More than Just Words: Modeling Non-textual Characteristics of Podcasts

Longqi Yang
Cornell Tech, Cornell University
ylongqi@cs.cornell.edu

Yu Wang
Himalaya Inc.
yu.wang@himalaya.com

Drew Dunne
Cornell University
asd222@cornell.edu

Michael Sobolev
Cornell Tech, Cornell University
michael.sobolev@cornell.edu

Mor Naaman
Cornell Tech, Cornell University
mor.naaman@cornell.edu

Deborah Estrin
Cornell Tech, Cornell University
destrin@cornell.edu

ABSTRACT

Recent years have witnessed the flourishing of podcasts, a unique type of audio medium. Prior work on podcast content modeling focused on analyzing Automatic Speech Recognition outputs, which ignored vocal, musical, and conversational properties (e.g., energy, humor, and creativity) that uniquely characterize this medium. In this paper, we present an Adversarial Learning-based Podcast Representation (ALPR) that captures non-textual aspects of podcasts. Through extensive experiments on a large-scale podcast dataset (88,728 episodes from 18,433 channels), we show that (1) ALPR significantly outperforms the state-of-the-art features developed for music and speech in predicting the *seriousness* and *energy* of podcasts, and (2) incorporating ALPR significantly improves the performance of topic-based podcast-popularity prediction. Our experiments also reveal factors that correlate with podcast popularity.

KEYWORDS

Podcast; Spoken word; Content modeling; Popularity prediction

ACM Reference Format:
Longqi Yang, Yu Wang, Drew Dunne, Michael Sobolev, Mor Naaman, and Deborah Estrin. 2019. More than Just Words: Modeling Non-textual Characteristics of Podcasts. In *The Twelfth ACM International Conference on Web Search and Data Mining (WSDM '19), February 11–15, 2019, Melbourne, VIC, Australia.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3289600.3290993

1 INTRODUCTION

Podcast is a portable and on-demand form of spoken-word audio content, which has emerged as a significant channel for information, entertainment, and advertising. According to a recent national survey [40], as of 2017, there are 67 million monthly and 42 million weekly podcast listeners in the United States, and the per-listener average listening time is over five hours per week. Compared to text and video content, audio is easier to consume when users have limited visual attention, which makes podcasts a perfect fit for commuting, exercising, cooking, and household chores. On the content supply side, tens of thousands of high-quality podcasts are produced on a daily basis. For example, most established news media companies publish content in the form of podcasts.[1]

Prior work on podcast content modeling focused on the task of spoken content retrieval [11, 27, 28, 36] aimed at indexing Automatic Speech Recognition (ASR) outputs for media search. Whereas transcriptions characterize important properties of podcasts (e.g., keywords, phrases, and topics), they do not capture conversational, paralinguistic, vocal, and musical aspects of this medium. These non-textual properties may inform search [28] and personalized recommendation [51] as well as content production.

In this paper, we model non-textual characteristics of podcasts and explore their benefits to podcast-popularity prediction. To benchmark the modeling performance, we collected a podcast dataset containing 88,728 episodes from 18,433 channels. In addition, we crowdsourced labels for a randomly sampled subset, where each audio snippet was labeled with a *seriousness* and *energy* score. These non-textual characteristics were chosen based on analysis of iTunes reviews and published literature.

We initially experimented with existing audio modeling algorithms for the task of predicting the *seriousness* and *energy* of podcasts. These algorithms include state-of-the-art hand-crafted music and speech features (MFCC [30], IS09 [49], IS13 [48]), and standard DNN-based representation learning frameworks (autoencoder and variational autoencoder [12]). However, our experimental results suggest that these methods manifest suboptimal prediction performance, because they are unable to capture complex variations in podcast audio. To address this limitation, we leverage adversarial learning [13] and investigate an unsupervised learning algorithm that progressively builds podcast representations from fine-grained spectrogram details. **A**dversarial **L**earning-based **P**odcast **R**epresentation (**ALPR**) captures subtleties of complex audio spectrograms and achieves significantly better performance in predicting non-textual attributes.

In addition, we conducted a podcast-popularity-prediction experiment with different features, including topics mined from transcriptions, existing audio features, and ALPR. We observe significant performance gain by incorporating ALPR into the topic-based predictor, whereas there is no improvement in cases where prior audio features are used. Our experiments also reveal factors that correlate with podcast popularity, including positively correlated factors (e.g.,

WSDM '19, February 11–15, 2019, Melbourne, VIC, Australia

ACM ISBN 978-1-4503-5940-5/19/02…$15.00
https://doi.org/10.1145/3289600.3290993

[1] NPR: http://www.npr.org/podcasts/; New York Times: https://www.nytimes.com/podcasts/; Washington Post: https://www.washingtonpost.com/podcasts/.

perceived energy and topics related to family, politics, crime, and food), and negatively correlated factors (e.g., extensive use of functional words). These findings may have implications for podcast recommendation and production.

The main contributions of this paper are three-fold: (1) a labeled podcast dataset that contains 88,728 episodes from 18,433 channels, (2) a representation learning algorithm that captures non-textual characteristics of podcasts and significantly outperforms existing approaches from speech and music communities, and (3) enhanced podcast-popularity prediction by incorporating improved podcast representation. The code and dataset are available at: https://github.com/ylongqi/podcast-data-modeling

2 PROBLEM FORMULATION AND LITERATURE REVIEW

We set out to construct features or representations that are predictive of non-textual podcast characteristics. End applications, such as recommendation engines [52], can then leverage these features as additional data inputs. This problem formulation is inspired by research in other content domains [47]. For example, computer vision has focused on designing informative image features for object identification and classification, where these features were hand-crafted [31] or were the outputs of the last layer of a Deep Neural Network (DNN) [18].

Based on this problem formulation, we evaluate the modeling performance of different features by using them for binary classification of non-textual attributes and measuring the classification accuracy. Such an approach has been validated and widely adopted in other fields [47]. We chose *seriousness* and *energy* attributes based on analysis of iTunes reviews and related literature:

iTunes reviews analysis. We collected 850K reviews that iTunes subscribers posted against 2.5K podcast channels and counted the word frequencies in all of the five-star reviews. The top adjectives that listeners mentioned were *funny*, *entertaining*, and *hilarious*, which reveals that the humorousness or *seriousness* is an important attribute of an appealing podcast.

Literature review. We also studied the music recommendation literature [16, 46] to discover non-textual attributes that may be important for podcasts. Previous work [16, 46] has suggested that (fast or slow) rhythm and (energetic or calm) sound are important attributes for context-aware music recommendation. Since consuming contexts are significantly overlapped for music and podcasts, the *energy* may become an important podcast characteristic.

