Topic Modeling on Podcast Short-Text Metadata

Francisco B. Valero, Marion Baranes, and Elena V. Epure(✉)

Deezer Research, 22-26, rue de Calais, 75009 Paris, France
research@deezer.com

Abstract. Podcasts have emerged as a massively consumed online content, notably due to wider accessibility of production means and scaled distribution through large streaming platforms. Categorization systems and information access technologies typically use topics as the primary way to organize or navigate podcast collections. However, annotating podcasts with topics is still quite problematic because the assigned editorial genres are broad, heterogeneous or misleading, or because of data challenges (e.g. short metadata text, noisy transcripts). Here, we assess the feasibility to discover relevant topics from podcast metadata, titles and descriptions, using topic modeling techniques for short text. We also propose a new strategy to leverage named entities (NEs), often present in podcast metadata, in a Non-negative Matrix Factorization (NMF) topic modeling framework. Our experiments on two existing datasets from Spotify and iTunes and Deezer, a new dataset from an online service providing a catalog of podcasts, show that our proposed document representation, NEiCE, leads to improved topic coherence over the baselines. We release the code for experimental reproducibility of the results (https://github.com/deezer/podcast-topic-modeling).

Keywords: Podcasts · Short-text · Topic modeling · Named entities

1 Introduction

Podcasts is an audio content listened to on-demand for educational, entertainment, or informational purposes. Known as the "spoken" version of blog posts, they have evolved towards a wide variety of formats (e.g. monologues, multi-party conversations, narratives) spanning a wide range of categories (e.g. business, true crime). Podcasts have been massively popularised in the recent years due to increased use of streaming platforms and availability of underlying technology for information access, recording and publishing [11,20,35]. As of summer 2021, the number of available podcasts in the world exceeds $2M$ and over $48M$ episodes [32]. Likewise, the podcast listening audience has grown massively: 50% of the American population has listened to at least a podcast in 2019 and over 32% have listened to podcasts monthly (compared to 14% in 2014) [25].

Given the sharp growing interest in podcasts, researchers and industry players have searched for more suitable ways to enable listeners to effectively navigate these overwhelming collections [3,8,20,35]. Topics are central to any of

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2022
M. Hagen et al. (Eds.): ECIR 2022, LNCS 13185, pp. 472–486, 2022.
https://doi.org/10.1007/978-3-030-99736-6_32

the adopted podcast information access technology such as automatic categorization, search engines or recommender systems. However, annotating podcasts with topics is still quite problematic. First, although podcast metadata entails topic-related genres, manually assigned by creators or providers, in reality these are often noisy and unreliable [11,28]: genres could be too heterogeneous or broad (e.g. Kids & Family includes both sleep meditation and parenting advice); and podcast providers may misleadingly label their shows with unrelated genres for broadening exposure. Second, using topic modeling has its limitations inherited from the input text associated with podcasts: 1) metadata, such as title or description, is typically a short text of varying quality; 2) the automatically transcribed speech is noisy having a high word-error-rate especially for NEs and requires the engagement of more resources [3,8,11].

In the current work, we revisit the feasibility of discovering relevant topics from podcast metadata, titles and descriptions, usually documented by podcast providers, creators, or editors. While previous work [3] found podcast metadata less promising for topic-informed search compared to when using podcast transcripts, we hypothesize that it could still be a very useful data source for topic modeling when exploited with appropriate technology. If proven feasible, topic modeling on podcast metadata can be a more economic alternative than automatically extracting and exploiting transcripts of a rapidly growing podcast corpus. Additionally, the discovery of topics directly from metadata offers many opportunities for improved podcast retrieval. Identifying podcast categories at different granularity levels could help editors evolve manually created podcast taxonomies and automatically annotate podcasts with these categories. The discovered topics could also support the consolidation of podcast knowledge graphs [2,11], recently exploited in recommendation, by adding new edges that capture topic-informed podcast similarity based on metadata.

First, we take advantage of advancements in topic modeling, and benchmark multiple algorithms designed for short text on three podcast datasets. Two of these datasets are public: one from Spotify [8] and one from iTunes [23]. We have built a third dataset using Deezer[1], an online service providing a large podcast catalog. This dataset is the largest with both titles and descriptions available at the podcast level. Second, we propose a strategy to leverage NEs, frequently present in podcast titles and descriptions, in a NMF topic modeling framework. As we can see in the following example: *Shields Up! Podcast: Join Chris and Nev as they talk about their favourite Star Trek episodes covering everything from TOS to Lower Decks*, the metadata contains multiple NEs regarding the name of the speakers (*Join Chris* and *Nev*), but also the podcast topic (*Star Trek*, *TOS* and *Lower Decks*). By injecting cues from NEs in topic modeling, we improve over state-of-the-art (SOTA) methods using plain word embeddings, and show that the data sparsity (very low co-occurrences of semantically related terms) due to short text can be further alleviated.

To sum up the contributions of this work are: a) the most extensive study to date of topic modeling on podcast metadata, covering popular SOTA algorithms

[1] https://www.deezer.com/us/.

for short text and datasets from major podcast streaming platforms; b) NEiCE, a new NE-informed document representation for topic modeling with NMF, as an extension of CluWords [31]—our approach improves topic coherence over baselines in most evaluated cases; c) a new podcast dataset entailing English-language titles and descriptions from Deezer, an online service providing a podcast catalog, that is the largest in terms of the number of podcasts/shows.

