ethnicity: Discourse-driven topic modelling of immigrant discourses in the USA, 1898–1920. Digital Scholarship in the Humanities (2019)
21. Walker, D., Lund, W.B., Ringger, E.: Evaluating models of latent document semantics in the presence of ocr errors. In: Proceedings of the 2010 conference on empirical methods in natural language processing. pp. 240–250 (2010)
22. Yang, T.I., Torget, A., Mihalcea, R.: Topic modeling on historical newspapers. In: Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities. pp. 96–104 (2011)

C. Manuscript: A Comparison of Unsupervised Methods for Ad hoc Cross-Lingual

A Comparison of Unsupervised Methods for Ad hoc Cross-Lingual Document Retrieval

Elaine Zosa, Mark Granroth-Wilding, Lidia Pivovarova
University of Helsinki
Helsinki, Finland
firstname.lastname@helsinki.fi

Abstract

We address the problem of linking related documents across languages in a multilingual collection. We evaluate three diverse unsupervised methods to represent and compare documents: (1) multilingual topic model; (2) cross-lingual document embeddings; and (3) Wasserstein distance. We test the performance of these methods in retrieving news articles in Swedish that are known to be related to a given Finnish article. The results show that ensembles of the methods outperform the stand-alone methods, suggesting that they capture complementary characteristics of the documents.

1. Introduction

We address the problem of retrieving related documents across languages through unsupervised cross-lingual methods that do not use translations or other lexical resources, such as dictionaries. There is a multitude of multilingual resources on the Internet such as Wikipedia, multilingual news sites, and historical archives. Many users may speak multiple languages or work in a context where discovering related documents in different languages is valuable, such as historical enquiry. This calls for tools that relate resources across language boundaries.

We choose to focus on methods that do not use translations because lexical resources and translation models vary across languages and time periods. Our goal is to find methods that are applicable across these contexts without extensive fine-tuning or manual annotation. Much work on cross-lingual document retrieval (CLDR) has focused on *cross-lingual word embeddings* but topic-based methods have also been used (Wang et al., 2016). Previous work has applied such cross-lingual learning methods to *known item search* where the task is to retrieve one relevant document given a query document (Balikas et al., 2018; Josifoski et al., 2019; Litschko et al., 2019). We are interested in *ad hoc retrieval* where there could be any number of relevant documents and the task is to rank the documents in the target collection according to their relevance to the query document (Voorhees, 2003).

Here we evaluate three existing unsupervised or weakly supervised methods previously used in CLDR for slightly different tasks: (1) multilingual topic model (MLTM); (2) document embeddings derived from cross-lingual reduced rank ridge regression or Cr5 (Josifoski et al., 2019) and; (3) Wasserstein distance for CLDR (Balikas et al., 2018). These methods link documents across languages in fundamentally different ways. MLTM induces a shared cross-lingual topic space and represents documents as a language-independent distribution over these topics; Cr5 obtains cross-lingual document embeddings; and the Wasserstein distance as used by (Balikas et al., 2018) computes distances between documents as sets of cross-lingual word embeddings (Speer et al., 2016). The methods broadly cover the landscape of recent CLDR methods. To our knowledge, this is the first comparison of Cr5 and Wasserstein for ad hoc retrieval.

This paper adds to the literature on CLDR in three ways: (1) evaluating unsupervised methods for retrieving related documents across languages (ad hoc retrieval), in contrast to retrieval of a single corresponding document; (2) evaluating different ensembling methods; and (3) demonstrating the effectiveness of relating documents across languages through complementary methods.

2. Related Work

Previous work on linking documents across languages has used translation-based features, where the query is translated into the target language and the retrieval task proceeds in the target language (Hull and Grefenstette, 1996; Litschko et al., 2018; Utiyama and Isahara, 2003). Other methods used term-frequency correlation (Tao and Zhai, 2005; Vu et al., 2009), sentence alignment (Utiyama and Isahara, 2003), and named entities (Montalvo et al., 2006). In this paper, we are interested in language-independent models with minimal reliance on lexical resources and other metadata or annotations.

2.1. Multilingual topic model

The multilingual topic model (MLTM) is an extension of LDA topic modelling (Blei et al., 2003) for comparable multilingual corpora (De Smet and Moens, 2009; Mimno et al., 2009). In contrast to LDA, which learns topics by treating each document as independent, MLTM relies on a topically aligned corpus, which consists of tuples of documents in different languages discussing the same themes. MLTM learns separate but aligned topic distributions over the vocabularies of the languages represented in the corpus. One of the main advantages of MLTM is that it can extend across any number of languages, not just two, as long as there is a topically aligned corpus covering these languages. This can be difficult because aligning corpora is not a trivial task, especially as the number of languages gets larger. For this reason, Wikipedia, currently in more than 200 languages, is a popular source of training data for MLTM.

Another issue facing topic models is that the choice of hyperparameters can significantly affect the quality and nature of topics extracted from the corpus and, consequently,

its performance in the downstream task we want use it for. There are three main hyperparameters in LDA-based models: the number of topics to extract, K; the document concentration parameter, α, that controls the sparsity of the topics associated with each document; and the topic concentration parameter, β, which controls the sparsity of the topic-specific distribution over the vocabulary.

2.2. Cross-lingual document embeddings

Cross-lingual reduced-rank ridge regression (Cr5) was recently introduced as a novel method of obtaining cross-lingual document embeddings (Josifoski et al., 2019). The authors formulate the problem of inducing a shared document embedding space as a linear classification problem. Documents in a multilingual corpus are assigned language-independent concepts. The linear classifier is trained to assign the concepts to documents, learning a matrix of weights W that embeds documents in a concept space close to other documents labelled with the same concept and far from documents expressing different concepts.

They train on a multilingual Wikipedia corpus, where articles are assigned labels based on language-independent Wikipedia concepts. They show that the method outperforms the state-of-the-art cross-lingual document embedding method from previous literature (Litschko et al., 2018). Cr5 is trained to produce document embeddings, but can also be used to obtain embeddings for smaller units, such as sentences and words. One disadvantage is that it requires labelled documents for training. However, the induced cross-lingual vectors can then be used for any tasks in which the input document is made up of words in the vocabulary of the corresponding language in the training set.

2.3. Wasserstein distances for documents

Wasserstein distance is a distance metric between probability distributions and has been previously used to compute distances between text documents in the same language (*Word Mover's Distance* (Kusner et al., 2015)). In (Balikas et al., 2018) the authors propose the Wasserstein distance to compute distances between documents from different languages. Each document is a set of cross-lingual word embeddings (Speer et al., 2016) and each word is associated with some weight, such as its term frequency inverse document frequency (tf.idf). The Wasserstein distance is then the minimum cost of transforming all the words in a query document to the words in a target document. They then demonstrate that using a regularized version of the Wasserstein distance makes the optimization problem faster to solve and, more importantly, allows multiple associations between words in the query and target documents.

3. Experimental setup

3.1. Task and dataset

We evaluate using a dataset of Finnish and Swedish news articles published by the Finnish broadcaster YLE and freely available for download from the Finnish Language Bank[1]. The articles are from 2012-18 and are written separately in the two languages (not translations and not parallel). This dataset contains 604,297 articles in Finnish and 228,473 articles in Swedish. Each article is tagged with a set of keywords describing the subject of the article. These keywords were assigned to the articles by a combination of automated methods and manual curation. The keywords vary in specificity, from named entities, such as *Sauli Niinisto* (the Finnish president), to general subjects, such as *talous* (sv: *ekonomi*, en: economy). On average, Swedish articles are tagged with five keywords and 15 keywords for Finnish articles. Keywords are provided in Finnish and Swedish regardless of the article language so no additional mapping is required.

To build a corpus of related news articles for testing, we associate one Finnish article with one or more Swedish articles if they share three or more keywords and if the articles are published in the same month. From this we create three separate test sets: 2013, 2014, and 2015. For each month, we take 100 Finnish articles to use as queries, providing all of the related Swedish articles as a candidate set visible to the models.

To build a topically aligned corpus for training MLTM, we match a Finnish article with a Swedish article if they were published within two days of each other and share three or more keywords. As a result no Finnish article is matched with more than one Swedish article and vice-versa so that we have a set of aligned unique article pairs. To train MLTM we use a year which is preceding the testing year: e.g., we train a model using articles from 2012 and test it on articles from 2013. Unaligned articles are not used for either training or testing. The script for article alignment will be provided in the Github repository for this work.

