Decoding 16th-Century Letters: From Topic Models to GPT-Based Keyword Mapping

Phillip Benjamin Ströbel
University of Zurich
phillip.stroebel@uzh.ch

Stefan Aderhold
FS Theologenbriefwechsel der AdW
stefan@aderhold-net.de

Ramona Roller
Universiteit Utrecht
r.i.roller@uu.nl

Abstract

Probabilistic topic models for categorising or exploring large text corpora are notoriously difficult to interpret. Making sense of them has thus justifiably been compared to "reading tea leaves." Involving humans in labelling topics consisting of words is feasible but time-consuming, especially if one infers many topics from a text collection. Moreover, it is a cognitively demanding task, and domain knowledge might be required depending on the text corpus. We thus examine how using a Large Language Model (LLM) offers support in text classification. We compare how the LLM summarises topics produced by Latent Dirichlet Allocation, Non-negative Matrix Factorisation and BERTopic. We investigate which topic modelling technique provides the best representations by applying these models to a 16th-century correspondence corpus in Latin and Early New High German and inferring keywords from the topics in a low-resource setting. We experiment with including domain knowledge in the form of already existing keyword lists. Our main findings are that the LLM alone provides usable topics already. However, guiding the LLM towards what is expected benefits the interpretability. We further want to highlight that using nouns and proper nouns only makes for good topic representations.

1 Introduction

In the realm of digital humanities and computational linguistics, probabilistic topic models have found wide application in categorising and exploring large text corpora (Meeks and Weingart, 2012; Sia and Duh, 2021; Schöch, 2021; Hodel et al., 2022). Especially one technique, i.e., Latent Dirichlet Allocation (LDA) (Blei et al., 2003), has established itself as a "quasi-standard" (Hodel et al., 2022, p. 186). However, models like LDA, essential for distilling and interpreting large datasets, yield results in the form of *bags-of-words* that are opaque and difficult to decipher, drawing comparisons to the esoteric art of "reading tea leaves" (Chang et al., 2009).

Incorporating human judgement has been used to refine topics directly during the model training of so-called *interactive topic models* (Hu et al., 2014), but rarely to label the bags of words to make them more interpretable. This is mainly because labelling topics is subjective (Alokaili, 2021), labour-intensive (Rüdiger et al., 2022), and demands considerable cognitive effort and domain-specific knowledge, particularly when multiple topics are derived from expansive text collections.

Thus, it has long been a desire to find labels for topics automatically, which has been achieved with varying degrees of accuracy (Mei et al., 2007; Lau et al., 2011; Kou et al., 2015). The advent of Large Language Models (LLMs) like OpenAI's GPT-4 (OpenAI, 2023) has introduced new possibilities in the field of text analytics. These models, equipped with advanced capabilities in natural language understanding, offer a promising avenue for leveraging information gathered through topic modelling techniques such as LDA, Non-negative Matrix Factorisation (NMF) (Lee and Seung, 1999; Ding et al., 2008), and BERTopic (Grootendorst, 2022). While these techniques still represent a document as a mixture of topics, LLMs can interpret the topic words instead of relying on human analysis. This paper explores the potential of LLMs to support and refine the process of text classification, particularly by leveraging their capacity to analyse and generate coherent and interpretable topic representations.

To evaluate the effectiveness of these topic modelling techniques, we examine how topics summarised by an LLM compare in a low-resource setting – specifically within the historical linguistics domain, focusing on a corpus of 16th-century correspondence written in Latin and Early New High German.

Proceedings of the 20th Conference on Natural Language Processing (KONVENS 2024), pages 209–221
September 10-13, 2024 ©2024 Association for Computational Linguistics

Letters tend to treat several topics so that topic modelling can prove a valuable approach for *distant reading* (Moretti, 2000). However, the (compared to other corpora) small number of tokens in a multilingual environment, together with a high number of topics, makes topic modelling and keyword assignment a difficult problem. This work aims to reveal how well LLMs can generate usable topics under constrained resource conditions.

Additionally, this study explores the integration of domain-specific knowledge, utilising preexisting keyword lists to guide the LLM towards more accurate topic generation. By concentrating on document representations consisting of nouns and proper nouns, we assess the quality of the topic representations produced and discuss how directed LLM support can enhance the interpretability of the model outputs. Although our findings highlight the standalone capabilities of LLMs in topic generation and underscore the benefits of incorporating guided input to improve both the clarity and relevance of topic modelling, we also encountered difficulties, which we explain below.

2 Recent Research

Topic Modelling for Correspondence Data Topic modelling has been applied to all sorts of data, including correspondences. Wittek and Ravenek (2011) applied LDA, among other methods, to 17th-century scholarly correspondences. Their multilingual corpus comprised Dutch, English, French, German, Greek, Italian and Latin letters, accumulating over 7 million tokens. They trained separate topic models for the most common languages (Dutch, French, and Latin).

Topic Modelling in Low-Resource Scenarios Hao et al. (2018) contributed to the evaluation of topic models, especially in low-resource settings. They experimented with parallel data and normalised pointwise mutual information scores to measure topic coherence and train an estimator to predict topic coherence in the case of low-resource languages.

Sia and Duh (2021) investigated how to improve LDA for low-resource languages directly. Their research introduces a method that automatically balances externally trained continuous representations with traditional co-occurrence count-based statistics tailored to each word and topic. This approach adapts to variations in topic numbers and embedding dimensions without extra tuning, enhancing existing methods.

Keywords for Topic Modelling Jagarlamudi et al. (2012) already incorporated lexical priors to guide topic models to infer topics that are relevant to the user. They provided the algorithm with sets of seed words which, according to their view, represented a topic, thereby influencing the word-topic and document-topic distributions.

This approach was taken one step further by Eshima et al. (2024), who proposed the keyword-assisted topic model. In contrast to defining sets of seed words, this approach uses a list of keywords, which the authors make part of the data generation process and hence influence word distribution of the topics.

Using LLMs for Topic Modelling Rijcken et al. (2023) applied LDA and a fuzzy clustering-based topic modelling algorithm (Rijcken et al., 2021) to clinical notes from the psychiatric domain and had both human experts and ChatGPT produce summaries of the topics. In the human evaluation which subsequently compared both summaries, they found that only about half the summaries generated by ChatGPT were useful.

LLMs have recently also been used to evaluate topics directly. Stammbach et al. (2023) show that LLMs correlate well with human ratings on coherence tasks, whereas identifying intruders still poses challenges to LLMs.

Pham et al. (2024) directly use GPT for topic modelling and compare against BERTopic, LDA and seeded NMF. They provide GPT with a list of keywords, or topics, and let GPT infer the topics for texts from Wikipedia and bills from the U.S. Congress. We will also use GPT directly to generate keywords from texts, but make amendments to the methods proposed by Pham et al. (see Section 4.5).