2 Related Work

Topic modeling on short text faces the challenge of severe data sparsity due to the nature of this type of input [7]. Short text, as it consists of only few words, can be ambiguous and noisy and, in general, has limited context. This means that pairs of words that are topic-related do not or rarely co-occur in the same contexts, leading to conventional topic modeling techniques such as LDA [6] to perform poorly. Various topic modeling techniques have been designed to address this issue. Models can be classified in four groups: pseudo-documents-based [24,38], probabilistic [15], neural [17,33], and NMF-based [29,31]. We further review each group and some representative models.

The principle of pseudo-documents is to aggregate connected short texts in longer documents, which are further used as input to conventional topic modeling [15]. Initial aggregation methods leveraged metadata such as hashtags in tweets [16]. However, this proved limiting for other types of short texts (e.g. search queries) and led to self-aggregation methods, able to aggregate using topic cues based on the corpus only [24,38]. An issue identified with this type of methods is overfitting [38]. Also, they appear overall less competitive than the other groups of topic modeling techniques for short text [29,31], discussed further.

The second group entailing probabilistic models is the most related to conventional topic modeling (LDA) that represents documents and topics as multinomial distributions over topics, respectively words. The adaptation of these models to short text is to assume that each document is sampled only from a single topic, thus restricting document-topic distribution to a mixture of unigrams [22,36,37]. GPU-DMM [15], an effective and fast model in this group, is based on Dirichlet Multinomial Mixture (DMM) model and uses a Generalized Pólya Urna (GPU) as a sampling process to promote topic-related words. The word association is estimated by exploiting pre-trained word embedding [19]. This allows to alleviate data sparsity as it extends the context to words that are semantically related but they do not necessarily co-occur in the same text.

The third group has become popular in the last years with the rise of deep learning. Neural topic modeling is based on Variational Auto-Encoders (VAE) [4,17,30,33]. Typically, an encoder such as a MultiLayer Perceptron (MLP) compresses the Bag-of-Words (BoW) document representation into a continuous vector. Then, a decoder reconstructs the document by generating words independently [4,17]. Negative sampling and Quantization Topic Model (NQTM) [33], the latest topic modeling technique on short texts brings two contributions which yielded the current SOTA results. The first is a new quantification method

applied to the encoder's output whose goal is to generate peakier distributions for decoding. The second is to replace the standard decoder with a negative sampling algorithm that proves better at discovering non-repetitive topics.

The NMF-based group learns topics by decomposing the term-document (BoW) matrix representation of the corpus into two low-rank matrices, one corresponding to document representations over topics and the other to topic representations over words [14]. Given the limited contextual information, the Semantics-assisted Non-negative Matrix Factorization (SeaNMF) model [29] adjusts NMF to short texts by integrating into it word-context semantic correlations learnt from the skip-gram view of the input corpus. In contrast to SeaNMF which focuses on the learning part, CluWords [31] enhances the corpus representation before being factorized with standard NMF. The matrix is obtained with a proposed custom TF-IDF strategy that exploits pre-trained word embeddings.

The existing works include in their benchmark, datasets consisting of question or news titles, web snippets, review comments, or tweets [9,15,29,31,33]. Podcast metadata compared to these datasets exhibits a much higher frequency of NEs, which we exploit with the goal to further address data sparsity. To our knowledge, we are the first to assess existing models on podcast metadata and to explicitly consider NE-related cues in short-text topic modeling.

3 Methods

The topic modeling algorithms we benchmark are GPU-DMM [15], NQTM [33], SeaNMF [29] and CluWords [31]. By noticing the high frequency of NEs in podcast titles and descriptions, we also include in the benchmark another standard NMF-based model for which we design a new NE-informed document representation as input. The underlying hypothesis is that NEs convey the main topic information. Thus, we propose to promote vocabulary words related to these NEs by associating them with pseudo-term frequencies as presented in Sect. 3.2. For this, but also to capture word-to-word topic relatedness shown beneficial against data sparsity, we use pre-trained word and NE embeddings [34].

Finally, the rationale behind choosing to explore NE promotion in a NMF framework is twofold. Compared to probabilistic models, NMF-based ones have yielded better results on short text [7,29,31]. Then, the integration of background NE and word information in NMF topic modeling is more straightforward than in deep neural networks. Current autoencoders [30,33] are designed to exploit only the corpus, which we find insufficient by itself to exhibit NE-word relations, especially if these corpora are small or each NE mention is infrequent.

3.1 Notations and Preliminaries

Table 1 summarizes the notations used in the rest of the section. As outlined above, we obtain topics by factorising the short-text corpus representation. Formally, given the corpus $\mathcal{D}$, the vocabulary $\mathcal{V}$ consisting of unique words in $\mathcal{D}$, A the matrix corresponding to BoW representations of each document in $\mathcal{D}$,

and the target number of topics K, A can be approximated by the product of two low-rank matrices $A \approx HW$. Each row $W_{j,:}$ represents one of the K topics expressed in terms of words from $\mathcal{V}$ and each row $H_{i,:}$ represents an input document in terms of the learnt K topics.

Table 1. Notations used to present the topic modeling technique.