Table 1 shows the statistics of the training and test sets. As can be seen in the last column of the table, one Finnish article corresonds to almost twenty Swedish articles for the 2013 dataset and more than thirty for the other two datasets. This is typical for large news collections, since one article may have an arbitrary number of related articles. Thus, our corpus is more suitable for ad-hoc search evaluation than Wikipedia or Europarl corpus, since they contain only one-to-one relation[2].

	MLTM Train set	Test set	
	articles per lang	**#candidates**	**#related**
2012	7.2K	-	-
2013	7.2K	1.3K	19.5
2014	7.2K	1.4K	31.8
2015	-	1.5K	35.9

Table 1: Statistics of the training set for training MLTMs and test sets for each year. #candidates is the average size of the candidate articles set and #related is the average number of Swedish articles related to each Finnish article.

3.2. Models

We use our in-house implementation of MLTM training using Gibbs sampling[3]. The training corpus was tokenized, lemmatized and stopwords were removed. We limited the

[1] https://www.kielipankki.fi/corpora/

[2] CLEF 2000-2003 ad-hoc retrieval Test Suite, which also contains many-to-many relations, is not freely available

[3] https://github.com/ezosa/cross-lingual-linking.git

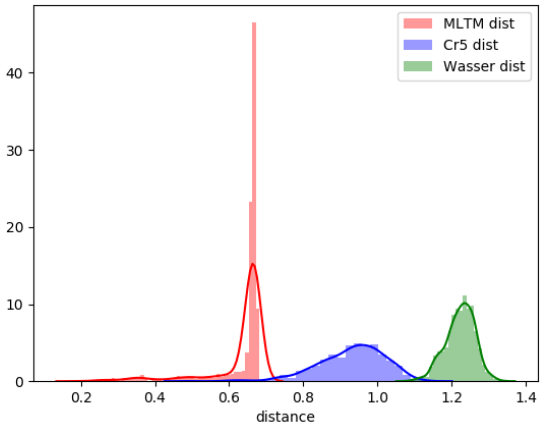


Figure 1: Density plots of the distances between one query document and the candidate documents.

vocabulary to the 9,000 most frequent terms for each language. We train three separate models for 2012, 2013, and 2014 (for the 2013, 2014, and 2015 test sets, respectively). We train all three models with $K = 100$ topics, $\alpha = 1/K$ and $\beta = 0.08$. We use 1,000 iterations for burn-in and then infer vectors for unseen documents by sampling every 25th iteration for 200 iterations. To obtain distances between documents, we compute the Jensen-Shannon (JS) divergence between the document-topic distributions of the query document and each of the candidate documents.

For Cr5, we use pretrained word embeddings for Finnish and Swedish provided by the authors[4]. We construct document embeddings according to the original method – by summing up the embeddings of the words in the document weighted by their frequency. We compute the distance between documents as the cosine distance of the document embeddings.

For Wasserstein distance, we use code provided by the authors for computing distances between documents and use the same cross-lingual embeddings they did in their experiments[5] (Speer et al., 2016). Wasserstein distance has a regularization parameter λ that controls how the model matches words in the query and candidate documents. The authors suggested using $\lambda = 0.1$ because it encourages more relaxed associations between words. Higher values of λ create stronger associations while too low values fail to associate words that are direct translations of each other. In this task, it might make more sense to use lower λ values, though an experiment with $\lambda = 0.01$ brought no noticeable improvement in performance (see Section 3.3.).

We created ensemble models by averaging the document distances from the stand-alone models and ranking candidate documents according to this score. We construct four ensemble models by combining each pair of models, as well as all three: **MLTM_Wass**; **Cr5_Wass**; **MLTM_Cr5**; and **MLTM_Cr5_Wass**.

3.3. Results and Discussion

Table 2 shows the results for each model and ensemble on each of the three test sets, reporting the precision of the top-ranked k results and mean reciprocal rank (MRR). Cr5 is the best-performing stand-alone model by a large margin. Cr5 was originally designed for creating cross-lingual document embeddings by classifying Wikipedia documents according to concepts. We did not retrain it for our particular task. Nevertheless, using these pre-trained word embeddings we were able to retrieve articles that discuss similar subjects in this different domain. However, it is worth noting that Cr5 can only be trained on languages for which labels are available for *some* similarly transferable training domain.

MLTM, being a topic-based model, would seem like the obvious choice for a task like this because we want to find articles that share some broad characteristics with the query document, even if they do not discuss the same named entities or use similar words. However, Cr5 outperforms MLTM on its own. One reason may be that 100 topics are too few. We chose this number because it seemed to give topics that are specific enough for short articles but still broad enough that they could reasonably be used to describe similar articles. Another drawback of this model is that it does not handle out-of-vocabulary words and the choice of using a vocabulary of 9,000 terms might be too low.

Wasserstein distance is the worst-performing of the stand-alone models especially for the 2014 and 2015 test sets where it offers little improvement when ensembled with Cr5 (Cr5_Wass). A possible reason is that it attempts to transform one document to another and therefore favors documents that share a similar vocabulary to the query document. The technique might be suitable for matching Wikipedia articles, as shown in (Balikas et al., 2018) because they talk about the same subject at a fine-grained level and use similar words, whilst in our task the goal is to make broader connections between documents.

In Figure 1, the density plots of the distances of one query document and the candidate documents. We see that MLTM and Wasserstein tend to have sharper peaks while Cr5 distances are flatter. MLTM has minimum and maximum distances of 0.2 and 0.68, respectively, while Cr5 has 0.49 and 1.14, and Wasserstein has 1.08 and 1.34. Topic modelling tends to predict that most of the target documents are far from the query document (peaks at the right side). This is not only true for this particular query document but for other query documents in our test set as well. We also see that Wasserstein has larger distances which is potentially problematic. We tried normalizing the distances produced by the models such that they are centered at zero and using these distances for the ensembled model however it produces the same document rankings as the unnormalized distances. This might be because we are only concerned with the documents with the smallest distances where Wasserstein does not contribute much.

For the ensemble models, combining all three models per-

[4]https://github.com/epfl-dlab/Cr5

[5]https://github.com/balikasg/WassersteinRetrieval

Test set:	2013				2014				2015			
Measure:	**P@1**	**P@5**	**P@10**	**MRR**	**P@1**	**P@5**	**P@10**	**MRR**	**P@1**	**P@5**	**P@10**	**MRR**
MLTM	21.8	18.2	16.3	31.6	24.1	22.4	20.6	34.8	30.8	29.0	27.1	41.6
Wass	21.1	13.7	11.3	30.8	21.0	16.9	14.7	31.9	25.1	20.6	17.9	37.2
Wass $\lambda = 0.01$	20.3	13.5	11.1	30.0	21.3	16.8	14.6	32.0	25.1	20.1	17.3	36.6
Cr5	32.5	24.5	21.2	41.7	38.3	30.2	26.0	48.0	43.1	37.1	33.5	53.8
MLTM_Wass	24.6	21.3	19.1	35.2	27.3	25.5	23.4	38.2	30.4	31.4	30.1	42.9
Cr5_Wass	35.4	27.4	23.2	45.2	38.1	32.2	28.2	49.2	41.2	37.7	34.9	52.9
MLTM_Cr5	36.4	28.2	24.4	46.6	**44.8**	34.3	30.1	53.6	42.7	40.1	36.9	54.5
MLTM_Cr5_Wass	**40.7**	**30.7**	**26.3**	**50.3**	43.0	**36.1**	**31.9**	**53.8**	**44.5**	**41.3**	**38.5**	**55.9**

Table 2: Precision at k and MRR of cross-lingual linking of related news articles obtained by three stand-alone models and four ensemble models.

Test set:	2013	2014	2015	AVG
MLTM, Wass	-0.039	-0.016	-0.022	-0.026
Cr5, Wass	0.128	0.027	0.026	0.060
MLTM, Cr5	0.156	0.164	0.178	0.166

Table 3: Mean Spearman correlation of the ranks of candidate documents for each pair of models.

forms best overall for all three test sets and all but one precision level—the only exception is P1 for 2014 where MLTM_Cr5 achieves roughly the same performance. This tells us that each model sometimes finds relevant documents not found by the other models. The correlation of candidate document rankings between the different methods is quite low (Table 3). We compute the correlation between the ranks for each of the 1200 query documents (100 queries for each month) for each year of our test set and average them. As can be seen in the table the correlations are rather low, which means that they retrieve documents based on different principles. The highest correlation is between **MLTM** has the **Cr5** while correlation between **MLTM** and **Wass** is the lowest.

