Disagreement Detection in Spotify Podcast Episodes

Connor Capitolo
Harvard University
connorcapitolo
@g.harvard.edu

Max Li
Harvard University
manli
@g.harvard.edu

Morris Reeves
Harvard University
morrisreeves
@g.harvard.edu

Jiahui Tang
Harvard University
jiahuitang
@g.harvard.edu

Abstract

We study the detection of disagreement in Spotify podcast episodes based on audio data and automatic transcriptions, and compare the predictiveness of prosodic and lexical features through supervised machine learning approaches. We also explore which avenues of feature engineering for audio data may be promising for disagreement detection and assess the utility of automatic transcripts. Our contribution is threefold: we introduce a first-of-its-kind annotated podcast disagreement dataset. We establish a data processing pipeline from data exploration to modeling. Lastly, we consider higher dimensional audio features that are relatively unexplored in the disagreement literature, including the spectrogram and wavelet representations.

1 Introduction

With the rise of social media platforms and online forums, the public sharing of opinions to wide audiences is easier than ever. Disagreement is ubiquitous yet also nebulous to strictly define. Disagreement can be instructive both at a societal level, where contentious topics can reflect pressing cultural issues, as well as at a dialogue level, where disagreement can reflect the stances of individual speakers.

Our goal is to generate insights about disagreement based on the Spotify English-Language 100K Podcast Dataset (Clifton et al., 2020b), consisting of audio files, transcriptions created by Google's Speech-To-Text API, and corresponding metadata. Disagreement detection may be beneficial to streaming services, users, and creators insofar as it can enhance recommendation systems for users and inform content creation. Additionally, given the increasing polarization around contentious topics such as abortion, gun rights, and election integrity, disagreement detection could be used to help streaming services moderate inflammatory podcasts that may violate their Terms of Use.

We make a multitude of contributions to the space of disagreement detection on podcast data. We have created a first-of-its-kind annotated disagreement dataset using podcast audio data as well as a complete data processing pipeline from data exploration to modeling. Relatedly, we have built baseline disagreement detection modeling approaches that can be expanded upon in future research. We have also identified different categories of disagreement that can be found in podcast data. Finally, we have determined a number of key challenges that a successful model needs to address: sparsity, data augmentation, and generalizability.

2 Related Works

2.1 Domains and media

Previous work on disagreement detection has focused primarily on modeling text data using dependency relations and lexical features such as n-grams. (Xu et al., 2019) utilize pairs of tweets with stance labels (Favor/Against/None) grouped by topic from the SemEval-2016 dataset, defining disagreement as pairs whose stance labels differ. In the realm of online forums, (Gokcen and de Marneffe, 2015) utilize quote-response pairs with agreement scores between -5 and 5 from the Internet Argument Corpus (IAC). Similarly, (Wang and Cardie, 2014) use the Authority and Alignment in Wikipedia Discussions (AAWD) dataset consisting of annotations of disagreement/agreement/neutral on English Wikipedia Talk pages.

Existing works which employ audio data for disagreement detection utilize low-dimensional prosodic features such as speech rate and fundamental frequency. For example, (Hillard et al., 2003) manually annotate transcribed ICSI meeting recordings to create labels such as positive,

backchannel, and negative. Statistics on pause duration, speaker-normalized vowel duration, and fundamental frequency are utilized alongside word-based features such as negative and positive keyword counts to make predictions. Similarly, (Wang et al., 2011) analyze English broadcast conversation data from the DARPA GALE program annotated for 11 (dis)agreement-related labels, utilizing lexical features as well as prosodic features such as pause, duration, speech rate, and pitch.

In this context, the Spotify Podcasts dataset stands out in both the volume and diversity of its audio content across various genres and podcast creators, and presents an opportunity to utilize both prosodic and lexical features to explore characteristics of human speech associated with disagreement. Additionally, (Wang et al., 2011) find sarcasm to be a common source of error in disagreement classification; the large volume of combined text and audio data in the Spotify Podcasts dataset may enable progress in this domain as well.

2.2 Definitions of disagreement

A central challenge to disagreement annotation is its inherent subjectivity; while some works in the literature provide explicit definitions of disagreement, most implicitly define disagreement through data processing decisions. (Wang et al., 2011) provide an explicit definition of (dis)agreement: it "occurs when a responding speaker agrees with, accepts, or disagrees with or rejects, a statement or proposition by a first speaker." Similar to this definition, we define disagreement as *a speaker contradicting or rejecting another speaker's idea(s) in a manner perceptible in the moment by the listener*. This allows for handling ambiguous situations such as sighs and laughter, which in some instances might not be immediately perceptible to the onset of disagreement; it also notably (and deliberately) excludes single-speaker podcast episodes, audio segments in which speakers explain the reasons for their disagreement, and audio segments in which speakers respond to soundbites.