Our work builds on prior research on speech modeling, spoken content retrieval, and music information retrieval.

2.1 Speech modeling

A large portion of podcasts are in the form of conversational, interview, or monologue speech, which have been widely studied by the speech community in the context of Automatic Speech Recognition (ASR) [15], dialog systems [29], and speech perception [21]. However, the datasets that have been studied so far are, by and large, limited to the pure speech form, which lacks diversity and variability compared to podcast audio. For example, TIMIT [9] contains clean speech recordings of English speakers reading sentences, TED-LIUM [42] transcribed TED talks for speech language modeling, LibriSpeech [38] collected clean speech from audiobooks, and the command dataset [41] includes speech data for short spoken command recognition. Recently, Google released AudioSet [10], a large scale dataset that contains sound clips collected from YouTube videos. Although it includes human speech as one of the categories, it is unclear how it can be used beyond classifying sound events.

Our work builds upon and greatly extends models developed in the speech community to analyze podcast audio. We compare our algorithm against the state-of-the-art representations for speech and demonstrate that our learned features are significantly more representative of the non-textual properties of podcasts.

2.2 Spoken content retrieval

The spoken content retrieval community [11, 27, 28, 36] has studied podcasts in the context of web search [3, 8, 14, 33, 37]. For example, Fuller et al. [8] explored the usage of term clouds for podcast visualization, Besser et al. [3] studied the user goals and strategies for podcast search, and Goto et al. [14, 33, 37] built the *Podcastle* system that used keywords to index podcast content. The existing content modeling algorithms for podcasts are limited to transcription analysis, which is insufficient in characterizing the diverse nature of podcasts.

In this work, we investigate the problem of modeling non-textual properties of podcasts, which was not studied previously. The developed model has applications not only to search, but also to other podcast applications, e.g., recommendation and content production.

2.3 Music information retrieval

Research from the Music Information Retrieval (MIR) community analyzed music audio to classify various aspects of musical content. For example, genre [2], chord [19], and rhythm [4]. Prior work has applied these analysis to many applications, such as music recommendation [5, 44–46, 50]. Although many podcast audio snippets contain background music, and it is an important aspect of this medium, musical analysis alone does not capture vocal, paralinguistic, conversational, and presentation aspects of podcasts.

In this work, we compare our learned representations to the classical feature sets used in the music community. We demonstrate that solely analyzing music yields suboptimal performance in characterizing podcasts and predicting their popularity.

3 PODCAST AUDIO REPRESENTATIONS

Representations for podcast audio can either be hand-crafted with expert knowledge or learned from data [12]. In this section, we review existing solutions from various domains, discuss their limitations, and propose an adversarial learning-based representation learning approach tailored for complex podcast audio.

3.1 Existing approaches

Hand-crafted features. In the speech and music communities [1, 48, 49], researchers have designed many feature sets to encode various audio properties. For example, MFCC [50], IS09 [49], and IS13 [48]. These representations achieve state-of-the-art performance (on par with supervised convolutional neural network based models) in many prediction tasks, such as recommendation [50]

and emotion recognition [1]. However, these feature sets fall short in characterizing the diverse nature of podcasts because fundamentally, podcast audio includes both musical and heterogeneous spoken components. Practically, due to the rapid growth of new channels, it is labor-intensive to exhaustively explore the content space of podcasts and manually design all the important features.

Standard feature learning approaches. Standard DNN-based representation learning algorithms, such as AutoEncoder (AE) [12] and Variational AutoEncoder (VAE) [12], tend to capture global patterns of input data but lose nuanced details. For example, in the image-to-image translation task, Isola et al. [22] demonstrated that the encoder–decoder architecture produces images that are mostly blurred. While such limitations are not critical for applications that rely on global patterns (e.g., natural image classification), they are vital for podcast audio modeling since the vocal and musical variations are usually manifested locally in spectrograms.

3.2 Adversarial learning-based approach

To tackle the limitations of existing methods, we apply adversarial learning [13] to learn podcast audio representations from data.

Motivations. Because of the heterogeneity of podcast audio, an ideal feature learning algorithm needs to be able to attend to subtle variations in the data, for which adversarial learning has shown great promise. For example, recent work demonstrated the power of Generative Adversarial Networks (GAN) in generating images with fine-grained textures [39]. Adversarial networks achieve this by co-evolving a generator and a discriminator. Throughout the process, a weak component is easily defeated by its opponent, which results in a final equilibrium where both components are relatively strong. In other words, a strong generator urges the discriminator to learn non-trivial feature representations that capture nuanced variations of input data.

General Framework. As shown in Fig. 1, our proposed framework operates on spectrograms, a commonly used raw representation of audio signals. It consists of two major components: a generator trained to generate spectrograms that are indistinguishable from real ones, and a discriminator trained to distinguish between real and generated spectrograms. After training, the discriminator network is used as the feature extractor, and the corresponding output is treated as the podcast audio representation.

Following the notation from [13], we use $G(\boldsymbol{z};\theta_g)$ to represent the generator function (parameterized by θ_g), which maps random vectors $\boldsymbol{z}$ drawn from a fixed distribution $p_{\boldsymbol{z}}(\boldsymbol{z})$ to generated spectrograms, and use $D(\boldsymbol{x};\theta_d)$ to represent the discriminator function (parameterized by θ_d) that takes (real or generated) spectrograms as inputs and ouputs feature representations. Our adversarial framework trains D and G networks to optimize a min-max criteria,

$$\min_{\theta_g}\max_{\theta_d,W,b} \mathbb{E}_{\boldsymbol{x}\sim p_{\text{podcast}}(\boldsymbol{x})}[\log(\sigma(W\cdot D(\boldsymbol{x};\theta_d)+b))]+ \mathbb{E}_{\boldsymbol{z}\sim p_{\text{uniform}}(\boldsymbol{z})}[\log(1-\sigma(W\cdot D(G(\boldsymbol{z};\theta_g);\theta_d)+b))],$$

where $\sigma(x)=\frac{1}{1+e^{-x}}$, and W, b, θ_g and θ_d are trainable parameters.