This suggests that there are different ways of retrieving related documents across languages and that the three methods of cross-lingual embeddings, cross-lingual topic spaces and cross-lingual distance measures capture complementary notions of similarity. A simple combination of their decisions is thus able to make better judgements than any can make on its own.

As an example, in Table 4 we show excerpts from a query article in Finnish and some of the related Swedish articles correctly predicted by the different models. For this article, Cr5 gave 10 correct predictions in its top 10 (perfect precision), MLTM gave 8 correct predictions and Wasserstein only 4. Like Cr5, the ensemble model MLTM_Cr5_Wass also achieved perfect precision. MLTM and MLTM_Cr5_Wass shared 4 correct predictions while Cr5 and MLTM_Cr5_Wass shared 7. All the articles correctly predicted by Wasserstein were also predicted by the other models. We show articles from Cr5, MLTM and MLTM_Cr5_Wass that was correctly predicted by that model only and for Wasserstein, we show the top correct article that it predicted.

4. Conclusions and Future work

In this paper we compare three different methods for cross-lingual ad hoc document retrieval by applying them to the task of retrieving Swedish news articles that are related to a given Finnish article. We show that a word-embedding based model, Cr5, performs best followed by the multilingual topic model and the distance-based Wasserstein model has the worst results of the stand-alone models. We then demonstrate that combining at least two of these methods by averaging their distances yields better results than the models used on their own. Finally we show that combining the three models yields the best results. These results tell us that relating documents based on different techniques such as embedding-based or topic-based techniques yields different results and that pooling these results make for a better model.

In the future we plan to investigate the performance of word embedding-based multilingual topic models in this task. There is already some work done on developing topic models that use word embeddings (Batmanghelich et al., 2016; Das et al., 2015). To our knowledge, they have not yet been applied to cross-lingual embeddings. Such a model could potentially combine the benefits of the multilingual topic model with word embeddings for retrieving similar documents across languages.

We also plan to further experiments with multilingual topic models for languages where the amount of linked documents is scarce. In this work, we trained the topic model with thousands of linked articles because the articles were annotated with tags however this might not always be the case, for instance with historical data sets or under-resourced languages where there are not readily available annotated data and manual annotation is time-consuming or requires expert knowledge. In such cases, we could still train a multilingual topic model with smaller amounts of aligned training data or perhaps a training set where some articles do not have a counterpart article in the other language.

There is also scope for further exploration of ensemble methods, going beyond the simple combination of distance metrics we have applied here. As well as combining models in different ways, further, potentially complementary,

Query article	Yleisradion YleX-kanavan kymmenen suosituimman kappaleen listalla,valtaosa on suomalaisartisteja tai -yhtyeitä. Radio Suomen kaikki,kymmenen eniten kuultua kappaletta ovat odotetusti kotimaisia. YleX ja Radio Suomi ovat koonneet listan eniten soittamastaan musiikista vuonna 2012.
MLTM	På min låtlista finns låtar som på olika sätt och från olika perspektiv beskriver livets grundläggande vemod eller "life bitter-sweet", som man brukar säga på Irland. Det säger Tom Sjöblom, som har valt musiken denna vecka i [Min musik.]
Cr5	De isländska banden tar över världen, vi träffade Sóley som nyligen varit på USA-turné med sina isländska kollegor Of Monsters And Men. **Sóley** är isländska och betyder solros. Sóley är också namnet på sångerskan som är en av de mest intressanta nya musikexporterna som kommit från Island.
Wasserstein	Både Radio Vega och Radio Extrem har börjat spela låtar som tävlar i Tävlingen för ny musik UMK. Radio Extrem har tagit in både Krista Siegfrids __Marry me__ och Diandras __Colliding into you__ på spellistan, och låtarna kommer att spelas två gånger om dagen åtminstone nu i början.
MLTM_Cr5_Wass	Smakproven på 30 sekunder av de tolv UMK låtarna kittlade fantasin så,där passligt, men nu behöver vi inte längre gissa oss till hur sångerna,låter i sin helhet. De färdigt producerade bidragen kan nu höras på,Arenan.

Table 4: Excerpt from a query Finnish article and some related Swedish articles correctly predicted by the models. The query article is about popular songs on Finnish radio.

measures of document similarity could be included: for example, explicitly taking into account overlap of named entities, or document publishing metadata if such information is available.

Acknowledgements

This work has been supported by the European Union's Horizon 2020 research and innovation programme under grant 770299 (NewsEye) and 825153 (EMBEDDIA).

References

Balikas, G., Laclau, C., Redko, I., and Amini, M.-R. (2018). Cross-lingual document retrieval using regularized Wasserstein distance. In *European Conference on Information Retrieval*, pages 398–410. Springer.

Batmanghelich, K., Saeedi, A., Narasimhan, K., and Gershman, S. (2016). Nonparametric spherical topic modeling with word embeddings. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, volume 2016, page 537. NIH Public Access.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Das, R., Zaheer, M., and Dyer, C. (2015). Gaussian LDA for topic models with word embeddings. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 795–804.

De Smet, W. and Moens, M.-F. (2009). Cross-language linking of news stories on the web using interlingual topic modelling. In *Proceedings of the 2nd ACM workshop on Social web search and mining*, pages 57–64. ACM.

Hull, D. A. and Grefenstette, G. (1996). Querying across languages: a dictionary-based approach to multilingual information retrieval. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 49–57. Citeseer.

Josifoski, M., Paskov, I. S., Paskov, H. S., Jaggi, M., and West, R. (2019). Crosslingual document embedding as reduced-rank ridge regression. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pages 744–752. ACM.

Kusner, M., Sun, Y., Kolkin, N., and Weinberger, K. (2015). From word embeddings to document distances. In *International conference on machine learning*, pages 957–966.

Litschko, R., Glavaš, G., Ponzetto, S. P., and Vulić, I. (2018). Unsupervised cross-lingual information retrieval using monolingual data only. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 1253–1256. ACM.

Litschko, R., Glavaš, G., Vulic, I., and Dietz, L. (2019). Evaluating resource-lean cross-lingual embedding models in unsupervised retrieval. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1109–1112. ACM.

Mimno, D., Wallach, H. M., Naradowsky, J., Smith, D. A., and McCallum, A. (2009). Polylingual topic models. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*, pages 880–889. Association for Computational Linguistics.

Montalvo, S., Martinez, R., Casillas, A., and Fresno, V. (2006). Multilingual document clustering: an heuristic approach based on cognate named entities. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 1145–1152. Association for Computational Linguistics.

Speer, R., Chin, J., and Havasi, C. (2016). ConceptNet 5.5: An open multilingual graph of general knowledge. *CoRR*, abs/1612.03975.

Tao, T. and Zhai, C. (2005). Mining comparable bilingual text corpora for cross-language information integration. In *Proceedings of the eleventh ACM SIGKDD interna-*

tional conference on Knowledge discovery in data mining, pages 691–696. ACM.

Utiyama, M. and Isahara, H. (2003). Reliable measures for aligning Japanese-English news articles and sentences. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 72–79. Association for Computational Linguistics.

Voorhees, E. (2003). Overview of TREC 2003. pages 1–13, 01.

Vu, T., Aw, A. T., and Zhang, M. (2009). Feature-based method for document alignment in comparable news corpora. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 843–851. Association for Computational Linguistics.

Wang, Y.-C., Wu, C.-K., and Tsai, R. T.-H. (2016). Cross-language article linking with different knowledge bases using bilingual topic model and translation features. *Knowledge-Based Systems*, 111:228–236.

D. Manuscript: Multilingual Dynamic Topic Model

Multilingual Dynamic Topic Model

Elaine Zosa and **Mark Granroth-Wilding**
Department of Computer Science
University of Helsinki
Helsinki, Finland
firstname.lastname@helsinki.fi

1 Abstract

Dynamic topic models (DTMs) capture the evolution of topics and trends in time series data. Current DTMs are applicable only to monolingual datasets. In this paper we present the multilingual dynamic topic model (ML-DTM), a novel topic model that combines DTM with an existing multilingual topic modeling method to capture cross-lingual topics that evolve across time. We present results of this model on a parallel German-English corpus of news articles and a comparable corpus of Finnish and Swedish news articles. We demonstrate the capability of ML-DTM to track significant events related to a topic and show that it finds distinct topics and performs as well as existing multilingual topic models in aligning cross-lingual topics.