3 Data

3.1 The *Heinrich Bullinger Briefwechsel (HBBW)*

The comprehensive collection of approximately 12,000 surviving letters from Swiss Reformer Heinrich Bullinger (1504–1575) provides not just insights but a crucial connection to the complex network of relationships Bullinger maintained with intellectuals, theologians, monarchs, and other influential figures throughout Europe during the Reformation period (Campi, 2004). Written predominantly in Latin – the dominant *lingua franca* of that era – these letters also contain a notable amount of Early New High German (ENHG).[1] The initiative, *Bullinger Digital*,[2] has made this diverse, multilingual legacy accessible and interactive using digital curation techniques.

Of the 12,000 letters in the HBBW, 3,000 have been edited in 20 volumes already (Gäbler et al., 1974–2020), and another 5,400 have been transcribed. In these letters, the authors wrote about various theological and reformatory matters and issues in everyday life, like illness, marriage, and food. Topic modelling, in connection with keyword assignment, can thus help users to "distant read" the correspondence and to get an overview of the governing themes in the corpus.

We downloaded the data preprocessed by Ströbel et al. (2024) in TEI-XML format from the open access GitHub repository.[3]

3.2 The *Theologenbriefwechsel*

We obtained data from the so-called *Theologenbriefwechsel im Südwesten des Reichs in der Frühen Neuzeit (1550-1620)* (Strohm, 2017).[4] The Theologenbriefwechsel is a research project that focuses on gathering, accessing, and partially publishing the letters of key theologians and church leaders from the Electoral Palatinate, Württemberg, and Strasbourg between 1550 and 1620. This effort aims to understand the process of confessionalisation and its impacts during the early modern period. By analysing these letters, not just from individual exchanges but across specific groups and regions during this time, the project helps reveal broader networks and patterns, highlighting the significant role of theologians in shaping religious confessions.

3.3 Data Overview and Preprocessing

Accumulating the HBBW and Theologenbriefwechsel letters leads to a corpus of 10,319 letters, 1,731 of which stem from the Theologenbriefwechsel. Since the HBBW data was already split into sentences and each sentence had received a language label, we extracted the Latin and ENHG sentences from the Bullinger correspondence. For the Theologenbriefwechsel, we split the text into sentences and tokens using the *Classical Language Toolkit*'s (Johnson et al., 2021) sentence tokeniser. Subsequently, we determined the language with a language identifier trained to distinguish between Latin and ENHG (Volk et al., 2022). For the HBBW data, we only applied the sentence tokeniser. Summarising all tokens of the HBBW and the Theologenbriefwechsel yields a corpus of 5,630,039 tokens, 4,060,754 (72.13%) of which are in Latin and 1,569,285 (27.87%) are in ENHG.

A common further preprocessing step for topic modelling is stopword removal (Hodel et al., 2022) and the limitation of the vocabulary to, e.g., nouns. We decided to focus on nouns and proper nouns only,[5] which required Part-of-Speech (POS) tagging. For Latin, we employed spaCy's *LatinCy* (Burns, 2023) and filtered the Latin texts for words with NOUN and PROPN tags. We extracted the lemmas for words with NOUNs and lowercased all of them.

In the case of ENHG, there have been attempts at POS tagging. Demske et al. (2014) reported accuracies between 69% and 75% with the TnT Tagger (Brants, 2000). Barteld et al. (2018) experimented with different POS taggers for Middle High and Middle Low German and reached accuracies of 85.95% and 86.37%, respectively. Since we were dealing with ENHG, we trained our own spaCy[6] tokeniser along the lines of Burns (2023). As a base language model, we used `bert-base-german-cased`.[7] We took the *Referenzkorpus Frühneuhochdeutsch* as training data (Wegera et al., 2021),[8] converted the CorA-XML files to CoNLL-U format, mapped the tagset of the Referenzkorpus to UPOS tags, and used almost 2.5 million tokens for training and roughly 300k tokens for development and testing each. With an initial learning rate of 0.00005 with 500 warmup

[1] The collection also letters in Greek, Hebrew, English, Italian, and French.

[2] See https://www.bullinger-digital.ch.

[3] See https://github.com/bullinger-digital/bullinger-korpus-tei.

[4] See https://thbw.hadw-bw.de.

[5] We consider nouns and proper nouns to be the main carriers of meaning. Moreover, since our task is to map topics to keywords, which are almost always (proper) nouns, we decided to employ this rather radical preprocessing step. However, we admit that at least adjectives and verbs can also carry meaning that could help describe topics. We will address the consequences of different preprocessing steps in future research. However, work on this topic has already shown that focusing on nouns only is beneficial for topic coherence (cf. Martin and Johnson (2015)).

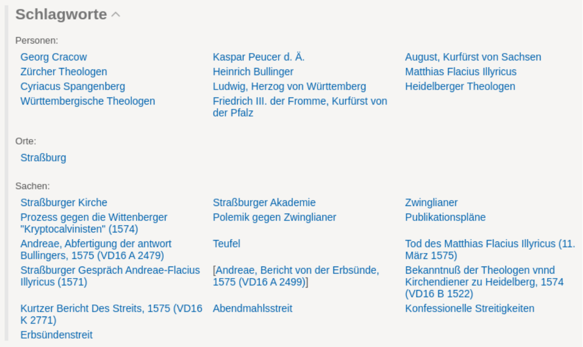
[6] See https://spacy.io.

[7] See https://huggingface.co/google-bert/bert-base-german-cased.

[8] We ignored the part from the Universität Potsdam since it did not contain lemmas.

steps and a total number of 20,000 trained steps (with early stopping), our POS tagger reached an accuracy of 54.39% on the test set,[9] while lemmatisation accuracy was considerably lower at 47.48%. This is due to the high number of types (254,374). Because of the low success rate of mapping surface forms to lemmas, we decided not to lemmatise the ENHG tokens but still only extracted words tagged as NOUN or PROPN and lowercased them. We have not investigated the low accuracy rates but plan to do so in future research endeavours. Still, we need to be aware that using the words identified as PROPN or NOUN and the corresponding word types instead of the lemmas makes inferring topics more difficult and will probably lead to worse results when compared to Latin.

Limiting ourselves to (proper) nouns only reduced the number of tokens from 5.6 million to 1.1 million, which means we are operating with 20.61% of the corpus. In addition to filtering (proper) nouns, extracting lemmas (for Latin only) and lowercasing, we further filtered out stopwords with a list of 657 words. We compiled and extended this list based on the topic modelling results: should certain words considered stopwords occur frequently among the indicative topic words, we included these words in this list. E.g., we added the ENHG word *wyr* (EN *we*) that was sometimes tagged as NOUN, as well as the Latin word *quid* (EN *this*). We also included words that occurred too often in the topics, like different versions of the proper names Heinrich (*heinrich*, *hainricus*, *hainrico*), Bullinger (*bullinger*, *bullingero*, *bullingerus*), and Zurich (*zürich*, *zurych*, *zürych*).