To optimize for this objective, we alternately train (W, b, θ_d) and θ_g, as Fig. 1 shows. The parameters θ_g are fixed while training (W, b, θ_d), and vice versa. Essentially, the generator is trained to *fool* the discriminator by generating examples that the discriminator perceives as *real*, and the discriminator is trained to attend to patterns that reliably distinguish generated spectrograms from real spectrograms. In reality, training such adversarial networks is notoriously hard and unstable [43]. To stabilize the training, we leverage the feature matching technique proposed in [43]. The idea is to replace the training objective for the generator with the objective of matching features' statistics, that is, minimizing the distances between the element-wise mean of $D(G(\boldsymbol{z}))$ and $D(\boldsymbol{x})$. Specifically, the generator is trained to solve the following minimization problem:

$$\min_{\theta_g} \|\mathbb{E}_{\boldsymbol{x}\sim p_{\text{podcast}}}[D(\boldsymbol{x};\theta_d)]-\mathbb{E}_{\boldsymbol{z}\sim p_{\text{uniform}}}[D(G(\boldsymbol{z};\theta_g);\theta_d)]\|_2^2$$

Next, we describe the detailed design of the two major components.

The generator G. The design of the generator G is summarized in Fig. 2. G takes as input a random vector $\boldsymbol{z}$ drawn from a uniform distribution and produces a spectrogram $\boldsymbol{x}$ of shape $(h, w, 1)$, where

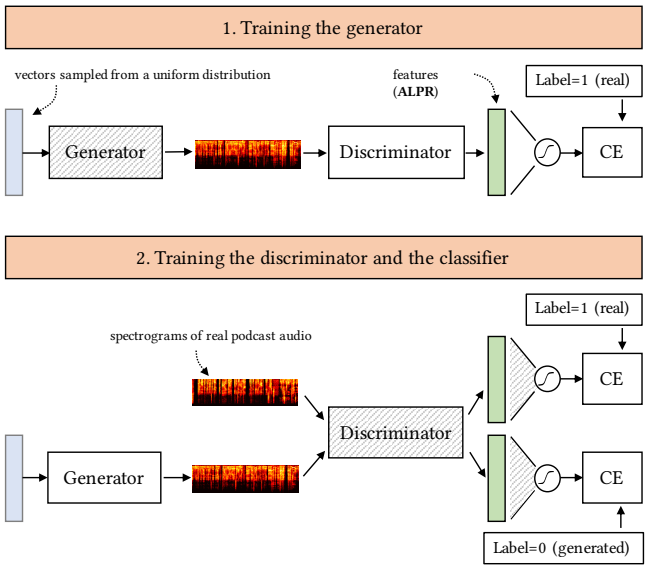


Figure 1: Our adversarial framework learns podcast audio representations. We alternately execute two steps to train the generator and the discriminator. The trainable components are shaded in both steps, and the green vectors are ALPR. CE: Cross Entropy.

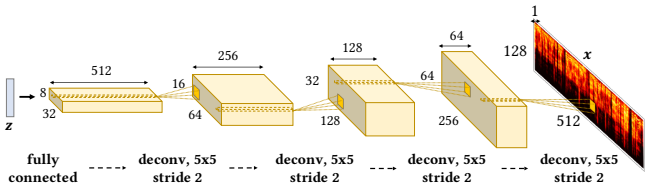


Figure 2: The architecture of the generator G. The input z is a random vector sampled from a uniform distribution, and the output x is the generated spectrogram with depth of 1. Each cube represents the output of a layer and is labeled with its height, width and depth (i.e., the number of channels or feature maps).

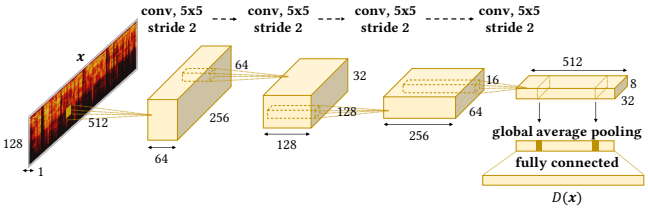


Figure 3: The architecture of the discriminator D. The input $\boldsymbol{x}$ is a real or generated spectrogram, and the output $D(\boldsymbol{x})$ is the ALPR feature representation (Legends follow Fig. 2).

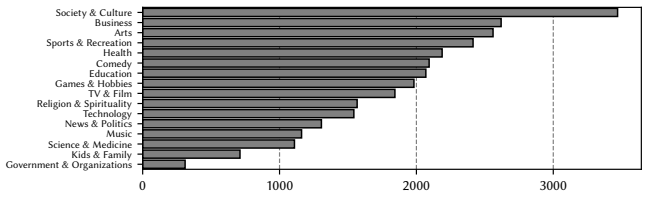


Figure 4: Number of channels collected in each of the 16 iTunes podcast categories.

corpus: 88,728 episodes (18,433 channels)	S_A (ALPR training & attributes pred.): 42,370 episodes (2,160 snippets labeled)
	S_B (popularity pred.): 46,358 episodes (6,511 episodes labeled)

Table 1: Summary of the podcast corpus. The corpus is randomly divided into two disjoint sets for different prediction tasks. The length of the labeled audio snippets in S_A is 12s.

h is the number of components used in the mel filter banks (i.e. the frequency granularity) and w is the number of sliding windows (time length). In this paper, we set $h = 128$ and $w = 512$.[2]

The generator consists of a fully connected layer and four de-convolutional layers (i.e., fractionally-strided convolutions) [35]. The fully connected layer projects the random vector $\boldsymbol{z}$ into a $8 \times 32 \times 1024$ tensor that contains 1024 feature maps (channels) with size $(8, 32)$ (i.e., the value of each cell is a linear combination of the elements in $\boldsymbol{z}$). Afterwards, successive de-convolutional layers with a stride of 2 and the filter size of $(5, 5)$ gradually upsample feature maps from size $(8, 32)$ to $(128, 512)$. In contrast to a convolutional layer, a de-convolutional layer connects a single input to multiple outputs in each filter window (See [35] for details). In our network, after each deconvolutional layer, the width and height of feature maps get doubled, and the number of channels get halved. At the end, the generator outputs the spectrogram $\boldsymbol{x}$ (a single channel with size $(128, 512)$). We apply Batch Normalization (BN) [20] and Rectified Linear Units (ReLU) [34] to the outputs of each layer except the last layer, for which we use element-wise tanh to normalize output values to the range of $[-1, 1]$. ReLU adds non-linearity to the network, and BN addresses the problem of vanishing and exploding gradients during training [20].

The discriminator D. The design of the discriminator D is illustrated in Fig. 3. Compared to the generator, D maps an input spectrogram $\boldsymbol{x}$ to a dense feature vector $D(\boldsymbol{x})$, which is consumed by a classifier to predict the truthfulness (Fig. 1). Specifically, the discriminator contains (1) an initial sequence of convolutional layers that downsample the input spectrogram from size $(128, 512)$ to $(8, 32)$, and the number of channels is doubled whenever the width or height is halved, (2) a global pooling layer that independently averages every feature map and outputs a 512-dimensional feature vector (the dimensionality is equal to the number of channels), and (3) a fully connected layer that produces the high-dimensional feature representation $D(\boldsymbol{x})$. We apply BN and Leaky ReLU to the outputs of all layers except the global average pooling layer, and BN is not applied to the first convolutional layer. LeakyReLU, instead of ReLU, is used to stabilize the model training [39]. We find that the *bottleneck structure* (global pooling+fully connected) used in the discriminator is critical for the feature learning, and the same evidence was also found in other applications using deep convolutional neural network (CNN) [18].