2 Introduction

Dynamic topic models (DTMs, Blei and Lafferty, 2006) capture themes or topics discussed in a set of time-stamped documents and how the words related to these topics change in prominence over time. Other topic models have been proposed that aim to model time series data (Wang and McCallum, 2006; Wei et al., 2007; Hall et al., 2008). These models can be used to explore historical document collections to study historical trends, language changes (Frermann and Lapata, 2016) and track the emergence and evolution of certain subjects (Hall et al., 2008; Yang et al., 2011).

With the internet becoming more multilingual it is increasingly important to build cross-lingual tools to bridge different linguistic groups online. Fortunately, large multilingual datasets such as Wikipedia, the Europarl parallel corpus (Koehn, 2005) and other datasets assembled from crawling the web (Van Gael and Zhu, 2007) are also becoming widely available to researchers. This has led to the development of several multilingual topic models to infer topics from multilingual datasets. Examples include the polylingual topic model (PLTM, Mimno et al., 2009), multilingual topic model for unaligned text (MuTo, Boyd-Graber and Blei, 2009), and JointLDA (Jagarlamudi and Daumé, 2010). What is currently lacking are topic models for multilingual time-stamped data that can model historical and linguistic changes in a specific context. Digitalization efforts in libraries and archives, such as the Europeana collections[1], have made available online historical document collections from different European countries. Collections such as these are valuable resources for comparing historical trends in different countries. However, scholars and other interested parties may not possess the linguistic skills necessary to explore such data and would benefit from tools to automatically discover connections across linguistic boundaries.

In this paper, we present the multilingual dynamic topic model (ML-DTM), a novel topic model that captures dynamic topics from broadly topically aligned multilingual datasets. We extend a DTM inference method by Bhadury et al. (2016) to train this model.

In the following sections, we give a broad review of related work, discuss existing *dynamic* and *multilingual* topic models in more detail, and then give a description of our proposed combined model. We then demonstrate usage of this model on a parallel dataset and a comparable dataset of news articles and present our results. We show that this novel topic model learns aligned bilingual topics as demonstrated by the cosine similarities of learned vector representations of named entities. Table 1 summarizes the notations used in this paper. Code is available at: https://github.com/ezosa/multilingual_dtm.

[1] https://www.europeana.eu

Symbol	Description
α	parameter for θ
β	hyperparameter for ϕ
ψ	hyperparameter for θ
θ	distribution of topics over a document
ϕ	distribution of words over a topic
D	set of documents
W_d	words in document d
N_d	number of words in document d, or $\|W_d\|$
Z_d	topic assignments of words in document d
K	number of topics
T	number of time slices
L	number of languages in the dataset
V	words in a vocabulary for language

Table 1: Summary of notations.

3 Related Work

Topic models capture themes inherent in document collections through the co-occurence patterns of the words in documents. Latent Dirichlet Allocation (LDA, Blei et al., 2003) is a popular method for inferring these themes or topics. It is generative document model where a document is described by a mixture of different topics and each topic is a probability distribution over the words in the vocabulary. In a document collection we can only observe the *words* in a document. Therefore, training a model involves inferring these latent variables through approximate inference methods.

In the case of documents with timestamps covering some time interval, such as news articles, we might want to capture *dynamic* co-occurence patterns that evolve through time. Dynamic Topic Model (DTM, Blei and Lafferty, 2006) divides time into discrete slices and chains parameters from each slice in order to infer topics that are aligned across time. DTM gives us a set of topic-term distributions that evolve from one time slice to the next. There are also other topic models for time-series data such as the Continuous Dynamic Topic Model (cDTM, Wang et al., 2008), a version of DTM that does not explicitly discretize time intervals. Dynamic Mixture Model (DMM, Wei et al., 2007) captures the evolution of documents across time and Topics over Time (TOT, Wang and McCallum, 2006) is a method that models the prominence of topics over time.

A limitation of LDA, as well as these dynamic models, is that it is not applicable to multilingual data. LDA captures co-occurences of words in documents and words from different languages would rarely, if ever, occur in the same document regardless of their semantics, as demonstrated by experiments on the Europarl corpus (Jagarlamudi and Daumé, 2010; Boyd-Graber and Blei, 2009). Multilingual topic models are developed to capture cross-lingual topics from multilingual datasets.

Polylingual Topic Model (PLTM, Mimno et al., 2009) is a multilingual topic model that extends LDA for an aligned multilingual corpus. Instead of running topic inference on individual documents as in LDA, PLTM infers topics for *tuples* of documents, where each document in the tuple is in a different language. PLTM assumes that the documents of a tuple discuss the same subject broadly and therefore share the same document-topic distribution.

Other topic models for multilingual data include Multilingual Topic Model for Unaligned Text (MuTo, Boyd-Graber and Blei, 2009) and JointLDA (Jagarlamudi and Daumé, 2010). MuTo attempts to match words between languages in the corpus and samples topic assignments for these matchings. JointLDA is a multilingual model that does not require an aligned corpus but requires a bilingual dictionary and uses concepts, instead of words, to infer topics where concepts can be entries in the bilingual dictionary.

In this work we will focus on DTM and PLTM because we want to capture topic evolution in multilingual settings without using additional lexical resources such as dictionaries.

3.1 Dynamic Topic Model

LDA uses Dirichlet and multinomial distributions for inferring both topic-term distributions ϕ and document-topic distributions θ. The conjugacy of these distributions allow ϕ and θ to be integrated out leaving us only with the posterior distribution for topic-term assignments Z, which we can sample through Gibbs sampling (Griffiths and Steyvers, 2004). Inference in DTM, however, is

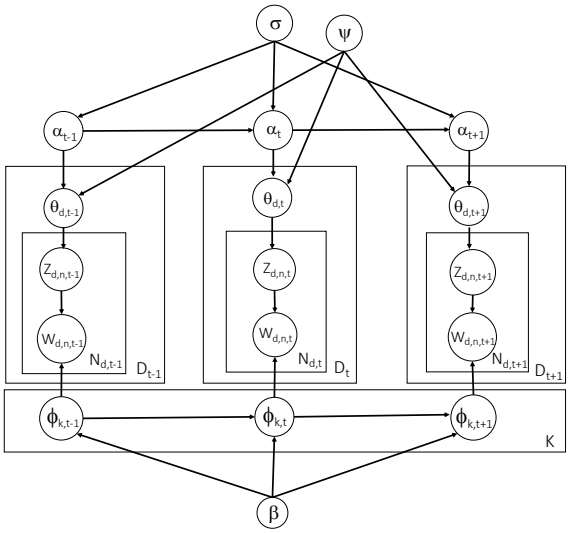


Figure 1: DTM for three time slices as shown in Bhadury et al. (2016).

more complicated due to the non-conjugacy of the distributions used in the model. Blei and Lafferty (2006) use variational Kalman filtering for topic inference, which does not scale well for a large number of topics and documents and large numbers of time slices (Bhadury et al., 2016; Wang et al., 2008). Bhadury et al. (2016) developed a method for inferring the posterior distributions of DTM with Gibbs sampling. In their method, the parameters α, θ, ϕ and Z are re-sampled during every iteration of the sampler.

The document-topic proportions θ, sampled for each document in each time slice, and the topic-term distributions ϕ, sampled for each topic in each time slice, are updated using Stochastic Gradient Langevin Dynamics (SGLD, Welling and Teh, 2011) which is based on Stochastic Gradient Descent (SGD). Figure 1 shows the plate diagram for DTM from Bhadury et al. (2016).

3.2 Polylingual Topic Model

The polylingual topic model (PLTM, Mimno et al., 2009) is an extension of LDA that infers topics from an aligned multilingual corpus composed of document tuples. Tuples are composed of documents in different languages that are thematically aligned, meaning that they discuss the subject in broadly similar ways. For instance, a news article in German and another article in English that report on the same event can compose a tuple.