Name	Description
K, k	Number of topics, the identifier of a single topic
$\mathcal{D}, d$	Short-text documents found in the corpus, a single document
$\mathcal{V}, t, t',\ v_t$	Vocabulary set, individual terms, the embeddings of term t
$\mathcal{E},\ e,\ v_e$	Set of linked NEs, a NE term, the embedding of a NE e
$A \in \mathbb{N}^{\lvert\mathcal{D}\rvert \times \lvert\mathcal{V}\rvert}$	Term-document matrix with BoW corpus representation
$C \in \mathbb{R}^{\lvert\mathcal{V}\rvert \times \lvert\mathcal{V}\rvert}$	Word similarity matrix computed with pre-trained embeddings
$W \in \mathbb{R}^{K \times \lvert\mathcal{V}\rvert}$	Latent low-rank word-topic matrix
$H \in \mathbb{R}^{\lvert\mathcal{D}\rvert \times K}$	Latent low-rank document-topic matrix
$A^* \in \mathbb{R}^{\lvert\mathcal{D}\rvert \times \lvert\mathcal{V}\rvert}$	Word-document matrix for CluWords corpus representation
$A^{NE} \in \mathbb{R}^{\lvert\mathcal{D}\rvert \times \lvert\mathcal{V}\rvert}$	Word-document matrix for NE-informed corpus representation
$\alpha^{word}, \alpha^{ent} \in [0, 1)$	Minimum cosine similarity between words, or words and NEs

While this is the basic frameworks for NMF-based topic modeling, in practice there are more effective corpus representations than the simple BoW matrix (A), proven to lead to better topics. CluWords [31] is such an example and is based on two components: 1) one that correlates each word, not only with those with which co-occurs in the corpus, but also with other semantically related words, identified with the help of external pre-trained embeddings; 2) another one that derives a novel document representation, inspired by TF-IDF, which is able to incorporate information from the first component regarding word-to-word relatedness. In our work, we choose to extend CluWords document representations to explicitly prioritize NE cues. We further present the original CluWords, followed by the introduced changes in the next Subsect. 3.2.

The first step of CluWords is to compute a matrix C where each element $C_{t,t'}$ is the cosine similarity (cos) of the embeddings corresponding to the pair of terms $t, t' \in \mathcal{V}$. C is constrained to be non-negative as it is used to compute A^*, which is the input to NMF. Thus, a positive cutoff α^{word} is used to select only the most similar term pairs, and nullify the rest of the matrix:

$$C_{t,t'} = \begin{cases} \cos(v_t, v'_t) & \text{if } \cos(v_t, v'_t) > \alpha^{word} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Then, the BoW representation is replaced by a TF-IDF-inspired one. Standard TF-IDF uses the corpus statistics to decrease the weight of very frequent terms and give more weight to terms that appear only in some contexts, thus judged more discriminative, while also accounting for term popularity in a document. Equation 2 shows how the TF-IDF score is computed for a term t and a

document d, where $\text{tf}(t,d) = A_{d,t}$ is the number of times t appears in d and n_t is the number of documents in $\mathcal{D}$ where t appears:

$$\text{tf\_idf}(t,d) = \text{tf}(t,d) \cdot \log\left(\frac{|\mathcal{D}|}{n_t}\right) \tag{2}$$

CluWords replaces t by $C_{t,:}$ in order to avoid obtaining a very sparse representation matrix due to the limited context of each word in short text. Thus, it redefines the tf and idf (the log ratio) from Eq. 2 to be computed over vector-based term representations instead of individual frequencies. The new tf^* and idf^* in Eq. 3 incorporate information about semantically similar words to the term t of a given document d in order to expand the term's context:

$$A^*_{d,t} = \text{tf}^*(d,t) \cdot \text{idf}^*(t) = (AC)_{d,t} \cdot \log\left(\frac{|\mathcal{D}|}{\sum_{d \in \mathcal{D}} \mu(t,d)}\right) \tag{3}$$

$\mu(t,d)$ is the mean cosine similarity between the term t and its semantically related terms t' in document d denoted $\mathcal{V}^{d,t} = \{t' \in d | C_{t,t'} \neq 0\}$, or 0 when the ratio in the first branch of Eq. 4 is undefined (t is not in d, thus $|\mathcal{V}^{d,t}| = 0$):

$$\mu(t,d) = \begin{cases} \frac{1}{|\mathcal{V}^{d,t}|} \cdot \sum_{t' \in \mathcal{V}^{d,t}} C_{t,t'} & \text{if } |\mathcal{V}^{d,t}| > 0 \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

Let us note that in the limit case where C is the identity matrix, i.e. each term is only similar to itself which can be obtained by taking $\alpha^{word} = \max_{t \neq t'} C_{t,t'}$, Eq. (3) becomes equivalent to Eq. (2).

3.2 NE-informed Corpus Embedding (NEiCE)

Our approach NEiCE consists of a preprocessing step followed by a computation step which creates a new corpus representation matrix A^{NE} leveraging NEs.

Preprocessing Step. We identify NE mentions in podcast titles and descriptions and link them to Wikipedia entities using the Radboud Entity Linker (REL) system [10]. The REL system is based on multiple modules in pipeline specific to different sub-tasks: 1) the detection of NE mentions using Flair [1], a SOTA Named Entity Recognition (NER) framework using contextualized word embeddings; 2) the disambiguation of the identified entity against a list of possible Wikipedia candidates and its linking to the final candidate. In this final linking phase, REL [10] uses Wikipedia2Vec embeddings [34].