Schlagworte ^

Personen:

Georg Cracow
Kaspar Peucer d. Ä.
August, Kurfürst von Sachsen
Zürcher Theologen
Heinrich Bullinger
Matthias Flacius Illyricus
Cyriacus Spangenberg
Ludwig, Herzog von Württemberg
Heidelberger Theologen
Württembergische Theologen
Friedrich III. der Fromme, Kurfürst von der Pfalz

Orte:

Straßburg

Sachen:

Straßburger Kirche
Straßburger Akademie
Zwinglianer
Prozess gegen die Wittenberger "Kryptocalvinisten" (1574)
Polemik gegen Zwinglianer
Publikationspläne
Andreae, Abfertigung der antwort Bullingers, 1575 (VD16 A 2479)
Teufel
Tod des Matthias Flacius Illyricus (11. März 1575)
Straßburger Gespräch Andreae-Flacius Illyricus (1571)
[Andreae, Bericht von der Erbsünde, 1575 (VD16 A 2499)]
Bekanntnuß der Theologen vnnd Kirchendiener zu Heidelberg, 1574 (VD16 B 1522)
Kurtzer Bericht Des Streits, 1575 (VD16 K 2771)
Abendmahlsstreit
Konfessionelle Streitigkeiten
Erbsündenstreit

Figure 1: Example of keywords on the platform of the Theologenbriefwechsel. The keywords are divided in *Personen* (EN *persons*), *Orte* (EN locations), and *Sachen* (EN *matters*). Example is taken from a letter from Jakob Andrea to Johannes Marbach on May 25, 1575 (See https://thbw.hadw-bw.de/brief/21212).

3.4 Keyword Lists

The Theologenbriefwechsel has been manually annotated with keywords. During the course of the project, the keyword catalogue grew to contain over 18k keywords. The keywords are organised hierarchically. As the example in Figure 1 shows, we find very specific keywords like *Straßburger Gespräch Andreae-Flacius Illyricus (1571)* and more general ones like *Teufel* (EN *devil*) or *Abendmahlstreit* (EN *controversy about the Sacrament of the Lord's Supper*). The keyword *Abendmahlstreit* is embedded as follows (top-down): *Streit* (EN *dispute*) → *Streitigkeiten* (EN *conflict*) → *Abendmahlstreit*. The top level contains 339 keywords. This did not seem practical for mapping purposes, especially since we plan to offer a keyword filtering option on the *Bullinger Digital* platform. So we reduced the top-level keyword list to the 53 most important ones (we call this list *meta-topic list*) based on a subjective assessment.[10] Still, we were also interested in whether it is possible to map more fine-grained keywords to topic words, so we included two further keywords for each sub-topic under a meta-topic. For example, the meta-topic about the controversy about the Sacrament of the Lord's Supper contains three further sub-topics on the next level. We took two further keywords from these and added them to the list. E.g., the third sub-topic lists *Brot und Wein als sakramentliche Zeichen* (EN *bread and wine as sacramental symbols*). This leads to a sub-topic list of 273 keywords.

4 Experiments

See Figure 3 for an overview of our experimental setup. We first trained a topic model with BERTopic, which, based on the inherent clustering algorithm, indicated the number of topics present in the corpus. Taking the inferred number of topics from BERTopic, we further trained topic models using LDA and NMF. We then let GPT-4 summarise the topic words into keywords in three ways: 1) on its own, 2) with the meta-topic list, and 3) with the sub-topic list. We then automatically evaluate the keywords generated for each method used to produce the topic model. A second evaluation

[9] The highest accuracy on the development set during training was 74.62%. The low accuracy on the test set hints at the high variability of the data.

[10] In hindsight, and this is what we will do in the future, it would have been better to make this assessment based on the actual distribution of keywords in the Theologenbriefwechsel.

takes 50 letters at random and lets GPT infer the topics based on the preprocessed letter texts. We then compare the keywords generated with each method against each other. In the following, we provide further details about the generation of the topic models.

4.1 BERTopic

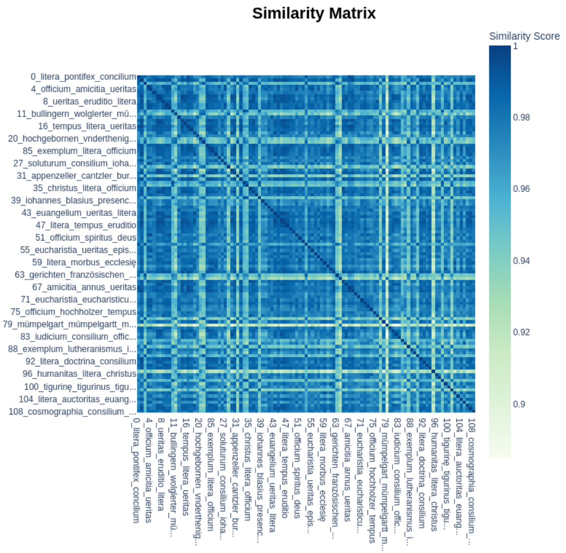


Figure 2: Heatmap of topic similarities by applying cosine similarities through the topic embeddings.

BERTopic is set up as modular architecture, which consists of 5 steps: 1) embed the documents, 2) apply dimensionality reduction, 3) cluster the documents, 4) represent them as bag-of-words, and 5) find the topic words. To embed the documents, we used the text embedding model `multilingual-e5-large`[11] (Wang et al., 2024). This multilingual model has also been trained on Latin and German, which is closest to ENHG.[12] For dimensionality reduction, we used the default UMAP algorithm (McInnes et al., 2018). However, running BERTopic with the standard settings leads to a very limited number of inferred topics (between 10 and 20). Given that we already have 53 items in the meta-topic list (and 273 keywords in the sub-topic list), we changed the parameter `n_neighbors` to 5.[13] Lowering this parameter causes the algorithm to focus on more local structures, leading to more clusters inferred in the next step. Due to the stochastic nature of UMAP, using `n_neighbors = 5` led to topic numbers between 100 and 168 for our texts. The final run we evaluate in this paper inferred 109 topics.[14] We then used the default clustering algorithm (HDBSCAN (Campello et al., 2013)) with its default settings.[15] A count vectoriser then represents each cluster as bag-of-words. We provided the vectoriser with our stopword list to filter out unwanted words (see Section 3.3). Finally, we used the class TF-IDF vectoriser to infer the topic words.