Inference on PLTM can be done via Gibbs sampling where the topic assignment of each term $z^l_{d,n}$ is resampled during every iteration. Following Vulić et al. (2015), we provide the update formulae for the bilingual case for brevity. The update formulae for documents in languages x and y are:

$$P(z^x_{d,n} = k|z^x, z^y, w^x, w^y, \alpha, \beta) \propto \frac{m^x_{d,k} - 1 + m^y_{d,k} + \alpha}{\sum_{i=1}^{K} m^x_{d,i} - 1 + \sum_{i=1}^{K} m^y_{d,i} + K\alpha} \cdot \frac{v^x_{k,w_{d,n}} - 1 + \beta}{\sum_{i=1}^{|V^x|} v^x_{k,w_{d,i}} - 1 + |V^x|\beta} \quad (1)$$

$$P(z^y_{d,n} = k|z^y, z^x, w^y, w^x, \alpha, \beta) \propto \frac{m^y_{d,k} - 1 + m^x_{d,k} + \alpha}{\sum_{i=1}^{K} m^y_{d,i} - 1 + \sum_{i=1}^{K} m^x_{d,i} + K\alpha} \cdot \frac{v^y_{k,w_{d,n}} - 1 + \beta}{\sum_{i=1}^{|V^y|} v^y_{k,w_{d,i}} - 1 + |V^y|\beta} \quad (2)$$

where $m^x_{d,k}$ is the number of times topic k has been assigned to a word in document d written in language x and $v^x_{k,w_{d,n}}$ is the number of times word $w_{d,n}$, that is, the word at position n in document d, has been assigned to topic k. $|V^x|$ is the vocabulary size of language x. The first part of these formulae links the two languages together and is language-independent while the second part is language-specific.

Figure 2 shows the graphical representation of PLTM for l languages.

4 Multilingual Dynamic Topic Model

Here we combine the above *dynamic* and *polylingual* models to produce a *Multilingual Dynamic Topic Model* (ML-DTM). Figure 3 shows the diagram of ML-DTM for two languages and three time slices. Although we show only the bilingual case here for brevity, the model is applicable for any number of languages.

The inference method of Bhadury et al. (2016) was originally motivated by the need to speed up DTM inference for very large datsets. We apply it here to the combined ML-DTM model. We propose the following posterior conditional distribution for $\theta_{x,t}$ where x is a tuple index in the dataset:

$$p(\theta_{x,t}|\alpha_t, Z_{x,t}) \propto \mathcal{N}(\theta_{x,t}|\alpha_t, \psi^2 I) \times \prod_{l=1}^{L} \prod_{n=1}^{N_{d_l,t}} Mult(Z_{d_l,n,t}|\pi(\theta_{x,t})) \quad (3)$$

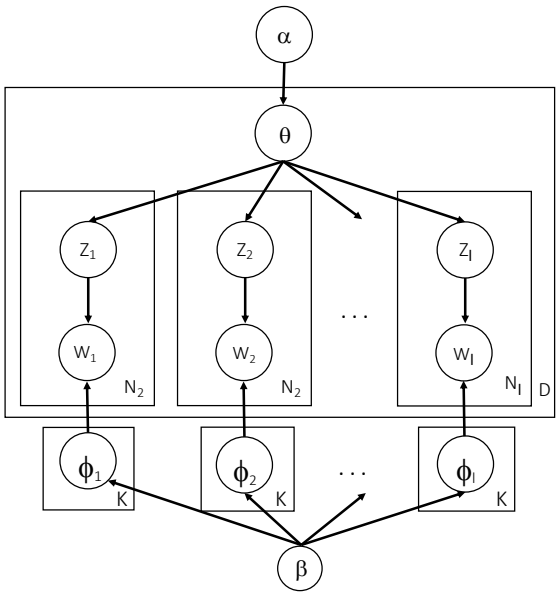


Figure 2: Polylingual topic model for l languages of Mimno et al. (2009).

Following Bhadury et al. (2016), the update equation to evaluate the gradient of $\theta^k_{x,t}$ becomes:

$$\nabla_{\theta^k_{x,t}} \log p(\theta_{x,t}|\alpha_t, Z_{x,t}) = \frac{-1}{\psi^2}(\theta^k_{x,t} - \alpha^k_t) + \sum_{l=1}^{L} C^k_{d_l,t} - \left(N_{d_l,t} \times \frac{\exp(\theta^k_{x,t})}{\sum_j \exp(\theta^j_{x,t})} \right) \quad (4)$$

where $Z_{x,t}$ are the topic assignments for the words in the documents in tuple x at time slice t; $C^k_{d_l,t}$ is the number of times topic k has been assigned to a word in document d_l at time t; and $N_{d_l,t}$ is the length of document d_l at time t.

Instead of evaluating $\theta_{d,t}$ for a single document as in monolingual DTM, we compute $\theta_{x,t}$ for a document *tuple*. The second term in (4) links the languages together by summing up the counts of each document in the tuple.

The equation for evaluating the gradient of the topic-term distributions $\phi_{k,t}$ is the same as in the original paper except that we compute separate distributions for each language since every language has a different vocabulary. This means that for each time slice, instead of updating K different ϕs (one for each topic), we will need to update $K \cdot L$ ϕs. Table 2 shows the dimensions of the parameters to be estimated.

Finally, the topic assignment $Z_{d_l,n,t}$ is sampled

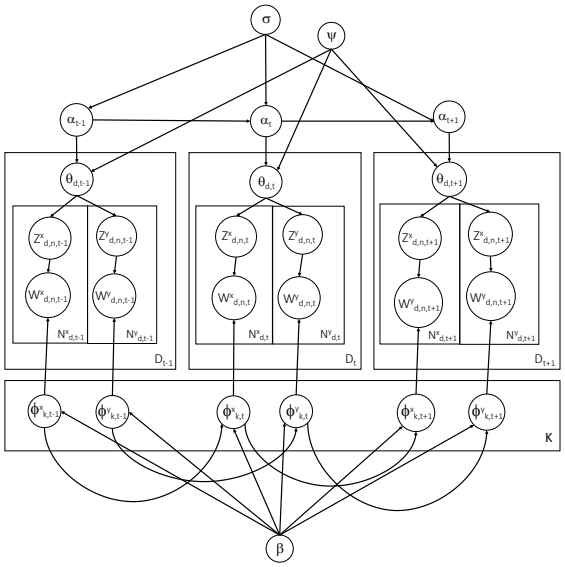


Figure 3: ML-DTM for two languages and three time slices.

Parameter	Dimension
α	$K \times T$
θ	$D^t \times K \times T$
ϕ	$\lvert V^l \rvert \times L \times K \times T$

Table 2: Dimensions of the sampled parameters in the multilingual dynamic topic model (ML-DTM). D^t is the number of document tuples in a dataset.

as in the original paper:

$$P(Z_{d_l,n,t} = k|\theta_{x,t}, \phi^{w_l}_{k,t}) \propto exp(\theta^k_{x,t}) exp(\phi^{w_l}_{k,t}) \quad (5)$$

where w_l is a word from the vocabulary of language l.

5 Evaluation

5.1 Datasets

We ran experiments on ML-DTM with two kinds of data: a parallel dataset and a thematically-comparable one.

The DE-NEWS parallel dataset consists of German news articles from August 1996 to January 2000 with English translations done by human volunteers[2]. This dataset covers 42 months with an average of 200 articles per month. Since this is a parallel corpus there is no need to align the articles.

[2] http://homepages.inf.ed.ac.uk/pkoehn/publications/de-news/

For the comparable dataset, we use the YLE news dataset which consists of Finnish and Swedish articles from the Finnish broadcaster YLE, covering news in Finland from January 2012 to December 2018[3]. The Finnish and Swedish articles are written separately and are not direct translations of each other. We use existing methods for aligning comparable news articles (Utiyama and Isahara, 2003; Vu et al., 2009). Specifically, we create an aligned corpus by pairing a Finnish article with a Swedish article published within a two-day window and sharing three or more named entities. We want to have a one-to-one alignment in our dataset such that no article is duplicated, so we pair a Finnish article with the first Swedish article encountered in the dataset that fits the above criteria and remove the paired articles from the unaligned dataset. The unaligned dataset has a total of 604,297 Finnish articles and 228,473 Swedish articles and the final aligned dataset consists of 123,818 articles covering 84 months. A script for aligning articles using the method described is provided in the Github project associated with this work.

We tokenized, lemmatized (using WordNetLemmatizer for German and English and LAS (Mäkelä, 2016) for Finnish and Swedish) and removed stopwords for these two datasets and then used the 5,000 most frequent words of each language as the vocabulary for that language.