The Wikipedia2Vec embeddings that we also leverage in our solution, compared to other embeddings targeting words only [18,19], are learnt jointly for words and NEs from Wikipedia text. Their learning entails the optimization of three skip-gram sub-models [34]: 1) a regular word skip-gram; 2) an anchor context model—for each NE mention appearing as a hyperlink in text its surrounding words become context; and 3) a link graph model—the entities connected to a NE in the Wikipedia graph become context. From all the information REL

returns given a specific input, we use: the Wikipedia page of the disambiguated NE and the confidence score that helps us to choose if we treat a span of text as a NE or favour instead to process its words separately.

Finally, when NEs are processed as separate words instead of being linked to Wikipedia entities, we apply an extra vocabulary cleaning step. As we noticed that in podcast metadata mentions of actors, athletes, or celebrities were very common and we want to avoid the extraction of topics focused on names, we remove these concerned words using the package NameDataset[2].

Computation Step. We derive a new corpus representation matrix A^{NE} as explained next. If NEs are identified in a document with high confidence, then we exploit this information as the main topic-related cues. One strategy to achieve this from previous work on regular text [13] is to favour NEs among the top words to describe topics. Specifically, during preprocessing NEs are treated as n-gram terms and included in the vocabulary. Then, re-weighting approaches are applied to these terms before being served as input to a standard or variations of LDA. The idea behind re-weighting is to associate a larger pseudo-frequency (tf) to NEs such that they are more likely to be picked as topic descriptors.

Contrary to the above-mentioned approach, our goal is to take into account NEs without including them in the vocabulary. While indeed humans will find NEs very expressive to convey topics, this only happens if they already know them. For popular NEs which typically appear in news data exploited in [13], this would not necessarily pose a problem. However, the NEs from podcast metadata tend to be less common or very specific to certain domains, hence less informative for humans trying to associate a topic label. For instance, "That Peter Crouch Podcast" requires knowing that Peter Crouch is a footballer before being able to relate this podcast to football or sport.

The approach we propose is to still use re-weighting to boost NEs importance, but, instead of directly targeting NEs, focus on their semantically-related words. Let $\mathcal{E}^e = \{t | \cos(v_e, v_t) \geq \alpha^{ent}, \forall t \in \mathcal{V} - \mathcal{E}\}$ be the set of non-NE words from $\mathcal{V}$ most similar to a NE e. Similar to when we computed C, a threshold α^{ent} is applied to fix a minimum cosine similarity value between a pair of Wikipedia2Vec embeddings involving a NE ($e \in \mathcal{E}$) and a word ($t \in \mathcal{V}$). Then, we still compute A^{NE} with Eq. 3, but replace tf^* with tf^{NE} as follows:

$$\mathrm{tf}^{NE}_{d,t} = \begin{cases} (AC)_{d,t} + \max_{t' \in \mathcal{V}^{d,t}} (AC)_{d,t'} & \text{, if } t \in \mathcal{E}^e, \ e \text{ in } d \text{ and } |\mathcal{V}^{d,t}| > 0 \\ (AC)_{d,t} & \text{otherwise} \end{cases} \tag{5}$$

We chose to apply the NE-related re-weighting to the tf factor because we wanted to use NE-related words as the main signal for topics and the direct frequencies allowed us to have more control on it, as also emphasized by [13]. Second, there are two branches depending on whether t is a term very similar to a NE e present in d. If that is the case, a pseudo-frequency is computed by taking into account

[2] https://github.com/philipperemy/name-dataset.

the maximum in the CluWords tf matrix (tf^*) for a document d. This means that the words related to a NE e become either as important as the term with the largest weight (t') or more important if the word t already appeared in d.

4 Datasets

We start with describing the existing podcast datasets from iTunes [23] and Spotify [8]. Then, we introduce our newly collected dataset, Deezer, which is the largest one among the three as shown in Table 2. All these datasets contain podcast metadata, titles and descriptions, in English-language. Metadata is documented by providers or creators in an RSS feed, used by podcast aggregators and streaming platforms to make podcasts available to listeners. Although metadata exists for both podcasts (shows) and episodes within shows, we currently focus on shows as their information seemed more reliable. By manually analysing episode metadata in the podcast catalog to which we had access, we noticed they often lacked description or inherited show description.

The iTunes dataset [23] consists of 10 155 podcasts, popular at the moment of creation. The Spotify dataset [8] has 105 360 episodes sampled uniformly at random from podcasts proposed by professional creators (about 10%) and by amateur creators (about 90%). The metadata of each episode contains the title and description of the parent show which we extract to create the final dataset used in the experiments. From these two datasets, we keep podcasts with unique titles and with the concatenations of title and description longer than 3 terms. Additionally, for Spotify we select only the podcasts associated with the language identifiers "en" and "en-US".

Table 2. Summary of podcast datasets: the number of podcasts, the vocabulary size, the total number of NE mentions, the total number of podcasts with NEs in metadata, the mean number of words per title, and the mean number of words per description.