Inspecting the topic model reveals that many topics are similar (see Figure 2). The dissimilarities in the heatmap mainly stem from the low similarity scores when Latin topics are compared to ENHG topics. Comparing topics to each other, we see that topics 2 and 40 in Table 1 are very different from each other, 2 being about the sacraments and 49 most probably about war. Topic 60, on the other hand, contains words that we find in topics 2 and 49. Other words in this topic hint at the fact that it could be about illness, but this example shows the difficulty BERTopic has in finding different topics and also foreshadows that this could be problematic when automatically inferring topics. The same is true for topics 64 (matters of law), 102 (sin), and 34, which is content-wise closer to 102 but also contains elements of 102 (though not explicitly).

4.2 Latent Dirichlet Allocation and Non-negative Matrix Factorisation

To infer topics with LDA and NMF, we used the *gensim* framework (Řehůřek and Sojka, 2010). The further processing consisted of converting the preprocessed texts to a gensim corpus and filtering out extremes. This means we excluded tokens that occurred in less than 20 documents and in more than 10% of the documents. This reduced the vocabulary from 95k to 4,600 tokens and the effective corpus size from 1.1 million tokens to 416k. We then used 30 passes for both topic modelling techniques to infer 109 topics (the number we obtained

[11] Available on the Hugging Face model hub: https://huggingface.co/intfloat/multilingual-e5-large.

[12] To the best of our knowledge, the only model available is *Turmbücher LM* by the University of Bern (trained on 16th-century texts and available on the Hugging Face model hub: https://huggingface.co/dh-unibe/turmbuecher-lm-v1) However, this model cannot embed Latin texts.

[13] We kept the rest of the parameters at their default values: `n_components = 5`, `min_dist = 0.0`, `metric = 'cosine'`.

[14] Setting the `random_state` parameter fixes the results. However, we did not evaluate the different runs against each other. Again, we would approach matters differently in the future, setting `random_state` from the beginning to ensure a better and reproducible experimental setting.

[15] `min_cluster_size = 15`, `metric = 'euclidean'`, `cluster_selection_method = 'eom'`, `prediction_data = True`

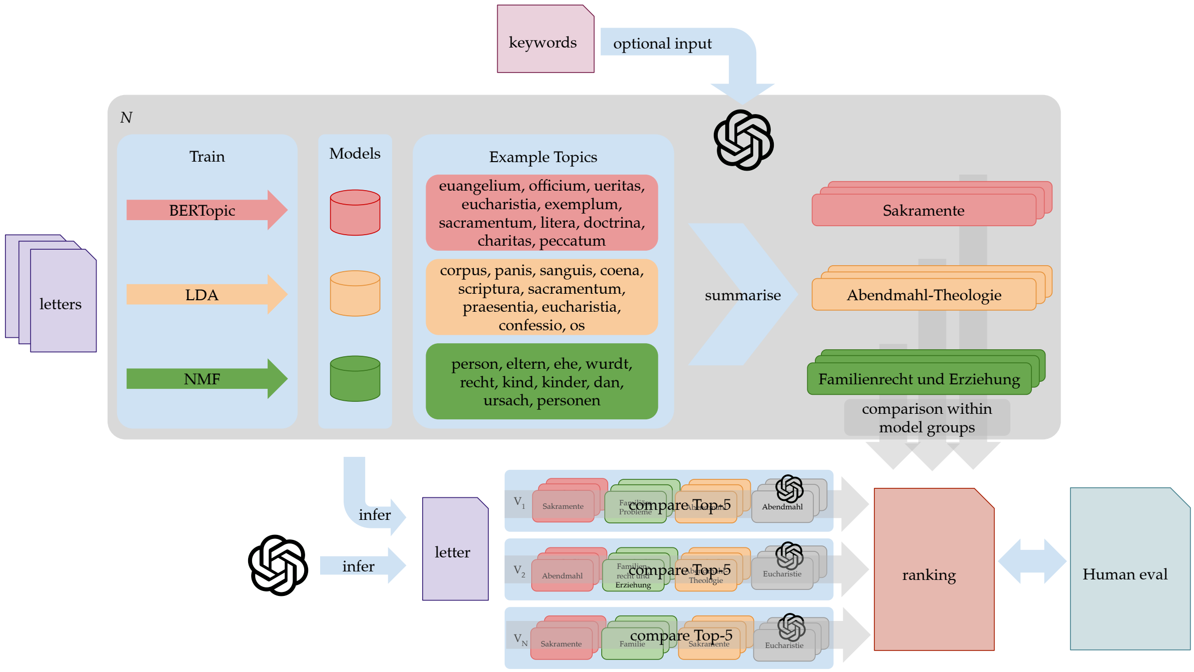

keywords
optional input
N
Train
Models
Example Topics
BERTopic
LDA
NMF
letters
euangelium, officium, ueritas, eucharistia, exemplum, sacramentum, litera, doctrina, charitas, peccatum
corpus, panis, sanguis, coena, scriptura, sacramentum, praesentia, eucharistia, confessio, os
person, eltern, ehe, wurdt, recht, kind, kinder, dan, ursach, personen
summarise
Sakramente
Abendmahl-Theologie
Familienrecht und Erziehung
comparison within model groups
infer
infer
letter
V1
Sakramente
Familiäre Probleme
Abendmahl
compare Top-5
V2
Abendmahl
Familienrecht und Erziehung
Abendmahl-Theologie
Eucharistie
VN
Sakramente
Familie
Sakramente
Eucharistie
ranking
Human eval

Figure 3: Workflow overview. We first train topic models with different algorithms on the letters. Each model represents the topics differently, but by default, with 10 words. We then used GPT to summarise the topic words into one keyword. For LDA and NMF, we also experimented with summarising the topics using 20 and 30 words. Additionally, we provided GPT with two lists of keywords: the *meta-topic list* and the *sub-topic* list (see Section 3.4). We then evaluated for each combination within a topic modelling category (BERTopic, LDA, and NMF) how different the keywords or key phrases are using sentence transformers. For a test set of 50 letters, we generated keywords with GPT directly and compared to the ones inferred from topic words.

from BERTopic).

Table 2 shows two exemplary topics for each algorithm. We purposely chose similar topics for LDA and NMF. The two Latin topics 1 (LDA) and 39 (NMF) express the sacraments with very similar topic words. In the same way, the two ENHG topics 88 (LDA) and 64 (NMF) describe the debate about the Last Supper in a comparable style. We would, therefore, expect GPT to produce the same (or similar) keywords for those topics.