5.2 Cross-Lingual Alignment

We compare the cross-lingual alignment of topics of ML-DTM and PLTM by evaluating the similarity of the learned vector representations of named entities (NEs) that appear in both languages of the same dataset. This method is suggested by Vulić et al. (2015) on the basis that NEs tend to be spelled in the same way in different languages and can be expected to have a similar association with topics across languages. The K-dimensional vector of a NE w for language s is thus:

$$vec(w_s) = [P(z_1|w_s), P(z_2|w_s), ..., P(z_K|w_s)] \quad (6)$$

Under an assumption of a uniform prior over topics, this vector can be computed as:

$$P(z_k|w_s) \propto \frac{P(w_s|z_k)}{P(w_s)} = \frac{\phi_{l,z_k,w_s}}{Norm_{\phi_{s,.,w_s}}} \quad (7)$$

$$Norm_{\phi_{s,.,w_s}} = \sum_{k=1}^{K} \phi_{s,z_k,w_s} \quad (8)$$

$$vec(w_s) = \frac{[\phi_{l,z_1,w_s}, \phi_{l,z_2,w_s}, ..., \phi_{l,z_K,w_s}]}{Norm_{\phi_{s,.,w_s}}} \quad (9)$$

We then take the cosine similarities between the L different vector representations of the NE (for both datasets, $L = 2$).

We evaluate the cosine similarities of NEs that occur five or more times in each time slice. To make the comparison between PLTM and ML-DTM, we train one ML-DTM model on three time slices for 10 topics and three separate PLTM models for each time slice, also capturing 10 topics. We set $\alpha = 1.0$ and $\beta = 0.08$ for PLTM and $\alpha = 0.5$ and $\beta = 0.5$ for ML-DTM for both datasets, which achieved the best results of a small range of values tried. We did not, for now, perform more extensive optimisation of hyperparameters.

5.3 Topic Diversity

We also measure the *diversity* of the topics ML-DTM finds by computing the Jensen-Shannon (JS) divergence of every topic pair for each time slice for each language and averaging the divergences. Wang and McCallum (2006) used this method, though with KL divergence. It is desirable for the model to find topics that are as distinct as possible from each other.

We compare the diversity of the topics found by ML-DTM, trained as in the previous section, with the topics found by DTM. To make this comparison we train separate DTM models for each language in our two datasets, giving us four different models and compare the divergences of the topics found by these models with their ML-DTM counterparts. We use the Gensim implementation of DTM[4] where we set the chain variance to 0.1 and leave other parameters to be inferred during training. We train both ML-DTM and DTM on 10 time slices for 10 topics.

[3] https://www.kielipankki.fi/corpora/

[4] https://radimrehurek.com/gensim/models/ldaseqmodel.html

Time slice	# of NEs	PLTM	ML-DTM
Aug 1996	53	**0.880**	0.692
Sept 1996	65	0.876	**0.908**
Oct 1996	64	0.840	**0.885**

Table 3: Average cosine similarity of topic vectors for NEs over three time slices in DE-NEWS.

Time slice	# of NEs	PLTM	ML-DTM
Jan 2012	79	0.800	**0.896**
Feb 2012	71	**0.810**	0.796
Mar 2012	72	0.722	**0.745**

Table 4: Average cosine similarity of the vectors of NEs for three time slices in the YLE dataset.

6 Results and Discussion

Tables 3 and 4 show the average cosine similarity between NEs for each language in the DE-NEWS and YLE datasets, respectively. In the DE-NEWS data (Table 3), PLTM outperforms ML-DTM in the first time slice but ML-DTM performs better on the succeeding time slices. This is an encouraging result, considering that the parameters of ML-DTM at time slice t are estimated from adjacent time slices, adding a large degree of complexity to the model, whereas PLTM estimates parameters based on the current time slice only (PLTM has no concept of time).

For the YLE dataset (Table 4), ML-DTM shows an improvement in the first time and third slices and comparable performance in the second. The comparable nature of this dataset makes aligning NEs a more challenging task for both models. One way to improve performance on this task might be to use stricter criteria in aligning the dataset, such as pairing articles only if they were published on the same day or if they share more named entities.

We compare topic diversity of the topics found by DTM and ML-DTM. Tables 5 and 6 show the average JS divergence of every topic pair for five time slices in the DE-NEWS and YLE datasets, respectively. ML-DTM consistently learns more diverse topics than DTM for both datasets.

In Figure 4, we show the evolution of one topic found by ML-DTM trained on DE-NEWS. We show the top words of a topic about labor unions for the first eight months of the dataset. The English and German words are not exact translations of each other but we see similar or related words

Time slice	DTM English	ML-DTM English
Aug 1996	0.372	**0.655**
Sep 1996	0.368	**0.660**
Oct 1996	0.366	**0.657**
Nov 1996	0.365	**0.664**
Dec 1996	0.363	**0.650**
	DTM German	ML-DTM German
Aug 1996	0.315	**0.661**
Sep 1996	0.312	**0.670**
Oct 1996	0.310	**0.665**
Nov 1996	0.308	**0.638**
Dec 1996	0.306	**0.666**

Table 5: Topic diversity comparison between DTM and ML-DTM: average JS divergences of each topic pair for five months of the DE-NEWS dataset for English and German.

and NEs in each time slice. For instance, in August 1996 'employer' and 'arbeitgeber' both appear, as does 'einzelhandel' and 'retail'. In Sept 1996, 'kohl' is the top term for both languages (referring to former German chancellor Helmut Kohl). There are cases where German terms have no direct translation in English but an equivalent concept appears in the English topic. This is the case with 'lohnfortzahlung' (sick-leave pay) where the terms 'sick' and 'pay' appear on the English side; and 'steuerreform' (tax reform) where 'reform' appears on the English side as well.

A named entity, 'thyssen', appears in March 1997 in both languages but not in other months. This is because of an event that happened around mid-March where the German steel company Thyssen was being bought by competitor Krupp-Hoesch (also a top term in the German topic) prompting concerns about job losses[5].

Figure 5 shows the first six months of a topic about political news from the YLE dataset. The first two months has terms related to presidential elections. This refers to the Finnish presidential election in 2012, where rounds of voting took place in January and February 2012[6]. These time slices also mention the two candidates in the runoff election, Sauli Niinistö and

[5] https://www.nytimes.com/1997/03/19/business/krupp-hoesch-confirms-bid-of-8-billion-for-thyssen.html

[6] https://en.wikipedia.org/wiki/2012_Finnish_presidential_election

Time slice	DTM Finnish	ML-DTM Finnish
Jan 2012	0.332	**0.445**
Feb 2012	0.324	**0.465**
Mar 2012	0.322	**0.470**
Apr 2012	0.353	**0.498**
May 2012	0.357	**0.495**
	DTM Swedish	ML-DTM Swedish
Jan 2012	0.365	**0.480**
Feb 2012	0.360	**0.491**
Mar 2012	0.354	**0.497**
Apr 2012	0.388	**0.535**
May 2012	0.393	**0.537**

Table 6: Topic diversity comparison between DTM and ML-DTM: average JS divergences of each topic pair for five months of the YLE dataset for Finnish and Swedish.

Pekka Haavisto. Sauli Niinistö eventually won the election which explains why the next time slices ceases to mention Pekka Haavisto while 'niinistö' is still a prominent term. After March 2012, the topic stops talking about presidential elections and moves on to other political news. This gives us an insight into how the model can track significant events, such as high-profile elections, related to a topic. Another example is May 2012, where Greece ('kreikka' in Finnish, 'grekland' in Swedish) suddenly becomes a prominent term for both languages due to the Greek legislative elections which took place on 6 May 2012. The term 'syyria'/'syrien' appears in May and June, corresponding to the beginning of the Syrian Civil War.

Figure 6 shows the posterior probabilities of some terms related to the presidential elections ('niinistö'), Greece ('kreikka' or 'grekland') and Syria ('syyria' or 'syrien') in the political news topic for both languages. We see the rise and fall of the prominence of the terms according to their relevance in the news.

7 Conclusions and Future Work

In this paper we present a novel topic model, the *multilingual dynamic topic model* (ML-DTM), that combines dynamic topic modeling (DTM) and polylingual topic modeling (PLTM) to infer dynamic topics from aligned multilingual data. ML-DTM uses an extension of the DTM inference method of Bhadury et al. (2016) to aligned multilingual data.