[2]With a window size of 2048, a step size of 1024, and a sampling rate of 44100Hz, a spectrogram spans approximately 12s of audio.

Model generalization. Our framework is generally applicable to podcast data — although the model has fixed parameters and is trained using spectrograms with a fixed shape, it can be used to extract features for spectrograms of any time length, because convolutional filters are agnostic to input shapes, and the global average pooling layer reduces feature maps of any size into a fixed size (512-dimensional) vector.

4 DATASET COLLECTION AND ANNOTATION

We collected a large-scale podcast dataset for model training and evaluation. Specifically, we scraped the iTunes podcast directory and kept only active channels that published at least five episodes from July 2016 to July 2017. For each channel, we downloaded the raw audio of the most recent five episodes and deleted ill-formatted files. Our podcast audio corpus (summarized in Table. 1) contains 88,728 episodes from 18,433 channels covering podcasts from a wide range of categories, as shown in Fig. 4. In this work, we used at most the leading 10 minutes of each episode. To test the generalizability of different features, we evenly split episodes into two disjoint sets (S_A and S_B) for the attributes-prediction and the popularity-prediction tasks, respectively (Table. 1).

4.1 Attributes annotation

We divided sound signals in S_A into snippets of 12s each (i.e., 524,288 data points under standard sampling rate of 44100 Hz) and randomly sampled 2,160 audio snippets from distinct episodes to collect labels. We used Amazon Mechanical Turk platform for annotations. Each worker was instructed to indicate how energetic and serious is an audio snippet by using sliders ranging from *calm* to *energetic* and from *humorous* to *serious* respectively (we ensured that workers played the entire sample) (Fig. 5). For each snippet, we collected labels from five distinct workers. To calibrate the scale of the attributes, we provided an audio sample for each adjective shown in Fig. 5, and workers were required to listen to the samples before starting the annotation task. To control label quality, we recruited only workers who were located in United States, had over 90%

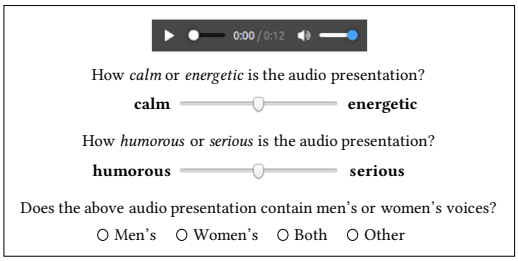


Figure 5: A sample user interface for the Amazon Mechanical Turk task. Each task consists of 12 repetitive blocks shown in this figure but with different audio sources. We place the initial position of the sliders at the middle, and the last question is single-choice.

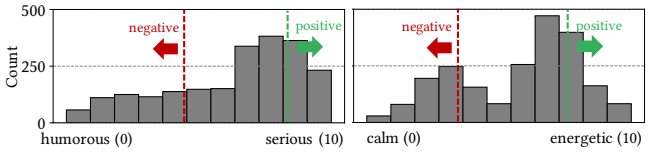


Figure 6: The distributions of the annotated *seriousness* and *energy* scores. Under each attribute, the score is discretized into 10 bins. Audio snippets that have scores higher than the green dotted lines are treated as positive samples, and those that have scores lower than the red dotted lines are treated as negative samples.

approval rate, and were identified as *masters* by the mechanical turk platform. In addition, we grouped audio chunks into batches of size 12 (i.e., each task contained 12 snippets)[3] and added a verification question for each audio snippet, that is, *does the above audio presentation contain men's or women's voices?* (Fig. 5). Workers who submitted wrong answers[4] to the verification question were excluded from the labeling task, and we recollected the corresponding labels so that every audio snippet had five valid annotations. At the end, 178 unique and valid workers participated in the annotation.

4.2 Evaluation dataset for attributes prediction

To build an evaluation dataset for snippet-level attributes, we constructed a balanced training set (i.e., it contains the same number of positive and negative samples) and a disjoint, balanced, and held-out testing set, for each attribute.

For each annotated audio snippet, we discretized workers' ratings from 0 (*calm* or *humorous*) to 10 (*energetic* or *serious*) based on the position of the slider and regarded the median of five annotations to be the ground truth *seriousness* or *energy* score. In Fig. 6, we show the distributions of both scores among 2,160 audio snippets: the *seriousness* score is uni-modally distributed and is skewed towards *serious* while the *energy* score is bi-modal. To use numerical ground truth scores for binary classifications, we treated the audio snippets that were scored at the top 25 percentile as positive samples and the snippets that were scored at the bottom 25 percentile as negative samples (Fig. 6), because the boundary between *somewhat energetic/serious* and *somewhat calm/humorous* may be blurry. Such a process is a widely adopted practice to alleviate potential ambiguity [6]. We also tried other score aggregation methods[5], and they produced significantly overlapped binary labels. To demonstrate the reliability of the labels, we computed the Krippendorff's Alpha coefficient [17, 25] with *interval distance* [25], that is, $d = (a - b)^2$ where a and b are two labels. The coefficient was chosen for its ability to handle numerical labels. The agreement scores for attributes *energy* and *seriousness* were 0.64 and 0.77 respectively for snippets in our training and testing sets. According to Landis et al. [26], annotations for both attributes have reached *substantial* consensus. Finally, for each attribute, 540 samples were identified as positive, and another 540 were identified as negative. We split them evenly and randomly into a training set and a testing set (i.e., each set contained 270 positive and 270 negative samples).

5 PREDICTING NON-TEXTUAL ATTRIBUTES

With the collected attributes-prediction dataset, we compared the performance of ALPR to several baseline features.

5.1 Experimental setups

Training ALPR. We used all chunked audio snippets in S_A to train ALPR. For each audio snippet (with 524,288 data points each), we calculated log-scaled mel-spectrograms with 128 components using a window size of 2048 and a step size of 1024, which produced spectrograms with shape (128, 512). The values in spectrograms were capped to the range of $[-4, 0]$ and then linearly re-scaled to the range of $[-1, 1]$. Our final training set contains 2,081,325 unique podcast audio spectrograms.