Dataset	$\lvert\mathcal{D}\rvert$	$\lvert\mathcal{V}\rvert$	#NE mentions	#podc. with NE	#w/title	#w/descr.
Spotify	17 456	7 336	20 885	9 198	3.5	38.2
iTunes	9 859	7 331	24 973	6 994	4.9	56.4
Deezer	29 539	14 322	67 083	19 969	4.0	62.6

Deezer differs from the others in that it is the largest. It covers 18 genres (Culture & Society, Business, Films & Games, Music & Audio Commentary, Comedian, Sports, Education, Spirituality & Religion, Information & Politics, Health & Fitness, Art, Entertainment, Lifestyle & Entertainment, Stories & Fiction, Science, Child & Family, True Crime, and History), with a minimum of 300 podcasts per genre. Although these categories are related to topics, as we previously discussed in Sect. 1, they tend to be broad and not always reliable. We could notice a significant overlapping (e.g. Entertainment with Lifestyle & Entertainment, Stories & Fiction with True Crime, or Sports with Health & Fitness), but also how a single category gathers multiple topics.

To create the dataset we randomly sampled from the accessed collection, public podcasts which had titles and descriptions, and the language identifier "en". As the language provided in the metadata was not always reliable, we also used two automatic language detectors, fastText [12] and CLD3 [27]. We filtered out podcasts which were not found to be in English by both detectors. Additionally, we also removed podcasts from unpopular genres (<300 shows). Finally, we applied the same preprocessing as for the other two datasets.

Table 2 presents additional statistics of the used datasets. All datasets contain a large number of NEs and we can find NE mentions in 50%–70% of the podcasts per dataset. We can also observe that the average number of words per title is quite similar for all datasets, while the descriptions in Spotify tend to be shorter.

5 Experimental Setup

We describe next the evaluation metric, the detailed preprocessing and experimental setup, and the environment we used for running the models.

We evaluated topic quality by relying on the widely used topic coherence [26]. A set of facts are said to have high coherence if they could support each other. In topic modeling, this translates into mapping terms on facts and measuring the extent to which these terms tend to co-occur in corpora. While the spectrum of word co-occurrence metrics for topic coherence is quite large [21], the exhaustive search performed in [26] shows that C_V correlates best with human judgement of topic ranking. Thus, we decided to report C_V scores in our evaluation. Given a topic k defined by its T top words $t_1, t_2, ..., t_T$, C_V is defined as:

$$C_V(k) = \frac{1}{T}\sum_{i=1}^{T} \cos(v_{NPMI}(t_i), v_{NPMI}(t_{1:T})) \tag{6}$$

$v_{NPMI}(t_i)$ and $v_{NPMI}(t_{1:T})$ yield two vectors computed with the Normalized Pointwise Mutual Information (NPMI) metric as follows:

$$v_{NPMI}(t_i) = \big(\mathrm{NPMI}(t_i, t_j)\big)_{j=1,..,T} \tag{7}$$

$$v_{NPMI}(t_{1:T}) = \left(\sum_{i=1}^{T} \mathrm{NPMI}(t_i, t_j)\right)_{j=1,..,T} \tag{8}$$

$$\mathrm{NPMI}(t_i, t_j) = \frac{\log \frac{p(t_i,t_j)}{p(t_i)p(t_j)}}{-\log(p(t_i, t_j))} \tag{9}$$

where p is the probability of a term occurrence or co-occurrence in an external corpus. We use Palmetto [26] to compute C_V for each topic k on Wikipedia as external corpus, and average over all K topics to obtain an aggregated value.

In all the reported experiments, we fix the number of top words T to 10 and vary the number of topics K between 20, 50, 100 and 200. During preprocessing, we keep all the linked NEs whose REL confidence score is higher than 0.9 even

if they only appear once in the corpus. For normal words, same as in [33], we filter out from vocabulary those that appear less than 5 times. We also remove stop words using NLTK [5]. The same preprocessing is applied before each topic modeling baseline. We evaluate GPU-DMM [15], NQTM [33], SeaNMF [29] and CluWords [31] with their default hyper-parameters. We assess the original CluWords with both fastText and Wikipedia2Vec embeddings [18].

As discussed in Sect. 3, NEiCE requires two parameters α^{word} and α^{ent}. [31] motivates the choice of α^{word} between 0.35 and 0.4 in CluWords as it allows to select top 2% of most similar pairs of words. Compared to this approach which assumes α^{word} mainly dependent on the pre-trained embeddings, we investigate if it varies per dataset. Thus, we test α^{word} with multiple values (0.2, 0.3, 0.4, 0.5), where larger the value is, fewer words are selected as being semantically-related to a given term. We proceed similarly for α^{ent}. We run the experiments on an Intel Xeon Gold 6134 CPU @ 3.20 GHz with 32 cores and 128 GB RAM.

6 Results and Discussion

The topic coherence scores obtained by the different topic modeling techniques for short text are presented in Table 3. First, we could notice that NMF-based methods (SeaNMF and CluWords) obtain the best scores in most of the cases. Second, when comparing individual techniques, the ranking depends on the case (number of topics and dataset), but few trends emerge. SeaNMF yields best topic coherence for the lowest number of topics (20) on two datasets. Aligned with the previous literature [31,33], the SOTA models, NQTM and CluWords, obtain very often the best or second best scores, with CluWords ranking first in most cases (7/12). These observations support our choices to work in a NMF framework and devise NEiCE as a CluWord extension, but informed by NEs.

Table 3. Topic coherence scores (C_V in %) obtained by baselines on the three podcast datasets for 20, 50, 100 or 200 topics. CluWords is used with fastText embeddings and the default $\alpha^{word} = 0.4$. Best scores are in bold and second best scores are underlined.