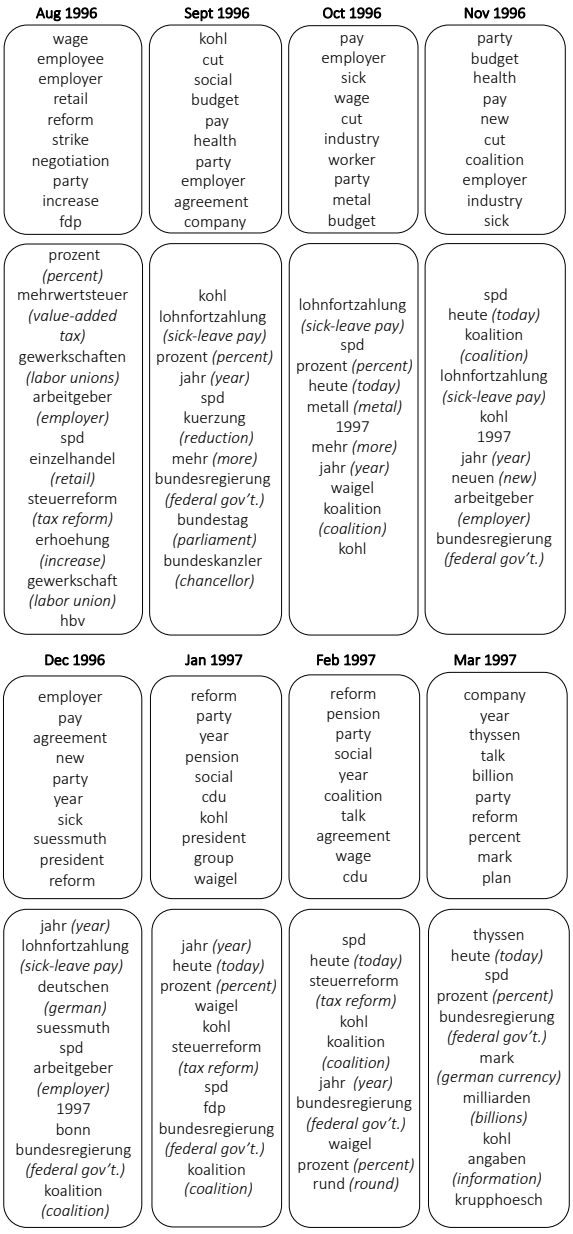


Figure 4: Top words of a topic concerning news about labor unions from the DE-NEWS dataset for English (top) and German (bottom) from Aug 1996 to March 1997. English translations of the German words excluding named entities are enclosed in parentheses.

We ran experiments on ML-DTM with parallel and comparable datasets. We compare cross-lingual topic alignment of PLTM and ML-DTM by evaluating the cosine similarities of topic vectors corresponding to named entity terms across languages for corresponding time slices. ML-DTM achieves similar performance to PLTM on DE-NEWS and the comparable dataset (YLE). We also demonstrate the ability of ML-DTM to detect

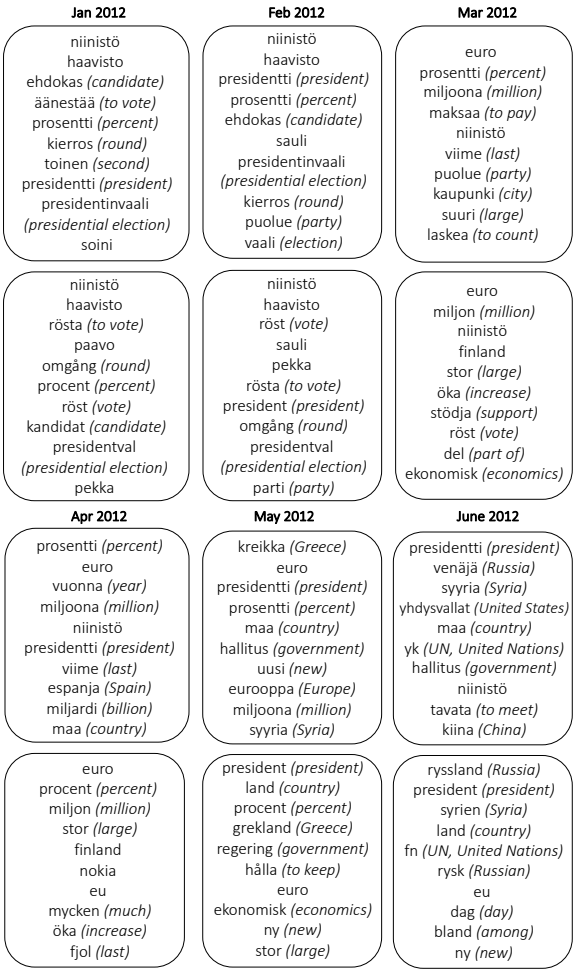


Figure 5: Top words of a topic on political news in Finland from the YLE dataset for Finnish (top) and Swedish (bottom) from Jan to June 2012. English translations of the words excluding named entities are enclosed in parentheses.

significant events regarding a topic through sudden changes in the prominent terms of the topic. This same method can also detect approximately when the event emerged and when it ended.

In a further experiment, we compared ML-DTM to the monolingual DTM, showing that ML-DTM achieves a consistently higher topic diversity within a single language.

We plan to run further experiments with ML-DTM using noisy datasets, such as historical news data where OCR errors might affect upstream tasks such as tokenization and lemmatization. We also plan to use named-entity recognition to improve our model such that named entities are treated as distinct items in the model's vocabulary, allowing us to track mentions of an entity across time slices and languages.

Historical news data covering a longer time span (several decades or more) would also enable us to study the changes in the use of words in a language and compare these changes with other languages. Historical news data from different regions would enable us to compare the way certain historical events were discussed in these places.

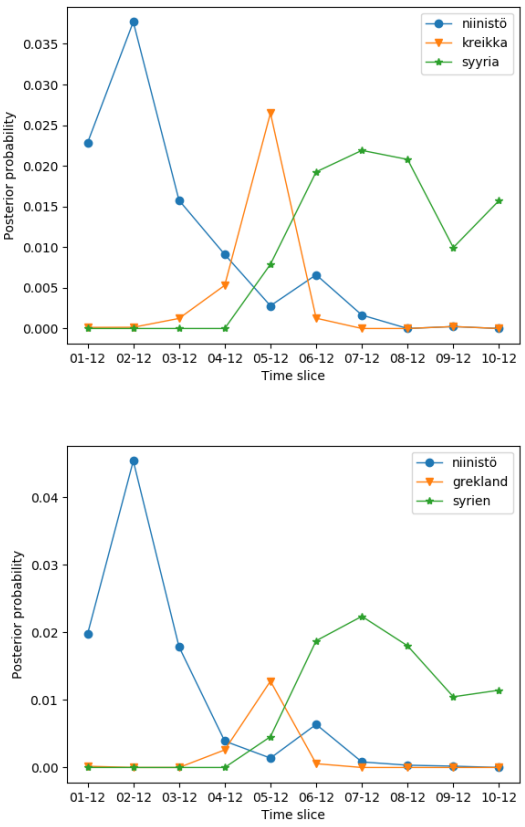


Figure 6: Posterior probabilities of salient terms in Finnish (top) and Swedish (bottom) related to events in the political news topic captured by ML-DTM from the YLE dataset.

Acknowledgements

This work has been supported by the European Union's Horizon 2020 research and innovation programme under grants 770299 (NewsEye) and 825153 (EMBEDDIA).

References

Arnab Bhadury, Jianfei Chen, Jun Zhu, and Shixia Liu. 2016. Scaling up dynamic topic models. In *Proceedings of the 25th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, pages 381–390.

David M Blei and John D Lafferty. 2006. Dynamic topic models. In *Proceedings of the 23rd interna-*

tional conference on Machine learning. ACM, pages 113–120.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3(Jan):993–1022.

Jordan Boyd-Graber and David M Blei. 2009. Multilingual topic models for unaligned text. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*. AUAI Press, pages 75–82.

Lea Frermann and Mirella Lapata. 2016. A bayesian model of diachronic meaning change. *Transactions of the Association for Computational Linguistics* 4:31–45.

Thomas L Griffiths and Mark Steyvers. 2004. Finding scientific topics. *Proceedings of the National academy of Sciences* 101(suppl 1):5228–5235.