Regarding the granularity of disagreement windows, (Wang et al., 2011) and (Wang and Cardie, 2014) define disagreement at the utterance (subturn) level, whereas (Hillard et al., 2003) annotate disagreement at the "spurt" level (period of speech by a single speaker, with no pauses longer than 0.5 seconds). Our definition involves granularity at the utterance level in order to allow for detection of sub-windows of disagreement within long uninterrupted speech segments, which is common in podcasts. Given the dialogic nature of most podcast episodes, this naturally diverges from that of (Xu et al., 2019) in which disagreement is non-dialogic (independent, differing, stance-bearing tweets on a topic).

2.3 Evaluation Metric

The F1 score (balancing the weight placed on precision and recall) is a common evaluation metric in the literature. However, for downstream use cases of disagreement detection on podcast data, it is important for streaming services that recommendations (or search results) are relevant. For example, if a user wants to find a podcast involving lively debate on the Silk Road, it is desirable that all podcasts returned from the search meet these conditions; therefore, false positives are more important to minimize than false negatives, and a higher priority should be placed on precision than recall.

Given the problem context, we make model comparisons using precision. The focus on precision also makes it easy to see how the models perform on podcasts that do not have previously annotated disagreement labels, as it is straightforward to listen to the periods where the model predicts disagreement and determine whether or not they are correctly classified.

3 Data

3.1 Overview

The Spotify Podcast Dataset consists of $105,360$ episodes across $18,360$ shows, amounting to around $50,000$ hours of audio and over 600 million words. The podcasts are sampled from Jan 1, 2019 to March 1 2020, filtered for the English language. Given the mix of both professional and amateur creators, the podcasts are diverse in terms of audio quality, topics, and structural formats. Figure 4 displays a Pareto chart of the total number of shows for each of the 23 major categories found in the metadata, with Comedy and Education being the two largest categories. Figure 5 presents a histogram of episodes less than 90 minutes long, similar to that in the data description paper (Clifton et al., 2020a); there are 125 episodes that exceeded 90 minutes in length that are not shown. However, most episodes are less than 20 minutes long, so it's unsurprising that the total number of episodes

decreases as the length increases. A large number of podcast episodes last less than 5 minutes; many of these are trailers or music remixes, in which we would not expect to find a high concentration of disagreement.

There are three different types of data in the Spotify Podcast Dataset: audio, transcript, and metadata. The transcript data contains speaker tags and timestamps for each word in the podcast. The speaker tags are automatically generated by the Google's Speech-to-Text API, which can be inaccurate when multiple speakers talk simultaneously. The podcast metadata contain features such as episode_title, author, category, subcategory, show_name, show_description, publisher, language, episode_name, episode_description, and duration. Since the metadata is provided by the show creators, some of these features may not be reliable.

3.2 Annotations

The dataset does not contain labeled disagreements, so creating them for each podcast is the first step of data preparation. Specifically, we establish a data pipeline: finding podcasts to annotate using keyword search, annotating disagreements in the podcasts, combining annotations, and discretizing text and audio data for these annotated podcasts. We create functions for the pipeline, except for the step of annotating disagreements in the podcasts; this step requires manually listening to the podcast and using Prodigy (an easy-to-use annotation tool) to label the occurrence of disagreements according to our definition of disagreement.

In total, we annotate 11 podcast episodes. These episodes amount to 206 minutes of audio, of which 193 seconds are labeled as disagreement. The average length of a labeled disagreement segment is approximately 2.62 seconds. The disagreements are very sparsely distributed within and across the episodes, meaning we have an imbalanced binary classification problem. We found the sparsity of disagreements is due to the nature of conversational podcasts, where most podcast creators look to avoid direct confrontations that may disrupt the "flow" of the podcast. During the annotation process, it was found that speakers often convey their disagreements in a playful and implicit manner, instead of expressing explicit and overt contradiction.

3.3 Transcript Coverage

Coverage in the automatic transcript data is imperfect: spoken words are occasionally missed or incorrectly transcribed. Transcript coverage can limit model performance if large portions of the missed transcript overlap with the podcast disagreement. First, we approximate the proportion of missed transcripts in the whole text dataset by calculating the ratio between the total duration of the transcribed text and the total duration of the podcast for each episode. The median transcribed episode proportion is 95.6%; the distribution of the transcribed episode proportions can be seen in Figure 7. Only episodes with calculated transcribed proportions of $<= 1$ are visualized, since those with proportion larger than 1 could result from inaccurate podcast duration records in the metadata. Most podcasts have a high transcribed proportion, suggesting that the text data has reasonable coverage.