Model	Dataset											
	Deezer				**Spotify**				**iTunes**			
	20	50	100	200	20	50	100	200	20	50	100	200
GPU-DMM	39.0	38.3	37.6	40.1	39.5	39.4	39.7	40.1	39.6	38.5	42.0	41.1
NQTM	38.5	42.2	42.9	45.8	42.9	41.6	39.3	**40.2**	**48.4**	**46.6**	38.2	42.8
SeaNMF	**47.7**	40.5	37.3	39.0	**45.5**	36.4	36.6	35.7	42.2	41.8	35.1	36.9
CluWords$_{ft}$	39.7	**44.0**	**46.3**	**54.5**	40.2	**42.3**	**43.4**	39.5	42.7	40.1	**48.6**	**47.9**

Table 4 shows the results for CluWords with Wikipedia2Vec words embeddings for different values of α^{word}. As mentioned in Sect. 5, previously [31] this parameter was fixed depending on the source of embeddings to 0.4 for fastText and 0.35 for word2vec. However, no parameter sensitivity analysis was conducted, which we do now per dataset. We can see that the choice of α^{word}:

1) has a significant impact on the results which could vary up to almost 12 percentage points for Spotify, $K = 50$; 2) is dependent on the assessed case (dataset, K) which previously was not considered; and 3) some values appear to emerge as better choices per dataset (e.g. 0.4 for iTunes or 0.5 for Deezer).

Table 4. Topic coherence scores (C_V in %) obtained by CluWords for different α^{word} values (0.2, 0.3, 0.4, 0.5) with Wikipedia2Vec embeddings on the three podcast datasets for $K \in \{20, 50, 100, 200\}$ topics. Best scores are in bold.

Dataset	**Deezer**				**Spotify**				**iTunes**			
	20	50	100	200	20	50	100	200	20	50	100	200
CluWords$_{wk}$(0.2)	41.3	42.8	42.0	**45.9**	43.2	**49.0**	41.9	**43.0**	46.6	46.8	36.6	40.9
CluWords$_{wk}$(0.3)	39.8	41.3	45.6	44.1	42.8	37.8	46.4	37.8	44.6	40.7	39.0	40.3
CluWords$_{wk}$(0.4)	40.2	48.7	42.5	44.4	**48.4**	39.3	41.8	39.9	**52.9**	**48.5**	**49.6**	40.0
CluWords$_{wk}$(0.5)	**43.0**	**49.1**	**47.7**	41.6	47.3	37.2	**49.9**	42.7	45.3	40.4	41.1	**44.9**

Further, we present in Table 5 the topic coherence scores obtained with our proposed document representation, NEiCE, and different values of α^{word} and α^{ent}. First, we could notice that the introduction of NE cues has a positive impact and NEiCE obtains larger coherence scores than the baselines in most cases (datasets and numbers of topics). The average of NEiCE increase over the best baseline scores is of 15.7% for our best choice of parameters α^{word} and α^{ent}, with a maximum increase of 37.7% on Deezer and $K = 50$. Additionally, the underlined scores in Table 5, which represent scores larger than those obtained by the baselines, show that, no matter the choice of α^{word} and α^{ent}, NEiCE still yields better topic coherence in a majority of cases (85.4%). The most challenging case remains Deezer and $K = 200$ in which only $\alpha^{word} = 0.5$ and $\alpha^{ent} = 0.3$ lead to a larger score than the best baseline, although the increase is small so most likely not significant statistically.

Table 5. Topic coherence scores (C_V, in %) obtained by NEiCE, our document embedding strategy, for different values of (α^{word}, α^{ent}) using Wikipedia2Vec embeddings on the three podcast datasets. Best scores per dataset and number of topic are in bold. Scores larger than all baselines presented in Table 3 are underlined.

Dataset	**Deezer**				**Spotify**				**iTunes**			
	20	50	100	200	20	50	100	200	20	50	100	200
NEiCE (0.2, 0.3)	50.2	48.9	51.4	48.4	51.7	49.0	45.2	46.5	49.3	43.3	49.5	47.0
NEiCE (0.2, 0.4)	53.1	49.2	50.8	50.6	48.7	48.7	43.5	41.7	47.2	49.5	**50.7**	**51.3**
NEiCE (0.3, 0.3)	48.5	52.1	51.5	49.8	52.2	49.0	47.5	47.6	50.3	**52.5**	49.0	48.2
NEiCE (0.3, 0.4)	53.3	50.9	**55.3**	51.6	50.1	48.5	51.1	**49.8**	52.5	49.5	49.2	49.8
NEiCE (0.4, 0.3)	53.2	51.5	52.2	50.0	53.2	49.5	**50.5**	45.9	**52.8**	50.1	50.6	51.1
NEiCE (0.4, 0.4)	**56.4**	52.6	48.1	49.0	51.0	48.2	47.3	47.8	52.4	51.9	49.9	47.4
NEiCE (0.5, 0.3)	52.5	56.3	50.8	**55.4**	51.3	47.7	45.6	45.4	50.6	46.5	46.7	49.0
NEiCE (0.5, 0.4)	56.3	**60.6**	54.9	53.3	**55.0**	**49.9**	46.7	45.0	50.5	52.0	48.7	46.1

From Tables 4 and 5, we can notice that the best α^{word} in CluWords is not necessarily the best in NEiCE. For instance, on iTunes, $\alpha^{word} = 0.4$ was the best choice in Table 4, while in Table 5 $\alpha^{word} = 0.2$ appears a better choice. Also, the best pair of values for these parameters seems to depend largely on the case (dataset and K). Thus, a grid search on a hold-out set is advisable with NEiCE.