David Hall, Daniel Jurafsky, and Christopher D Manning. 2008. Studying the history of ideas using topic models. In *Proceedings of the conference on empirical methods in natural language processing*. Association for Computational Linguistics, pages 363–371.

Jagadeesh Jagarlamudi and Hal Daumé. 2010. Extracting multilingual topics from unaligned comparable corpora. In *European Conference on Information Retrieval*. Springer, pages 444–456.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*. volume 5, pages 79–86.

Eetu Mäkelä. 2016. Las: an integrated language analysis tool for multiple languages. *The Journal of Open Source Software* 1.

David Mimno, Hanna M Wallach, Jason Naradowsky, David A Smith, and Andrew McCallum. 2009. Polylingual topic models. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*. Association for Computational Linguistics, pages 880–889.

Masao Utiyama and Hitoshi Isahara. 2003. Reliable measures for aligning japanese-english news articles and sentences. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*. Association for Computational Linguistics, pages 72–79.

Jurgen Van Gael and Xiaojin Zhu. 2007. Correlation clustering for crosslingual link detection. In *IJCAI*. pages 1744–1749.

Thuy Vu, Ai Ti Aw, and Min Zhang. 2009. Feature-based method for document alignment in comparable news corpora. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, pages 843–851.

Ivan Vulić, Wim De Smet, Jie Tang, and Marie-Francine Moens. 2015. Probabilistic topic modeling in multilingual settings: An overview of its methodology and applications. *Information Processing & Management* 51(1):111–147.

Chong Wang, David Blei, and David Heckerman. 2008. Continuous time dynamic topic models. In *Proceedings of the Twenty-Fourth Conference on Uncertainty in Artificial Intelligence*. AUAI Press, Arlington, Virginia, United States, UAI'08, pages 579–586. http://dl.acm.org/citation.cfm?id=3023476.3023545.

Xuerui Wang and Andrew McCallum. 2006. Topics over time: a non-markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pages 424–433.

Xing Wei, Jimeng Sun, and Xuerui Wang. 2007. Dynamic mixture models for multiple time-series. In *Ijcai*. volume 7, pages 2909–2914.

Max Welling and Yee W Teh. 2011. Bayesian learning via Stochastic Gradient Langevin Dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*. pages 681–688.

Tze-I Yang, Andrew Torget, and Rada Mihalcea. 2011. Topic modeling on historical newspapers. In *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*. pages 96–104.

E. Topic model training pipeline

Below is the definition of the TM training pipeline applied for the focused working groups, described in Section 3.

This is a Pimlico pipeline, where each block defines a module in the pipeline. The modules are run in turn, once their input (typically from earlier modules) is ready. This is an abstract pipeline, which contains the definition of most of the processing for all languages. It is instantiated for each language, with a correspondingly set `lang_code` variable.

For more details of Pimlico pipeline definitions, see the Pimlico documentation[15].

```
%%abstract
[vars]
dump_path=/wrk/group/newseye/corpora/demonstrator_data_dump/2020_01_27

[articles]
type=newseye.modules.input.demonstrator
path=%(dump_path)s/%(lang_code)s
# Create a small variant of the pipeline for testing
%%(small) limit=100

[years_input]
type=newseye.modules.input.demonstrator.years
path=%(dump_path)s/%(lang_code)s
%%(small) limit=100

# Store the years before we use them
# This is so we can debug if anything is going wrong before we use them in subsampling
[years]
type=pimlico.modules.corpora.store
input=years_input

# Consider replacing this with a better tokenizer for each language.
# Note that it might be better to use this very cautious, somewhat OCR robust tokenizer.
# Tokenization using a very simple tokenizer suitable for use on this noisy data
[tokenize]
type=newseye.modules.text_proc.ocr_tokenize
input=articles

# Get rid of single-character words
# Tokenization leaves lots: punctuation, some individual letters or other symbols
# They don't help with learning a topic model
[filter_short]
type=newseye.modules.text_proc.filter_short_tokens
input=tokenize
min_length=2
# Run together with the next one
filter=T

# Get rid of documents with very few words
```

[15] https://pimlico.readthedocs.io/

```
# These contribute little (and can even be harmful) to the LDA model training
[filter_short_docs]
type=newseye.modules.text_proc.filter_short_docs
input=filter_short
min_length=20

# Perform lemmatization on all the data using Eetu's LAS tool
[lemmatize]
type=newseye.modules.text_proc.las.lemmatize
input=filter_short_docs
locales=%(lang_code)s

# Subsample a number of docs for each year for training LDA
[subsampled_by_year_lda]
type=newseye.modules.data_prep.subsample_by_year
input_corpus=lemmatize
input_years=years_input
n=300

# Lets try a bigger corpus for training LDA
[subsampled_by_year_lda_big]
type=newseye.modules.data_prep.subsample_by_year
%%copy subsampled_by_year_lda
n=1000

# Subsample a number of docs for each year for training DTM
[subsampled_by_year_dtm]
type=newseye.modules.data_prep.subsample_by_year
input_corpus=lemmatize
input_years=years_input
n=100

# Build vocabulary
[vocab]
type=pimlico.modules.corpora.vocab_builder
input=lemmatize
threshold=30
limit=20k
oov=OOV
prune_at=100k
# Exclude words that appear in over 10% of documents:
# they're not going to tell the topic models anything useful
max_prop=0.1

############## LDA training ################
# Map words to IDs using the vocab
[ids_lda]
type=pimlico.modules.corpora.vocab_mapper
tie_alts=T
input_vocab=vocab
input_text=subsampled_by_year_lda
# Leave out any OOVs
```

```
oov=skip

# Train basic vanilla LDA with 20 topics
[lda20]
type=pimlico.modules.gensim.lda
tie_alts=T
input_corpus=ids_lda
input_vocab=vocab
num_topics=20
ignore_terms=OOV
tfidf=T
# If the corpus is small, it's best to do multiple passes
passes=20
multicore=T

# Now with 50 topics
[lda50]
type=pimlico.modules.gensim.lda
%%copy lda20
num_topics=50

# Now with 100 topics
[lda100]
type=pimlico.modules.gensim.lda
%%copy lda20
num_topics=100

# Train LDA models for the bigger subcorpora
# Map words to IDs using the vocab
[ids_lda_big]
type=pimlico.modules.corpora.vocab_mapper
tie_alts=T
input_vocab=vocab
input_text=subsampled_by_year_lda_big
# Leave out any OOVs
oov=skip

# Train basic vanilla LDA with 20 topics
[lda20_big]
type=pimlico.modules.gensim.lda
tie_alts=T
input_corpus=ids_lda_big
input_vocab=vocab
num_topics=20
ignore_terms=OOV
tfidf=T
# If the corpus is small, it's best to do multiple passes
passes=10
multicore=T

# Now with 50 topics
[lda50_big]
```

```
type=pimlico.modules.gensim.lda
%%copy lda20
num_topics=50

# Now with 100 topics
[lda100_big]
type=pimlico.modules.gensim.lda
%%copy lda20
num_topics=100

############## DTM training ################
# Map words to IDs using the vocab
[ids_dtm]
type=pimlico.modules.corpora.vocab_mapper
tie_alts=T
input_vocab=vocab
input_text=subsampled_by_year_dtm
# Leave out any OOVs
oov=skip

# Train DTM for 20 topics
[dtm20]
type=newseye.modules.topics.train.dtm
input_corpus=ids_dtm
input_vocab=vocab
num_topics=20
num_time_slices=%(time_slices)s
size_time_slice=100

# Now with 50 topics
[dtm50]
type=newseye.modules.topics.train.dtm
%%copy dtm20
num_topics=50

############## Inferring unseen documents ################
# Map words to IDs using the vocab
[ids_lda_all]
type=pimlico.modules.corpora.vocab_mapper
tie_alts=T
input_vocab=vocab
input_text=lemmatize
# Leave out any OOVs
oov=skip

# Analyse each document in the corpus using the trained model
[lda20_topic_vectors]
type=pimlico.modules.gensim.lda_doc_topics
input_model=lda20
input_corpus=ids_lda_all

[lda50_topic_vectors]
```

```
type=pimlico.modules.gensim.lda_doc_topics
%%copy lda20_topic_vectors
input_model=lda50

[lda100_topic_vectors]
type=pimlico.modules.gensim.lda_doc_topics
%%copy lda20_topic_vectors
input_model=lda100
```