We used Adam [24] for optimization. During training, for each iteration, the generator was trained twice, while the discriminator was trained once. We found that this was critical in training the adversarial network in order to prevent the training loss going to zero while training the discriminator. Other hyperparameter settings of our model include: the random vector $\boldsymbol{z}$ was set to be 100 dimensions and was sampled from a uniform distribution over the range $[-1, 1]$; the dimensionality of the feature representation ($D(\boldsymbol{x})$), was set to be 4096; and the model was trained for 75,000 iterations with the batch size of 64.

Baselines. The baselines include three advanced hand crafted feature sets from music and speech communities, as well as AE and VAE that learn features from unlabeled data using CNN:

MFCC. Mel-frequency Cepstrum Coefficients (MFCC) is a classical feature set used in many music and speech applications, such as Music Information Retrieval (MIR) [30], recommendation [50], and speech recognition [7]. The baseline MFCC feature set was calculated as follows: (1) we computed 13 MFCCs from a window of size 2048. With a step size of 1024, 512 vectors were derived for each audio chunk in the dataset, then (2) we used the K-means algorithm to learn a dictionary of 4096 elements from a randomly selected subset of spectrograms, and each MFCC vector was assigned to the closest element (We set the size of the dictionary to

[3]Workers were compensated for $0.5 per batch (Estimated hourly wage: $7.5).
[4]We manually checked inconsistent answers from different workers.
[5]The mean of five annotations produced labels overlapped 95% (*seriousness*) and 98% (*energy*), and applying the z-score normalization for each worker before calculating the median produced labels overlapped 82% (*seriousness*) and 80%(*energy*).

be the dimensionality of ALPR). Finally, (3) for each audio chunk, we counted the number of times that each element or cluster was assigned, and the 4096-dimensional bag-of-words vector was the MFCC representation of a podcast audio snippet.

IS09. IS09 [49] is the feature set used in the INTERSPEECH 2009 Emotion Challenge. It is a 384-dimensional feature vector that covers a wide range of low-level descriptors for various audio patterns, e.g., frequency, pitch, harmonics, and frame energy. We refer readers to the original challenge for feature details [49]. Until recently [1], it still achieves performance on par with supervised CNN-based algorithms in speech emotion recognition.

IS13. IS13 [48] is the feature set used in the INTERSPEECH 2013 computational paralinguistics challenge, which includes detecting non-linguistic events such as laughter or sigh of a speaker, recognizing conflicts in group discussions, classifying emotions, and determining the type of pathology of a speaker. Because of the complexity and diversity of these tasks, 6373 hand-crafted features were introduced, which is arguably the best feature set for speech related tasks. Similar to IS09, IS13 also shows performance on par with supervised CNN-based algorithms [1]. We used openSMILE software [7] to extract IS09 and IS13 features for audio snippets.

Autoencoder (AE) and *Variational AutoEncoder (VAE)*. Autoencoder [12] consists of an encoder that transforms input data to fixed-sized representations and a decoder that reconstructs input data from the representations. The training is driven by minimizing a reconstruction error, and the trained encoder is used as the feature extractor. Different from AE, for VAE, the encoder outputs a distribution parameterized by two vectors representing mean and variance, respectively, and the decoder takes as input a sampled vector from the distribution. During training, in addition to the reconstruction error, VAE leverages KL-divergence to regulate the generated distributions to a standard normal. Eventually, the *mean vectors* are treated as representations. To design AE and VAE that are directly comparable to our adversarial learning framework, we used the structure of the discriminator D (Fig. 3) as the encoder and the structure of the generator G (Fig. 2) as the decoder, that is, we replaced the vector $\boldsymbol{z}$ in the generator with the direct (AE) or sampled outputs (VAE) of the discriminator. We used element-wise Mean Square Error (MSE) to quantify the reconstruction loss.

We trained the baseline AE and VAE with the same amount of data (i.e., 2,081,325 samples), training procedure (i.e., Adam with the batch size of 64), and convergence criteria (i.e., 75,000 iterations) as ALPR. We also used the same model parameter settings except that we used 32 channels instead of 64 for the output of the first fully connected layer in the decoder (generator) because the original fully connected layer ($4096 \times 8 \times 32 \times 64$) is memory intractable. Although we cut the number of channels into half (only for the first layer of the decoder), the model capacity of AE and VAE is still larger than our adversarial framework ($4096 \times 8 \times 32 \times 32 \gg 100 \times 8 \times 32 \times 64$).

5.2 Evaluation protocol

Given a baseline feature or ALPR and a classifier, we performed a model selection for the classifier's regularization parameters via 5-fold cross-validation in the training set. We chose the parameters that achieved the best cross-validation performance and retrained the model on the whole training set. Finally, the performance on the held-out testing set was measured using the classification accuracy and the Area Under AUC Curve (AUC). To control for dimensionality, we additionally conducted experiments with reduced 384-dimensional feature vectors. 384 is the lowest dimensionality among all baselines. Principle Component Analysis (PCA) was applied for the dimensionality reduction. We considered two major classifiers, that is, logistic regression (LR) and linear support vector machine (SVM).

5.3 Results and analysis

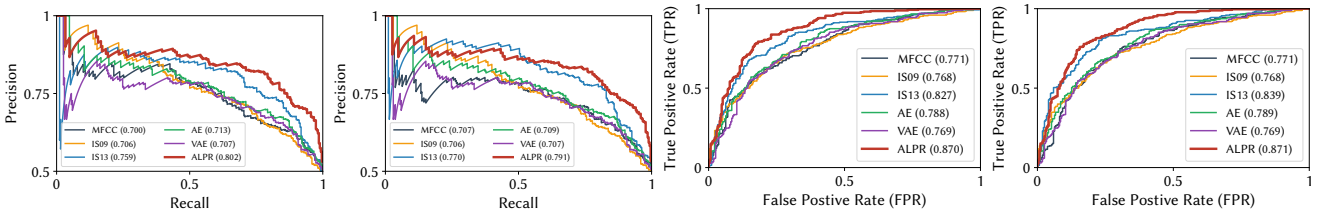
We present the Precision-Recall (PR) and ROC curves of different LR-based *seriousness* and *energy* classifiers in Fig. 7 and Fig. 8 respectively. The evaluation results demonstrate that, under most scenarios, ALPR significantly outperforms other baselines, including the state-of-the-art feature sets (MFCC, IS09, and IS13) and classical DNN-based feature learning algorithms (AE and VAE). However, under low-recall scenarios, the advantage of ALPR is inconclusive due to the sparse evaluation samples. The experiments with SVM-based classifiers produced similar results.