Table 6. Topics obtained with NEiCE or NQTM on Deezer and $K = 50$.

k	**NEiCE**	**NQTM**
1	mindfulness, yoga, meditation, psychotherapy, psychotherapist, hypnotherapy, psychoanalysis, hypnosis, therapist, psychology	psychotherapist, beirut, displays, remixes, weddings, adversity, namaste, kimberly agenda introducing
2	fiction, nonfiction, novel, author, book, novelist, horror, cyberpunk, anthology, fantasy	avenues, werewolf, criminal, pure, imaginative, strategies, demand, agree, oldies, hang
3	republican, senator, senate, libertarian, election, candidate, nonpartisan, conservative, caucus, liberal	hour, sudden, key, genres, keeps, round, neighbor, conservatives, realize, fulfillment

We selected some examples of topics obtained with NEiCE and NQTM[3] for Deezer and $K = 50$ in Table 6. We selected these topics considering the 18 genres introduced in Sect. 4 and assumed them likely related to Health & Fitness (1), Stories & Fiction or True Crime (2), and Information & Politics (3). Although NQTM yields more diverse top words, their association with a topic is less straightforward compared to NEiCE. However, topic 2 in NQTM is clearly about True Crime, while in NEiCE could be also about Stories & Fiction.

Finally, a qualitative analysis of the topics obtained with NEiCE on Deezer also revealed that many topics were related to world regions which, although easy to interpret, may be noisy if too frequent. These results may be related to the podcasts' topics, but a more likely explanation is that region-related NEs are overweighted. Thus, a detailed study of NE weighting in NEiCE is still needed.

7 Conclusion

We presented a detailed study of topic modeling on podcast metadata covering popular SOTA techniques for short text. Moreover, we proposed NEiCE, a new NE-informed document representation exploited in a NMF framework, and we showed it was more effective in terms of topic coherence than the baselines in various evaluation scenarios including three datasets (one of which, the largest, being newly released). Future work aims to extend the study at the episode level, assess the document representation in downstream tasks, gain more insights into NEiCE especially in relation to the pre-trained embeddings and the choices of αs, and conduct expert studies with editors to further validate mined topics.

[3] CluWords has similar top words as NEiCE for topics 1&2 and did not find topic 3.

References

1. Akbik, A., Blythe, D., Vollgraf, R.: Contextual string embeddings for sequence labeling. In: COLING 2018, 27th International Conference on Computational Linguistics, pp. 1638–1649 (2018)
2. Benton, G., Fazelnia, G., Wang, A., Carterette, B.: Trajectory based podcast recommendation. arXiv preprint arXiv:2009.03859 (2020)
3. Besser, J., Larson, M., Hofmann, K.: Podcast search: user goals and retrieval technologies. Online Inf. Rev. **43**(3), 395–419 (2010). https://doi.org/10.1108/14684521011054053
4. Bianchi, F., Terragni, S., Hovy, D.: Pre-training is a hot topic: contextualized document embeddings improve topic coherence. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pp. 759–766. Association for Computational Linguistics, August 2021. https://doi.org/10.18653/v1/2021.acl-short.96. https://aclanthology.org/2021.acl-short.96
5. Bird, S., Klein, E., Loper, E.: Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit. O'Reilly Media, Inc., Sebastopol (2009)
6. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3**(null), 993–1022 (2003)
7. Chen, Y., Zhang, H., Liu, R., Ye, Z., Lin, J.: Experimental explorations on short text topic mining between LDA and NMF based schemes. Knowl. Based Syst. **163**, 1–13 (2019)
8. Clifton, A., et al.: 100,000 podcasts: a spoken English document corpus. In: Proceedings of the 28th International Conference on Computational Linguistics, Barcelona, Spain, December 2020 pp. 5903–5917. International Committee on Computational Linguistics (2020). https://doi.org/10.18653/v1/2020.coling-main.519. https://aclanthology.org/2020.coling-main.519
9. He, R., Zhang, X., Jin, D., Wang, L., Dang, J., Li, X.: Interaction-aware topic model for microblog conversations through network embedding and user attention. In: Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe, New Mexico, USA, August 2018, pp. 1398–1409. Association for Computational Linguistics (2018). https://aclanthology.org/C18-1118
10. van Hulst, J.M., Hasibi, F., Dercksen, K., Balog, K., de Vries, A.P.: REL: an entity linker standing on the shoulders of giants. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR 2020, New York, NY, USA, pp. 2197–2200. Association for Computing Machinery (2020). https://doi.org/10.1145/3397271.3401416
11. Jones, R., et al.:: Current challenges and future directions in podcast information access. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR 2021, New York, NY, USA, pp. 1554–1565. Association for Computing Machinery (2021). https://doi.org/10.1145/3404835.3462805
12. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, Valencia, Spain, April 2017, pp. 427–431. Association for Computational Linguistics (2017). https://aclanthology.org/E17-2068