4.3 Inferring Topics with GPT

After training the three different topic models (BERTopic, LDA, and NMF), we want to test GPT's capability of inferring a keyword based on the topic words. For BERTopic, GPT uses 10 topic words per topic to solve this task.[16] For LDA and NMF, we expand the setting and let GPT generate a keyword based on 10, 20, and 30 topic words. We want to know whether more information, i.e., more topic words, helps GPT to produce better keywords. For example, for topic 64 of Table 2, the additional topic words *leibs, essen, wesen, paulus, nachtmal, worten, menschen, christo, todt, leyb* could steer GPT away from the rather general keyword *Abendmahl* to *Abendmahlsstreit*. The following two sections *Prompting A* and *Prompting B* further detail our methodology.

4.4 Prompting A – Keyword Mapping

We used the prompts shown in Table 3 to map topic words to keywords. The "Role" first defines the general role of GPT. The two different prompts represent our two different settings: The "GPT only" prompt asks GPT to generate a keyword (maximum 3 words) that best captures the meaning of the 10, 20, or 30 topic words. The "GPT with keyword list" provides the topic words and either the meta-topic or the sub-topic list, instructing GPT to choose a keyword from the list. With these two methods, we map the topic words to keywords.

[16] Although the documentation of BERTopic mentions that it is possible to infer more than 10 topic words with the `top_n_words` parameter, we did not manage to make this functionality work.

Topic #	Topic words
2	euangelium, officium, ueritas, eucharistia, exemplum, sacramentum, litera, doctrina, charitas, peccatum
49	exercitus, litera, communitas, philippus, uxor, legatus, bellum, frater, salus, iohannes
60	litera, morbus, ecclesię, tempus, euangelium, christus, spiritus, gratia, deus, caęlum
64	gerichten, französischen, gemeinden, handlung, pündten, rächt, baß, gmeinden, meinung, gewalt
102	sünden, frucht, menschen, kirchen, ergernuß, gott, mentschen, oberkeit, gotsforcht, namen
34	rechtfertigung, confeßion, gerechtigkeit, vnderthenigkeit, sünden, glauben, urchlaucht, meinung, gerechtigkaitt, wirtenbergh

Table 1: Example topics from BERTopic. Topics 2, 49 and 60 are Latin, 64, 102, and 34 ENHG.

Topic # LDA	Topic words
1	corpus, panis, sanguis, coena, scriptura, sacramentum, praesentia, eucharistia, confessio, os
88	christi, leib, blut, meinung, leibs, wein, zwinglianer, wesen, mensch, bluts
Topic # NMF	**Topic words**
39	corpus, panis, coena, sanguis, uictima, manducatio, figure, peccatum, coenae, os
64	leib, christi, but, will, wurdt, wein, brott, glauben, wort, meinung

Table 2: Example topics from LDA and NMF.

4.5 Prompting B – Direct Keyword Generation

In addition to keyword mappings based on topic words, we prompted GPT to generate a keyword for 50 randomly chosen letters. We only slightly adapted the prompt already presented in Table 3, instructing GPT to generate 5 keywords per letter. We used the same preprocessed letters as for BERTopic. Similar to the setting in the previous section, we also had two runs during which GPT had access to the meta-topic and sub-topic lists.

4.6 Evaluation and Results

We aimed at an automatic evaluation of the topic-words-to-keyword-mappings. For the setting *Prompting A* described in Section 4.4, we compared the generated keywords by embedding them with `multilingual-t5-large` (see Section 4.1) and computing the cosine similarities for each algorithm. E.g., we compared the keyword generated by GPT using LDA's 10-word-topic *Kirchliche Gesetzgebung* (EN *church legislation*) to the keyword inferred from LDA's 20-word-topic *Kirchenrecht und Besitzverhältnisse* (EN *canon law and ownership structure*), obtaining a cosine similarity of 0.916. Perfect matches resulted in a cosine similarity of 1, while the lowest score of 0.726 was a comparison of NMF's 10-word-topic keyword *Osmanisches Heer* (EN *Ottoman army*) to NMF's 20-word-topic keyword *Glaube* (EN *faith*) (both times generated with the sub-topic list). We then averaged the similarity scores over the 109 topics for each comparison and obtained the results in Table 4 in Appendix A.

For the experiment in *Prompting B*, we took the five keywords generated by GPT based on the preprocessed letter texts and compared them to the top 5 topics the three methods BERTopic, LDA, and NMF have inferred from the texts. It is sometimes possible that one of these algorithms has only attributed one topic to a letter. We still compared this one topic in the form of its keyword to the 5 GPT-generated keywords. To compute the cosine similarity between the GPT-generated keywords and the inferred keywords based on the topic words of BERTopic, LDA, and NMF, we concatenated the keywords with commata and embedded them with `multilingual-t5-large`. E.g., these are the keywords GPT inferred for the letter by Hieronymus Zanchi to Thomas Erastus in the year 1570:[17] *Thomas Erastus, Gott und Teufel, Exorzismus und Dämonen, Hexen und Zauberer, Augustinus und Thomas Aquinas*. The following are the top-5 keywords inferred by GPT based on 10 topic topics words by LDA: *Kirchliche Kontroversen, Biblische Kommentararbeit, Finanzielle Kirchenangelegenheiten, Familienrecht, Stuttgarter Prädikanten-Korrespondenz*. We observe a rather low (compared to other scores) similarity score of 0.812. In-

[17] See https://thbw.hadw-bw.de/brief/19786.

Role	Prompt GPT only	Prompt GPT with keyword list
You are a historian and an expert of 16th-century correspondence. You are presented with topics from letters from the correspondence of the 16th century and have to find a keyword or keyphrase that best matches the topic words. The correspondence discusses not only the reformation, but also various other religious topics and everyday life situations.	For the following topic words in Latin or Early New High German separated by '-', find one German keyword or keyphrase (maximum 3 words) that captures the overall meaning best {}. Be more specific than 'Theologie' or 'Reformation'. Only output the keyword. Don't explain your decision. Don't translate.	For the following topic words in Latin or Early New High German separated by '-': {}, choose one keyword or keyphrase from the following list where keywords are separated by '|': {}. Choose the keyword that best summarises the topic words. Don't explain your decision. Don't translate.

Table 3: Role and prompts used for mapping topic words to a keyword. The "Role" primes GPT for the task. The "Prompt GPT only" replaces the curly brackets with the topic words and lets GPT define a keyword itself. The "Prompt GPT with keyword list" replaces the second set of curly brackets with a list of keywords to choose from.

deed, there is no overlap between the keywords. If we look at the keywords the Theologenbriefwechsel has attributed to this letter, namely *Todesstrafe, Aberglaube, Astrologie, Auspizien, Nekromantie, Wahrsagegeist, Teufel, Dämonen, Hexerei, Ex 20,3, Dtn 5,7, Gen 35,2, Jos 24,16, Dtn 18,9-13, Lev 19,26, Lev 20,6, Lev 20,27*, we see that GPT manages to generate several keywords that also occur in the Theologenbriefwechsel based on the letter text alone. The keywords inferred by GPT based on topic words are, in that sense, less precise, although the keyword *Biblische Kommentararbeit* (EN *Bible commentaries*) reflects many Bible quotes present as keywords in the Theologenbriefwechsel.