To more deeply understand the underlying reasons of ALPR's superior performance, we examined the quality of generated spectrograms from AE, VAE, and the adversarial learning framework. Intuitively, if generated samples are similar to real spectrograms, the discriminator may learn stronger feature representations. We generated spectrograms by sampling the generator inputs from the exact or estimated input distributions. For the adversarial learning framework and VAE, we sampled inputs from a $[-1, 1]$ uniform distribution and a standard normal distribution respectively. For AE, we first computed representations $D(\boldsymbol{x})$ for 40,000 randomly selected spectrograms from S_A, and then estimated a multivariate normal distribution through maximum likelihood, i.e., $D(\boldsymbol{x}) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\mu}$ is the mean vector, and $\boldsymbol{\Sigma}$ is the co-variance matrix. Finally, the inputs were randomly sampled from the estimated distribution.

In Fig. 9, we present generated spectrograms from randomly sampled inputs for AE, VAE and the adversarial learning framework respectively. For comparison, we also include real spectrograms randomly selected from S_A. The spectrograms generated by the adversarial network are almost indistinguishable from real ones, and contain clear details of podcast audio, such as music, pauses, conversations, and overlaps, whereas the generations from AE and VAE lose most of the fine-grained signals, and there is hardly any variation with different input vectors. This phenomenon resonates with our original motivation of using adversarial learning to capture subtle spectrogram variations.

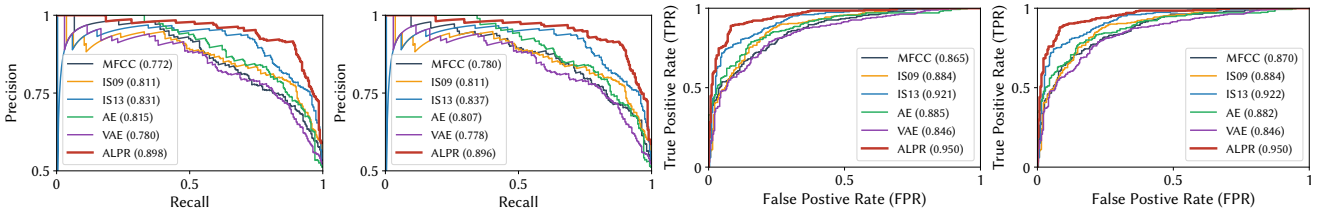
6 PREDICTING PODCAST POPULARITY

The previous experiment demonstrates that ALPR significantly outperforms existing audio modeling methods in characterizing non-textual properties of podcasts. In this section, we further investigate *to what extent ALPR can improve the performance of end applications*. Specifically, we focus on the task of popularity prediction, which could enable popularity-based recommendations for cold-start podcasts, and address fundamental questions of podcast production, such as, *what makes a podcast popular?* and *can we predict the popularity of a podcast before it goes to public?*



(a) PR curve (original dim.). **(b) PR curve (384 dim.).** **(c) ROC curve (original dim.).** **(d) ROC curve (384 dim.).**

Figure 7: Precision-Recall (PR) and ROC curves of the LR-based *seriousness* classifiers with different input features. Legends in (a) and (b) show classification accuracy, and those in (c) and (d) show AUC values. We also experimented with SVM-based classifiers, which produced similar results.



(a) PR curve (original dim.). **(b) PR curve (384 dim.).** **(c) ROC curve (original dim.).** **(d) ROC curve (384 dim.).**

Figure 8: Precision-Recall (PR) and ROC curves of the LR-based *energy* classifiers with different input features. Legends in (a) and (b) show classification accuracy, and those in (c) and (d) show AUC values. We also experimented with SVM-based classifiers, which produced similar results.

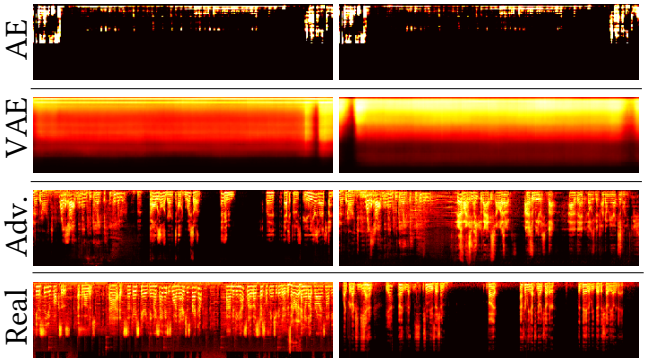


Figure 9: Real and generated podcast audio spectrograms. We compare the generations from AE, VAE and the adversarial network (Adv.) to real spectrograms randomly sampled from our dataset. The Adv. model captures more nuanced details of podcast audio spectrograms.

6.1 Evaluation dataset

We built an evaluation dataset by leveraging episodes from set S_B, which is disjoint from set S_A used in the previous experiment. To ensure the timeliness of popularity labels, we used only episodes that were published in the most recent two weeks and were from distinct channels; that is, for any podcast channel, at most one episode was included in the dataset. We defined the popularity of a channel as its ranking in the iTunes chart and collected popularity labels from the iTunes RSS feed: channels that were listed as top 200 podcasts[6] under any of the 16 categories were treated as *top channels*, and episodes from these top channels were regarded as *popular*; otherwise, episodes were *long-tail*. Finally, the evaluation dataset contains 6511 episodes, among which 837 were identified as popular. We randomly split the dataset into a training set (60%) and a testing set (40%)[7].

6.2 Baselines and evaluation protocol

To understand the utility of non-textual features beyond text, we constructed a topic-based baseline representation, referred to as TR. Specifically, we used a state-of-the-art commercial speech recognition cloud service to transcribe all episodes in set S_B and trained a topic model on the transcriptions using Mallet toolkit [32]. We chose to model 100 topics, because it produced the highest coherence score among {10, 50, 100, 200}. The trained topic model was then used to infer topic distributions for episodes in the evaluation dataset. In addition, for each episode, we extracted ALPR and IS13. IS13 was chosen because it achieved the best performance among existing features in predicting non-textual attributes. For ALPR, an episode representation was derived by taking the element-wise average of representations of chunked audio snippets (12s each). We chose such an approach because the average of vectors have been shown to be powerful in representing sets and sequences, such as sentences [23]. To make ALPR and IS13 directly comparable to TR, we used PCA to reduce their dimensionality to 100. We followed the same protocol as the previous experiment to train and

[6]The snapshot was taken at 10/02/2017.
[7]The training set contains 503 popular episodes and 3405 unpopular episodes, and the testing set contains 334 popular episodes and 2269 unpopular episodes

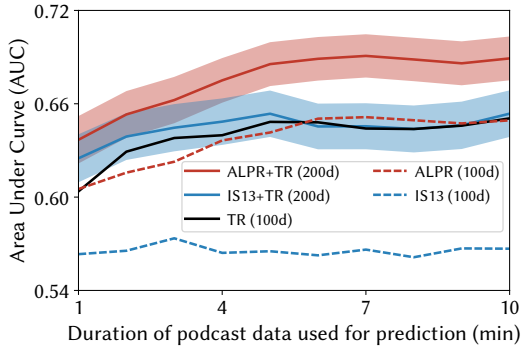


Figure 10: Podcast-popularity-prediction performance. Five feature sets are compared against AUC, and are evaluated with varied duration of podcast data for feature extraction. Shaded areas represent Standard Error of Mean (SEM).

test logistic regression-based popularity classifiers, and all features were z-normalized before feeding into classifiers.