13. Krasnashchok, K., Jouili, S.: Improving topic quality by promoting named entities in topic modeling. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Melbourne, Australia, July 2018. pp. 247–253. Association for Computational Linguistics (2018). https://doi.org/10.18653/v1/P18-2040. https://aclanthology.org/P18-2040
14. Kuang, D., Choo, J., Park, H.: Nonnegative matrix factorization for interactive topic modeling and document clustering. In: Celebi, M.E. (ed.) Partitional Clustering Algorithms, pp. 215–243. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-09259-1_7
15. Li, C., Wang, H., Zhang, Z., Sun, A., Ma, Z.: Topic modeling for short texts with auxiliary word embeddings. In: Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR 2016, New York, NY, USA, pp. 165–174. Association for Computing Machinery (2016). https://doi.org/10.1145/2911451.2911499
16. Mehrotra, R., Sanner, S., Buntine, W., Xie, L.: Improving LDA topic models for microblogs via tweet pooling and automatic labeling. In: Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR 2013, New York, NY, USA, pp. 889–892. Association for Computing Machinery (2013). https://doi.org/10.1145/2484028.2484166
17. Miao, Y., Yu, L., Blunsom, P.: Neural variational inference for text processing. In: Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48. ICML 2016, pp. 1727–1736. JMLR.org (2016)
18. Mikolov, T., Grave, E., Bojanowski, P., Puhrsch, C., Joulin, A.: Advances in pre-training distributed word representations. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan, May 2018. European Language Resources Association (ELRA) (2018). https://aclanthology.org/L18-1008
19. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2. NIPS 2013, Red Hook, NY, USA, pp. 3111–3119. Curran Associates Inc. (2013)
20. Mizuno, J., Ogata, J., Goto, M.: A similar content retrieval method for podcast episodes. In: 2008 IEEE Spoken Language Technology Workshop, pp. 297–300 (2008). https://doi.org/10.1109/SLT.2008.4777899
21. Newman, D., Lau, J.H., Grieser, K., Baldwin, T.: Automatic evaluation of topic coherence. In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. HLT 2010, USA, pp. 100–108. Association for Computational Linguistics (2010)
22. Nigam, K., McCallum, A.K., Thrun, S., Mitchell, T.: Text classification from labeled and unlabeled documents using EM. Mach. Learn. **39**(2), 103–134 (2000)
23. Ozturk, D.G.: Podcasts Data. https://github.com/odenizgiz/Podcasts-Data. Accessed 20 Sept 2021
24. Quan, X., Kit, C., Ge, Y., Pan, S.J.: Short and sparse text topic modeling via self-aggregation. In: Proceedings of the 24th International Conference on Artificial Intelligence. IJCAI 2015, pp. 2270–2276. AAAI Press (2015)
25. Research, E.: The Podcast Consumer 2019 (2019). https://www.edisonresearch.com/the-podcast-consumer-2019/. Accessed 20 Sept 2021
26. 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. WSDM 2015, New York, NY, USA, pp. 399–408. Association for Computing Machinery (2015). https://doi.org/10.1145/2684822.2685324

27. Salcianu, A., et al.: Compact Language Detector v3 (CLD3). https://github.com/google/cld3. Accessed 20 Sept 2021
28. Sharpe, M.: A review of metadata fields associated with podcast RSS feeds. arXiv preprint arXiv:2009.12298 (2020)
29. Shi, T., Kang, K., Choo, J., Reddy, C.K.: Short-text topic modeling via non-negative matrix factorization enriched with local word-context correlations. In: Proceedings of the 2018 World Wide Web Conference. WWW 2018, pp. 1105–1114. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE (2018). https://doi.org/10.1145/3178876.3186009
30. Srivastava, A., Sutton, C.: Autoencoding variational inference for topic models. In: ICLR (2017)
31. Viegas, F., et al.: CluWords: exploiting semantic word clustering representation for enhanced topic modeling. In: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. WSDM 2019, New York, NY, USA, pp. 753–761. Association for Computing Machinery (2019). https://doi.org/10.1145/3289600.3291032
32. Winn, R.: 2021 Podcast Stats & Facts (New Research From April 2021). https://www.podcastinsights.com/podcast-statistics/. Accessed 20 Sept 2021
33. Wu, X., Li, C., Zhu, Y., Miao, Y.: Short text topic modeling with topic distribution quantization and negative sampling decoder. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1772–1782. Association for Computational Linguistics, November 2020. https://doi.org/10.18653/v1/2020.emnlp-main.138. https://aclanthology.org/2020.emnlp-main.138
34. Yamada, I., et al.: Wikipedia2Vec: an efficient toolkit for learning and visualizing the embeddings of words and entities from Wikipedia. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pp. 23–30. Association for Computational Linguistics (2020)
35. Yang, L., Wang, Y., Dunne, D., Sobolev, M., Naaman, M., Estrin, D.: More than just words: modeling non-textual characteristics of podcasts. In: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. WSDM 2019, New York, NY, USA, pp. 276–284. Association for Computing Machinery (2019). https://doi.org/10.1145/3289600.3290993
36. Yin, J., Wang, J.: A dirichlet multinomial mixture model-based approach for short text clustering. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD 2014, New York, NY, USA, pp. 233–242. Association for Computing Machinery (2014). https://doi.org/10.1145/2623330.2623715
37. Zhao, W.X., et al.: Comparing Twitter and traditional media using topic models. In: Clough, P., et al. (eds.) ECIR 2011. LNCS, vol. 6611, pp. 338–349. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-20161-5_34
38. Zuo, Y., et al.: Topic modeling of short texts: a pseudo-document view. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD 2016, New York, NY, USA, pp. 2105–2114. Association for Computing Machinery (2016). https://doi.org/10.1145/2939672.2939880