5 Discussion

Our comparison focuses on the differences in the keywords generated from topic words. The big picture as presented in Table 4 shows that a certain guidance with the help of meta-topic and subtopic lists results in more consistent mappings than letting GPT imagining keywords on its own. Although this is somehow expected, it shows that we can steer GPT towards generating keywords based on existing catalogues, which is essential not only for the GLAM[18] sector but also for projects that want to offer their users additional search filters.

Another general trend seems to be that the more topic words GPT has at its disposal to generate a keyword, the more closely related the keywords are (see, e.g., the similarity score of 0.896 for "lda_10 vs. lda_20" and the one of 0.911 for "lda_20 vs. lda_30"). The low scores obtained when comparing keywords generated without lists to keywords generated with lists are again to be expected. However, the fact that they are not lower hints at a certain closeness. E.g., the GPT-generated keyword from LDA's 30-topic-words version for topic 66 is *Reichstag zu Augsburg*[19] and the keyword generated from LDA's 10-topic-words with the meta-topic list *Obrigkeit* (EN *lords*, or *authorities*) are related since the lords participated at the Reichstag.

Interestingly, as concerns the number of, e.g., meta-topics matched to topic words, we observed that the more topic words GPT has seen to choose a keyword from the meta-topic list, the fewer it uses in total. To be more concrete, for our LDA topic model, GPT assigned 26, 24, and 23 keywords from our meta-topic list of 53 keywords when seeing 10, 20, and 30 topic words, respectively. The trend for NMF is similar, going down from 30 to 29 and finally to 28 assigned keywords. We do not want to generalise this finding by saying that more information leads to heavier generalisation since this is counter-intuitive. Still, the fact that GPT never uses all the keywords at its disposal deserves further investigation.

The discrepancy between LDA and NMF in the keyword mapping leads us to assume that NMF infers more distinguishable topics. This is also reflected by the similarity scores in Table 4, which are almost always higher than the LDA scores.

For the comparison of similarities of assigned keywords for our 50 test letters, Table 5 in Appendix A lets us conclude that GPT chooses very similar keywords from the meta-topic list when it needs to do this with the preprocessed letter as basis instead of the topic words.

In a first small-scale, manual evaluation of 26 of our 50 test set letters, we provided an expert with the keywords generated by GPT on 1) BERTopics topic words, 2) NMF with 30 topic words, 3) the

[18] Galleries, Libraries, Archives, Museums.

[19] an event in 1530.

preprocessed letters, and 4) LDA with 10 topic words. With 9 votes, the expert prefers keywords generated by GPT directly based on the preprocessed text. LDA with 10 topic words is in second place with 8 votes, while BERTopics and NMF received 6 and 3 votes, respectively.

6 Conclusion

We presented an analysis of GPT-generated keywords based on outputs of three topic models (BERTopic, Latent Dirichlet Allocation and Non-negative Matrix Factorisation) and a small set of 50 letters, both "unguided" and with the help of meta-topic and sub-topic lists drawn from the already keyworded Theologenbriefwechsel. We conclude that, based on cosine similarity, GPT produces similar keywords, and that similarity increases the more topic words it is provided with. Moreover, we notice that GPT chooses similar topics from the meta-topic and sub-topic lists, albeit it does not make use of all possible keywords. In future research, we plan to investigate this issue to make GPT to use the complete list.

The results presented here cannot yet be used for indexing purposes. We need further human evaluation to assess the suitability of the inferred keywords.

In terms of preprocessing, obtaining better results for the Part-of-Speech tagging and lemmatisation of ENGH texts could bring further improvements, such as employing ENHG embeddings for BERTopic.

We could show that inferring topics with a preprocessed letter version containing only (proper) nouns yield useful keywords, reducing processing costs and adding an additional twist to the findings of Pham et al. (2024). Future research should also focus on using existing summaries of the letters as input for topic models. This would also decrease the cost of paying solutions like GPT and allow for training topic models on more data. At the same time, this enables to circumvent the problem of low-resource languages. Lastly, we want to test other LLMs for their keyword mapping capabilities.

Limitations

Topic Model Interpretation in Low-Resource Settings The study categorises letter contents using topic models like LDA, NMF, and BERTopic. However, these models sometimes produce overlapping or ambiguous topics, leading to challenges in accurately interpreting and matching the results with specific keywords. Despite utilising thousands of letters, the training data remain limited compared to contemporary corpora (or corpora from later periods), especially given the multilingual nature of the dataset. This limitation affects the robustness of the models, particularly in generating meaningful and representative keywords.

Historical Context and Language Variability The 16th-century letters exhibit considerable linguistic variability, particularly in Latin and ENHG, which can result in inaccuracies in topic modelling and keyword generation. ENHG, in particular, suffers from a lack of comprehensive linguistic tools, causing errors in POS tagging and keyword extraction.

Preprocessing Challenges The preprocessing step of extracting only nouns and proper nouns affects the granularity of topics. While this approach reduces noise, it may overlook key verbs or adjectives that could provide deeper insight into specific topics.

Biases in LLMs The GPT-based model employed for keyword mapping is trained on a vast and unknown text collection scraped from the internet, which may introduce biases when analysing historical texts. These biases could result in anachronistic interpretations that do not accurately reflect the period's sentiments and intentions. Moreover, LLMs are not deterministic and might produce inconsistent results.

Evaluation metrics The automated evaluation metric used to assess the generated keywords' accuracy and coherence might not fully capture the subtleties of historical themes or provide a comprehensive measure of the quality of the generated topics. Although having tried to counter this limitation with a human evaluation by an expert, this evaluation is small and might be subjective. More human feedback will be needed in the future to make more substantial claims.

Ethical Considerations

Given the different norms reflected in the letters, cultural sensitivity is crucial to avoid imposing modern biases on historical content. Interpretation of the results from topic modelling should be accurate, with clear mechanisms for review and correction to prevent misrepresentation. Additionally, LLMs may inherit biases from their training data, so care must be taken to ensure unbiased interpretations. Finally, the study aims to positively impact historical scholarship by carefully considering how results could influence perceptions of the individuals and events depicted in the letters.

References

Areej N. A. Alokaili. 2021. *Representing Automatically Generated Topics*. Ph.D. thesis, University of Sheffield.

Fabian Barteld, Chris Biemann, and Heike Zinsmeister. 2018. Variations on the theme of variation: Dealing with spelling variation for finegrained POS tagging of historical texts. In *Proceedings of the 14th Conference on Natural Language Processing, KONVENS 2018, Vienna, Austria, September 19-21, 2018*, pages 202–212. Österreichische Akademie der Wissenschaften.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(1):993–1022.