6.3 Results and analysis

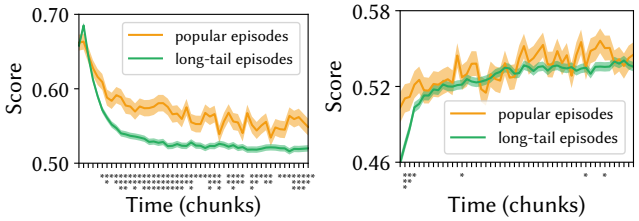
We explore how popularity-prediction performance may be affected by the duration of podcast audio that representations were computed on, that is, we varied the duration by feeding in only the leading N minutes of podcast audio, where N ranges from 1 through 10. As shown in Fig. 10, in addition to raw features, we experimented with two feature combinations, that is, ALPR+TR and IS13+TR. These results show that (1) ALPR+TR achieves significantly better performance than ALPR or TR alone, (2) IS13+TR does not outperform TR, (3) ALPR achieves competitive performance with TR and performs significantly better than IS13, and (4) incorporating more data to compute features improved performance only initially, where $N \leq 5$. In summary, ALPR brings significantly performance gain relative to predictions based on text alone, or with prior audio features. Also, the popularity of podcasts can be predicted well with the leading five minutes of audio data.

Built on these predictive models, we also investigate textual and non-textual properties that correlate with podcast popularity. For textual factors, we compute average topic distributions for popular and long-tail episodes respectively and conduct an one-sided independent t-test for every topic dimension to test whether the topic is more frequent in popular episodes than long-tail ones. To account for false positives that may result from multiple significance tests, we only report results that pass the Bonferroni-corrected significance level (i.e., $\alpha = 0.05/100 = 5\text{e-}4$). The results in Table 2 reveal topics that indicate popularity, such as crime (T1 and T2), family (T4 and T6), and politics (T9), as well topics that indicate long-tail, such as functional words (T11 and T14).

For non-textual factors, we used best-performed logistic regression models from the previous experiment (Section 5) to predict the *energy* and *seriousness* scores for chunked audio snippets of every episode. Based on predicted scores, for each time slot, we calculated an average score for popular episodes, as well as for long-tail ones. As shown in Fig. 11, in general, the *energy* level decreases over time. However, popular episodes have significantly higher *energy* levels than long-tail episodes after one minute into an episode. Such differences are not found in terms of *seriousness* levels.

ID	Topic (top words)	Sig.
T1	police, crime, case, prison, murder, found	↑↑↑
T2	war, military, army, battle, attack, force	↑↑↑
T3	story, man, thought, night, back, told	↑↑↑
T4	world, human, people, idea, sense, reality	↑↑↑
T5	food, eat, make, restaurant, good, eating	↑↑↑
T6	kids, children, child, family, parents, home	↑↑↑
T7	show, tv, shows, watch, comedy, great	↑↑↑
T8	language, english, word, words, means, speak	↑↑↑
T9	trump, president, election, donald, political, news	↑↑↑
T10	free, show, site, podcast, check, support	↑↑↑
T11	yeah, good, kind, thing, guess, pretty	↓↓↓
T12	radio, show, talk, today, great, program	↓↓↓
T13	kind, yeah, stuff, lot, cool, good	↓↓↓
T14	man, year, son, la, de, car	↓↓↓

Table 2: Topics that pass the Bonferroni-corrected significance level. For each topic, we select the top six words with the highest weights. An one-sided independent t-test is used to test whether a topic is more frequent in popular episodes than long-tail episodes (↑↑↑: $p < 0.001$, ↓↓↓: $p > 0.999$).



(a) Energy. **(b) Seriousness.**

Figure 11: The average *energy* and *seriousness* scores for popular and long-tail episodes. An average score is computed for every chunked time slot (12s each). For every slot, a two-sided independent t-test is used to test whether the mean score is different across two conditions (*: $p < 0.001$, **: $p < 0.01$, *: $p < 0.05$). Shaded areas represent SEM.**

7 CONCLUSIONS AND FUTURE WORK

We modeled non-textual characteristics of podcasts and presented an Adversarial Learning-based Podcast Representation. Through extensive experimentation with attribute-classification tasks, as well as a podcast-popularity-prediction task, ALPR was shown to significantly outperform existing approaches for capturing non-textual properties of podcasts and improving performance of end applications. In addition, we also contributed a large-scale podcast dataset that was partially labeled through crowdsourcing. This paper is an early step in building algorithms to model podcast content. Future work should address: (1) characterizing broader non-textual properties of podcast content, (2) modeling podcast content through multi-channel data fusion, and (3) studying the effectiveness of podcast content features in personalized recommendations.

ACKNOWLEDGMENTS

This research was supported by the National Science Foundation under grant IIS-1700832 and by Yahoo Research (via the Connected Experiences Laboratory at Cornell Tech). The work was further supported by the small data lab at Cornell Tech, which receives funding from NSF, NIH, RWJF, UnitedHealth Group, Google, and Adobe. We thank the anonymous reviewers for their insightful comments and suggestions.

REFERENCES

[1] Zakaria Aldeneh and Emily Mower Provost. 2017. Using regional saliency for speech emotion recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE.

[2] Thierry Bertin-Mahieux, Daniel PW Ellis, Brian Whitman, and Paul Lamere. 2011. The Million Song Dataset.. In *Ismir*.

[3] Jana Besser, Martha Larson, and Katja Hofmann. 2010. Podcast search: User goals and retrieval technologies. *Online information review* (2010).

[4] Sebastian Böck, Florian Krebs, and Gerhard Widmer. 2016. Joint Beat and Downbeat Tracking with Recurrent Neural Networks.. In *ISMIR*.

[5] Zhiyong Cheng and Jialie Shen. 2016. On effective location-aware music recommendation. *ACM Transactions on Information Systems (TOIS)* (2016).

[6] Cristian Danescu-Niculescu-Mizil, Moritz Sudhof, Dan Jurafsky, Jure Leskovec, and Christopher Potts. 2013. A computational approach to politeness with application to social factors. *arXiv preprint arXiv:1306.6078* (2013).