We then examine a specific clip of a podcast and manually locate untranscribed words to examine if those segments contain disagreement; the untranscribed words with their time stamps and speaker tags can be found in Table 3. 5.36% of the total excerpt duration was not transcribed, closely matching the approximation of the missed proportion of the whole dataset. We found that most untranscribed words occur when two speakers are talking simultaneously. Words from the speaker with a lower amplitude are typically missed, implying that the transcript data could miss some parts where disagreements occurred. To examine this point, we compare the occurrence of the missed transcript and our disagreement annotation, which can be seen in Figure 8. We see that there exists some overlap between the missed scripts and our annotated disagreements.

We conclude that imperfect transcript coverage can affect the comprehensiveness and accuracy of disagreement detection based only on text data.

4 Models

4.1 Windowing and Discretization

We segmented audio and text data to create features for our models using a sliding window approach. Specifically, each window contains a podcast segment length of 2.5 seconds, with the next window sliding forward 0.5 seconds from the previous one. For instance, the first window for a podcast episode would be 0 to 2.5 seconds of the episode, and the second window would be 0.5 to 3 seconds, etc. The length of the window is determined by the average length of the labeled disagreement segment in our annotations, which is approximately 2.62 seconds

(we use 2.5 seconds for simplicity of computation and expect it to be long enough to capture the disagreements). The sliding window approach was utilized so that the data points generated would contain context information and consecutive data points would differ by roughly one word. Then, for each 2.5-second segment, we label it as disagreement if over 50% of the segment length annotated contains disagreement.

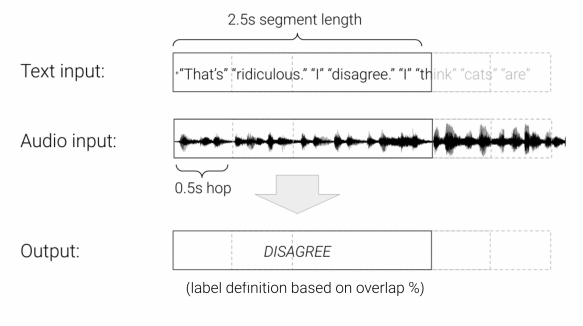

Figure 1: Windowing and Discretization

A summary of the models we use is presented in Table 4.

4.2 Text: Negation word count baseline model

As a baseline for modeling on text transcripts, we consider a Logistic Regression model which uses the count of 5 categories of negation words (in the given sliding window of 2.5 seconds) as features. We include only the words whose start and end times lie fully within the sliding window; words partially contained in a given window are not considered. To deal with class imbalance, we utilize class weights to penalize misclassification for disagreement labels.

This baseline is motivated by similar lexical features used in the literature (unigram counts), and is also consistent with the intuition that speakers use particular words to signal disagreement. The hypothesis is that higher counts of negation words in a given time window are associated with a higher probability of disagreement in that time window. Five categories of negation words are considered: analytic negation ('no', 'not'), contraction negation (e.g. 'didn't', 'shouldn't'), synthetic negation (e.g. 'neither', 'never', 'nobody'), synonyms for without (e.g. 'sans', 'without'), and other words expressing rejection ('disagree', 'incorrect', 'wrong', 'ridiculous', 'absurd').

The distribution of these words across all 8.4 million transcribed text segments (as defined by Google's Speech-to-Text API, i.e. up to 30 seconds of speech) is displayed in Figure 10. Usage of these negation words is relatively rare, and analytic negation ('no', 'not') is used more frequently than other negation categories.

4.3 Text: Averaged word2vec model

The averaged word2vec embedding model maps each word to a dense representation (300-dimensional vector) using publicly available type-based embeddings trained on part of the Google News dataset.[1] Words not in the vocabulary are taken to be the zero vector. The averaged word embeddings for each sliding window are then used as features in a Logistic Regression with class weights.

The primary motivation of the word2vec model is to address a major limitation of the negation word count baseline model: the negation categories are manually defined and may miss important words in the input. This model assumes that a linear combination of embedding dimensions is associated with the probability of disagreement.

4.4 Audio : Spectrogram-CNN model

One natural limitation of text-based approaches is the omission of nuances in intonation: intuitively, one might expect that speakers articulate disagreement with changes in pitch or speech rate in addition to words. To model patterns in frequencies across time, we translate the audio classification problem to an image classification problem and utilize the mel-spectrogram representation (of a given 2.5 second window) as features to a convolutional neural network. This approach is similar to that of (Lykartsis and Kotti, 2019), who utilize a CNN architecture for binary emotion classification (non-angry vs. angry) on audio segments using spectrogram features.

We utilize 50 mel bins, with the lower and upper frequency limits of the mel-scale set to 0 and 8000 Hz; the FFT size (frequency resolution) and FFT hop length are set to 1024 and 256, respectively. Data augmentation is applied to spectrograms using random frequency and time masking, which masks random horizontal and vertical strips of the spectrogram. Examples of resulting spectrograms are displayed in Figure 2.