Thorsten Brants. 2000. TnT – a statistical part-of-speech tagger. In *Sixth Applied Natural Language Processing Conference*, pages 224–231.

Patrick J. Burns. 2023. LatinCy: Synthetic trained pipelines for Latin NLP. *arXiv preprint arXiv:2305.04365*.

Ricardo J. G. B. Campello, Davoud Moulavi, and Joerg Sander. 2013. Density-based clustering based on hierarchical density estimates. In *Advances in Knowledge Discovery and Data Mining*, pages 160–172, Berlin, Heidelberg. Springer Berlin Heidelberg.

Emidio Campi. 2004. Heinrich Bullinger und seine Zeit. In Emidio Campi, editor, *Heinrich Bullinger und seine Zeit*, number 31 in Zwingliana, pages 7–35. Theologischer Verlag Zürich, Zürich.

Jonathan Chang, Sean Gerrish, Chong Wang, Jordan Boyd-Graber, and David Blei. 2009. Reading tea leaves: How humans interpret topic models. *Advances in neural information processing systems*, 22.

Ulrike Demske, Pavel Logacev, and Katrin Goldschmidt. 2014. POS-tagging historical corpora: The case of Early New High German. In *Proceedings of the thirteenth workshop on treebanks and linguistic theories (TLT 13)*, volume 2014, pages 103 – 112.

Chris Ding, Tao Li, and Wei Peng. 2008. On the equivalence between non-negative matrix factorization and probabilistic latent semantic indexing. *Computational Statistics & Data Analysis*, 52(8):3913–3927.

Shusei Eshima, Kosuke Imai, and Tomoya Sasaki. 2024. Keyword-assisted topic models. *American Journal of Political Science*, 68(2):730–750.

Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *Preprint*, arXiv:2203.05794.

Ulrich Gäbler, Endre Zsindley, Kurt Maeder, Matthias Senn, Kurt Jakob Rüetschi, Hans Ulrich Bächtold, Rainer Heinrich, Alexandra Kess, Christian Moser, Reinhard Bodenmann, Judith Steiniger, and Yvonne Häfner, editors. 1974–2020. *Heinrich Bullinger Briefwechsel*. Heinrich Bullinger Werke. Theologischer Verlag Zürich.

Shudong Hao, Jordan Boyd-Graber, and Michael J. Paul. 2018. Lessons from the Bible on modern topics: Low-resource multilingual topic model evaluation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1090–1100.

Tobias Hodel, Dennis Möbus, and Ina Serif. 2022. Von Inferenzen und Differenzen. Ein Vergleich von Topic-Modeling-Engines auf Grundlage historischer Korpora. In Selin Gerlek, Sarah Kissler, Thorben Mämecke, Dennis Möbus, Jennifer Eickelmann, Katrin Köppert, Peter Risthaus, and Florian Sprenger, editors, *Von Menschen und Maschinen: Mensch-Maschine-Interaktionen in digitalen Kulturen*, pages 185–209. Hagen University Press.

Yuening Hu, Jordan Boyd-Graber, Brianna Satinoff, and Alison Smith. 2014. Interactive topic modeling. *Machine learning*, 95:423–469.

Jagadeesh Jagarlamudi, Hal Daumé III, and Raghavendra Udupa. 2012. Incorporating lexical priors into topic models. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 204–213.

Kyle P. Johnson, Patrick J. Burns, John Stewart, Todd Cook, Clément Besnier, and William J. B. Mattingly. 2021. The Classical Language Toolkit: An NLP framework for pre-modern languages. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 20–29.

Wanqiu Kou, Fang Li, and Timothy Baldwin. 2015. Automatic labelling of topic models using word vectors and letter trigram vectors. In *Information Retrieval Technology*, pages 253–264.

Jey Han Lau, Karl Grieser, David Newman, and Timothy Baldwin. 2011. Automatic labelling of topic models. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1536–1545. Association for Computational Linguistics.

Daniel D. Lee and H. Sebastian Seung. 1999. Learning the parts of objects by non-negative matrix factorization. *nature*, 401(6755):788–791.

Fiona Martin and Mark Johnson. 2015. More Efficient Topic Modelling Through a Noun Only Approach. In *Proceedings of the Australasian Language Technology Association Workshop 2015*, pages 111–115.

Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

Elijah Meeks and Scott B Weingart. 2012. The digital humanities contribution to topic modeling. *Journal of Digital Humanities*, 2(1):1–6.

Qiaozhu Mei, Xuehua Shen, and ChengXiang Zhai. 2007. Automatic labeling of multinomial topic models. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '07, page 490–499, New York, NY, USA. Association for Computing Machinery.

Franco Moretti. 2000. Conjectures on World Literature. *New Left Review*, (1):54–68.

OpenAI. 2023. Gpt-4 technical report. Technical report, OpenAI.

Chau Minh Pham, Alexander Hoyle, Simeng Sun, Philip Resnik, and Mohit Iyyer. 2024. Topicgpt: A prompt-based topic modeling framework. *Preprint*, arXiv:2311.01449.

Emil Rijcken, Floortje Scheepers, Pablo Mosteiro, Kalliopi Zervanou, Marco Spruit, and Uzay Kaymak. 2021. A Comparative Study of Fuzzy Topic Models and LDA in terms of Interpretability. In *2021 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–8.

Emil Rijcken, Floortje Scheepers, Kalliopi Zervanou, Marco Spruit, Pablo Mosteiro, and Uzay Kaymak. 2023. Towards Interpreting Topic Models with ChatGPT. In *The 20th World Congress of the International Fuzzy Systems Association*.

Matthias Rüdiger, David Antons, Amol M. Joshi, and Torsten-Oliver Salge. 2022. Topic modeling revisited: New evidence on algorithm performance and quality metrics. *PLOS ONE*, 17(4):1–25.

Christof Schöch. 2021. Topic modeling genre: An exploration of french classical and enlightenment drama. *Digital Humanities quarterly*, 11.

Suzanna Sia and Kevin Duh. 2021. Adaptive mixed component LDA for low resource topic modeling. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2451–2469, Online. Association for Computational Linguistics.

Dominik Stammbach, Vilém Zouhar, Alexander Hoyle, Mrinmaya Sachan, and Elliott Ash. 2023. Revisiting automated topic model evaluation with large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9348–9357.

Phillip Benjamin Ströbel, Lukas Fischer, Raphael Müller, Patricia Scheurer, Bernard Schroffenegger, Benjamin Suter, and Martin Volk. 2024. Multilingual workflows in bullinger digital: Data curation for Latin and Early New High German. *Journal of Open Humanities Data*, 10(12):12.