[7] Florian Eyben, Martin Wöllmer, and Björn Schuller. 2010. Opensmile: the munich versatile and fast open-source audio feature extractor. In *Proceedings of the 18th ACM international conference on Multimedia*. ACM.

[8] Marguerite Fuller, E Tsagkias, Eamonn Newman, Jana Besser, Martha Larson, Gareth JF Jones, M Rijke, et al. 2008. Using term clouds to represent segment-level semantic content of podcasts. (2008).

[9] John S Garofolo, Lori F Lamel, William M Fisher, Jonathan G Fiscus, David S Pallett, Nancy L Dahlgren, and Victor Zue. 1993. TIMIT acoustic-phonetic continuous speech corpus. *Linguistic data consortium* (1993).

[10] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. 2017. Audio Set: An ontology and human-labeled dataset for audio events. In *IEEE ICASSP*.

[11] Jerry Goldman, Steve Renals, Steven Bird, Franciska De Jong, Marcello Federico, Carl Fleischhauer, Mark Kornbluh, Lori Lamel, Douglas W Oard, Claire Stewart, et al. 2005. Accessing the spoken word. *International Journal on Digital Libraries* (2005).

[12] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

[13] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*.

[14] Masataka Goto and Jun Ogata. 2011. PodCastle: Recent advances of a spoken document retrieval service improved by anonymous user contributions. In *Twelfth Annual Conference of the International Speech Communication Association*.

[15] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. 2014. Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567* (2014).

[16] Negar Hariri, Bamshad Mobasher, and Robin Burke. 2012. Context-aware music recommendation based on latenttopic sequential patterns. In *Proceedings of the sixth ACM conference on Recommender systems*. ACM.

[17] Andrew F Hayes and Klaus Krippendorff. 2007. Answering the call for a standard reliability measure for coding data. *Communication methods and measures* (2007).

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

[19] Eric J Humphrey and Juan Pablo Bello. 2015. Four Timely Insights on Automatic Chord Estimation.. In *ISMIR*.

[20] Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*.

[21] Toshio Irino, Eri Takimoto, Toshie Matsui, and Roy D Patterson. 2017. An Auditory Model of Speaker Size Perception for Voiced Speech Sounds. *Proc. Interspeech 2017* (2017).

[22] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2016. Image-to-image translation with conditional adversarial networks. *arXiv preprint arXiv:1611.07004* (2016).

[23] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759* (2016).

[24] Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[25] Klaus Krippendorff. 2012. *Content analysis: An introduction to its methodology*. Sage.

[26] J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *biometrics* (1977).

[27] Martha Larson, Gareth JF Jones, et al. 2012. Spoken content retrieval: A survey of techniques and technologies. *Foundations and Trends® in Information Retrieval* (2012).

[28] Martha Larson, FMG Jong, Wessel Kraaij, and Steve Renals. 2012. Special issue on searching speech. *ACM Transactions on Information Systems (TOIS)* (2012).

[29] Bing Liu and Ian Lane. 2017. An End-to-End Trainable Neural Network Model with Belief Tracking for Task-Oriented Dialog. *Proc. Interspeech 2017* (2017).

[30] Beth Logan et al. 2000. Mel Frequency Cepstral Coefficients for Music Modeling.. In *ISMIR*.

[31] David G Lowe. 1999. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, Vol. 2. Ieee, 1150–1157.

[32] Andrew Kachites McCallum. 2002. MALLET: A Machine Learning for Language Toolkit. (2002). http://mallet.cs.umass.edu.

[33] Junta Mizuno, Jun Ogata, and Masataka Goto. 2008. A similar content retrieval method for podcast episodes. In *Spoken Language Technology Workshop, 2008. SLT 2008. IEEE*. IEEE.

[34] Vinod Nair and Geoffrey E Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*.

[35] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. 2015. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*.

[36] Douglas W Oard. 2008. Unlocking the potential of the spoken word. *Science* (2008).

[37] Jun Ogata and Masataka Goto. 2009. PodCastle: Collaborative training of acoustic models on the basis of wisdom of crowds for podcast transcription. In *Tenth Annual Conference of the International Speech Communication Association*.

[38] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: an ASR corpus based on public domain audio books. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE.

[39] Alec Radford, Luke Metz, and Soumith Chintala. 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434* (2015).

[40] Edison Research. 2017. The Podcast Consumer 2017. (2017). http://www.edisonresearch.com/the-podcast-consumer-2017/

[41] Google Research. 2017. Launching the Speech Commands Dataset. (2017). https://research.googleblog.com/2017/08/launching-speech-commands-dataset.html

[42] Anthony Rousseau, Paul Deléglise, and Yannick Estève. 2014. Enhancing the TED-LIUM Corpus with Selected Data for Language Modeling and More TED Talks.. In *LREC*.

[43] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. 2016. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*.

[44] Markus Schedl. 2015. Listener-Aware Music Recommendation from Sensor and Social Media Data. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer.

[45] Markus Schedl, Georg Breitschopf, and Bogdan Ionescu. 2014. Mobile Music Genius: Reggae at the Beach, Metal on a Friday Night?. In *Proceedings of International Conference on Multimedia Retrieval*. ACM.

[46] Markus Schedl, Peter Knees, and Fabien Gouyon. 2017. New Paths in Music Recommender Systems Research. In *Proceedings of the 11th ACM Conference on Recommender Systems (RecSys 2017). Como, Italy*.

[47] Tobias Schnabel, Igor Labutov, David M Mimno, and Thorsten Joachims. 2015. Evaluation methods for unsupervised word embeddings.. In *EMNLP*.

[48] Björn Schuller et al. 2013. The INTERSPEECH 2013 computational paralinguistics challenge: social signals, conflict, emotion, autism. (2013).

[49] Björn Schuller, Stefan Steidl, and Anton Batliner. 2009. The interspeech 2009 emotion challenge. In *Tenth Annual Conference of the International Speech Communication Association*.

[50] Aaron Van den Oord, Sander Dieleman, and Benjamin Schrauwen. 2013. Deep content-based music recommendation. In *Advances in neural information processing systems*.

[51] Longqi Yang, Eugene Bagdasaryan, Joshua Gruenstein, Cheng-Kang Hsieh, and Deborah Estrin. 2018. OpenRec: A Modular Framework for Extensible and Adaptable Recommendation Algorithms. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. ACM, 664–672.

[52] Longqi Yang, Michael Sobolev, Christina Tsangouri, and Deborah Estrin. 2018. Understanding user interactions with podcast recommendations delivered via voice. In *Proceedings of the 12th ACM Conference on Recommender Systems*. ACM, 190–194.