Christoph Strohm. 2017. *Theologenbriefwechsel im Südwesten des Reichs in der Frühen Neuzeit (1550-1620): zur Relevanz eines Forschungsvorhabens*. Universitätsverlag Winter, Heidelberg.

Martin Volk, Lukas Fischer, Patricia Scheurer, Raphael Schwitter, Phillip Benjamin Ströbel, and Benjamin Suter. 2022. Nunc profana tractemus. Detecting code-switching in a large corpus of 16th century letters. In *Proceedings of LREC-2022*, Marseille. LREC.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Multilingual e5 text embeddings: A technical report. *arXiv preprint arXiv:2402.05672*.

Klaus-Peter Wegera, Hans-Joachim Solms, Ulrike Demske, and Stefanie Dipper. 2021. Referenzkorpus frühneuhochdeutsch (1350–1650), version 1.0.

Peter Wittek and Walter Ravenek. 2011. Supporting the exploration of a corpus of 17th-century scholarly correspondences by topic modeling. In *SDH 2011 Supporting Digital Humanities: Answering the unaskable*. University of Copenhagen.

Radim Řehůřek and Petr Sojka. 2010. Software framework for topic modelling with large corpora. In *Proceedings of LREC 2010 workshop New Challenges for NLP Frameworks*, pages 46–50, Valletta, Malta.

A Appendix A

BERTopic 1	**BERTopic 2**	**sim**	**LDA 1**	**LDA 2**	**sim**	**NMF 1**	**NMF 2**	**sim**
			lda_10	lda_20	0.896	nmf_10	nmf_20	0.899
			lda_10	lda_30	0.886	nmf_10	nmf_30	0.889
berttopic	berttopic_meta	0.824	lda_10	lda_meta_10	0.82	nmf_10	nmf_meta_10	0.837
			lda_10	lda_meta_20	0.817	nmf_10	nmf_meta_20	0.83
			lda_10	lda_meta_30	0.815	nmf_10	nmf_meta_30	0.827
			lda_10	lda_sub_10	0.823	nmf_10	nmf_sub_10	0.835
berttopic	berttopic_sub	0.829	lda_10	lda_sub_20	0.821	nmf_10	nmf_sub_20	0.828
			lda_10	lda_sub_30	0.82	nmf_10	nmf_sub_30	0.83
			lda_20	lda_30	0.911	nmf_20	nmf_30	0.918
			lda_20	lda_meta_10	0.82	nmf_20	nmf_meta_10	0.831
			lda_20	lda_meta_20	0.821	nmf_20	nmf_meta_20	0.829
			lda_20	lda_meta_30	0.817	nmf_20	nmf_meta_30	0.825
			lda_20	lda_sub_10	0.82	nmf_20	nmf_sub_10	0.83
			lda_20	lda_sub_20	0.822	nmf_20	nmf_sub_20	0.83
			lda_20	lda_sub_30	0.822	nmf_20	nmf_sub_30	0.829
			lda_30	lda_meta_10	0.822	nmf_30	nmf_meta_10	0.826
			lda_30	lda_meta_20	0.823	nmf_30	nmf_meta_20	0.821
			lda_30	lda_meta_30	0.822	nmf_30	nmf_meta_30	0.823
			lda_30	lda_sub_10	0.823	nmf_30	nmf_sub_10	0.821
			lda_30	lda_sub_20	0.825	nmf_30	nmf_sub_20	0.822
			lda_30	lda_sub_30	0.827	nmf_30	nmf_sub_30	0.826
			lda_meta_10	lda_meta_20	0.947	nmf_meta_10	nmf_meta_20	0.952
			lda_meta_10	lda_meta_30	0.93	nmf_meta_10	nmf_meta_30	0.938
berttopic_meta	berttopic_sub	**0.889**	lda_meta_10	lda_sub_10	0.862	nmf_meta_10	nmf_sub_10	0.875
			lda_meta_10	lda_sub_20	0.858	nmf_meta_10	nmf_sub_20	0.864
			lda_meta_10	lda_sub_30	0.853	nmf_meta_10	nmf_sub_30	0.87
			lda_meta_20	lda_meta_30	**0.952**	nmf_meta_20	nmf_meta_30	**0.957**
			lda_meta_20	lda_sub_10	0.846	nmf_meta_20	nmf_sub_10	0.867
			lda_meta_20	lda_sub_20	0.865	nmf_meta_20	nmf_sub_20	0.878
			lda_meta_20	lda_sub_30	0.861	nmf_meta_20	nmf_sub_30	0.874
			lda_meta_30	lda_sub_10	0.846	nmf_meta_30	nmf_sub_10	0.866
			lda_meta_30	lda_sub_20	0.854	nmf_meta_30	nmf_sub_20	0.864
			lda_meta_30	lda_sub_30	0.856	nmf_meta_30	nmf_sub_30	0.878
			lda_sub_10	lda_sub_20	0.903	nmf_sub_10	nmf_sub_20	0.917
			lda_sub_10	lda_sub_30	0.882	nmf_sub_10	nmf_sub_30	0.898
			lda_sub_20	lda_sub_30	0.94	nmf_sub_20	nmf_sub_30	0.931

Table 4: Average similarities of 109 GPT-generated keywords per topic model. The numbers behind the models indicate the number of topic words from which GPT inferred a keyword. If the model name contains “meta” or “sub”, GPT was given the respective meta- or sub-topic lists to choose the keyword from.

letter topics	**GPT**	**meta**	**sub**
bertopics	0.810	0.833	0.829
bertopics_meta	0.791	0.825	0.818
bertopics_sub	0.787	0.814	0.810
lda_10	0.851	0.861	0.862
lda_20	0.854	0.865	0.862
lda_30	0.854	0.868	0.864
lda_meta_10	0.859	0.909	0.897
lda_meta_20	0.860	**0.916**	**0.903**
lda_meta_30	0.858	0.914	0.901
lda_sub_10	0.852	0.878	0.881
lda_sub_20	0.854	0.889	0.889
lda_sub_30	0.848	0.880	0.884
nmf_10	0.847	0.859	0.857
nmf_20	0.847	0.863	0.860
nmf_30	0.847	0.859	0.856
nmf_meta_10	0.861	0.906	0.899
nmf_meta_20	**0.862**	0.905	0.898
nmf_meta_30	0.860	0.906	0.899
nmf_sub_10	0.856	0.879	0.880
nmf_sub_20	0.859	0.880	0.885
nmf_sub_30	0.855	0.877	0.882
AVG	0.846	0.875	0.872
STD	0.022	0.029	0.027

Table 5: Averaged similarities of 5 keywords produced by GPT on its own and with the help of the meta-topic and sub-topic lists and based on the preprocessed letter texts vs. the top-5 keywords generated from the topic words.