The CNN architecture comprises 4 convolutional layers of 5 filters each, with a square 4x4 kernel,

[1] https://code.google.com/archive/p/word2vec/

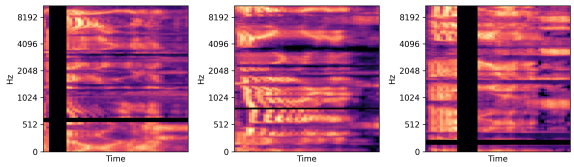

Figure 2: Sample CNN spectrograms with random frequency and time masking

stride of 1, relu activation, and same padding. Each convolutional layer is followed by 4x4 max pooling with a stride of 2. Dropout of 0.3 is applied between each max pooling layer and the subsequent convolutional layer. The flattened, batch normalized outputs are fed into a feed-forward neural network with 2 dense layers that contain 50 and 25 nodes, respectively, followed by a linear sigmoid layer.

4.5 Feature Engineering & Wavelet Transformation

The Wavelet Transform (WT) is a technique for analyzing signals. It was developed as an alternative to the Short Time Fourier Transform (STFT) to overcome problems related to its frequency and time resolution properties. A Wavelet is a wave-like oscillation that is localized in time. Wavelets have 2 fundamental properties: scale and location. Scale defines how stretched or squashed a wavelet is, while location is a position in time. The advantage that Wavelet Transforms has over Fast Fourier Transforms is that they capture spectral and temporal information simultaneously. A signal is transformed into a set of wavelets at different scales and positions (George Tzanetakis, 2001).

Wavelet transformations have two forms: discrete and continuous. The Discrete Wavelet Transform (DWT) is a special case of the WT that provides a compact representation of a signal in time and frequency that can be computed efficiently. More specifically, unlike the STFT that provides uniform time resolution for all frequencies, the DWT provides high time resolution and low frequency resolution for high frequencies as well as high frequency resolution and low time resolution for low frequencies. In this respect, it is similar to the human ear with time-frequency resolution characteristics (George Tzanetakis, 2001). Continuous wavelet transform (CWT) is similar to discrete wavelet transform except it continuously varies the values of the scale parameter.

4.5.1 Models using Discrete Wavelet Transform

We consider several binary classification models which use as features the detailed coefficients from DWT for each 2.5 second input audio segment. The length of each segment's DWT approximation coefficients is based on the sampling rate used when loading the audio file into a numpy array via the librosa library, which is a compressed and discretely sampled version of the original wavelets.

We utilize Logistic Regression, Random Forest, Boosting, and LSTM models. We explore several different boosting algorithms such as GradientBoost, AdaBoost, HistGradientBoost and XGBoost, but report only the performance of XGBoost as it is more computationally efficient and scalable while producing better performance in general.

For LSTM, the motivation is that our time series data does not meet the independent and identically distributed features assumption, even by chunking into sliding window intervals of 2.5 seconds with a hop length of 0.5 seconds. LSTM could better capture patterns in sequential data and also overcome long term dependency problems.

5 Results and Discussion

We evaluate the performance of both the text and audio-based models on two different test sets. The first test set ("Test Set 1") is a single podcast in which two Australian teenagers argue about abortion.[2] The second test set ("Test Set 2") consists of two episodes of the "Hot Take" podcast: one in which speakers argue about whether there should be no more buttons, and another about whether pie should be included in more meals in America for dinner.[3] In both cases, we train the model on all other annotated episodes.

As their descriptions suggest, the motivation for considering these two different test sets is that the former comprises more 'serious', genuine disagreement over which the speakers are emotionally invested, whereas the latter comprises more lighthearted, 'playful' disagreement. We would like to test our model on different genres and styles.

Below we report the precision of each model on these test sets: Precision 1 indicates the score on Test Set 1 and Precision 2 refers to the score on Test Set 2.

[2]Episode ID: 1XgTQnRlfJ0zpDdg2DccbR

[3]Episode IDs: 7r367wUYs1EvyBbeyOcq39, 0pIwpmg5oPcMWJXVSyrx4E

Models	Precision 1	Precision 2
Naive Baseline	0.04	0.04
Word Count	0.09	0.07
Word2Vec	0.06	0.03

Table 1: Text-based Model Comparison

Models	Precision 1	Precision 2
Naive Baseline	0.04	0.04
Spectrogram	0.04	0.06
Logistic Regression	0.04	0.02
Random Forest	0.00	0.00
XGBoost	0.00	0.00
LSTM	0.03	0.07

Table 2: Audio-based Model Comparison

The ROC curve of audio based models trained on DWT coefficients reflects performance comparable to a random classifier.

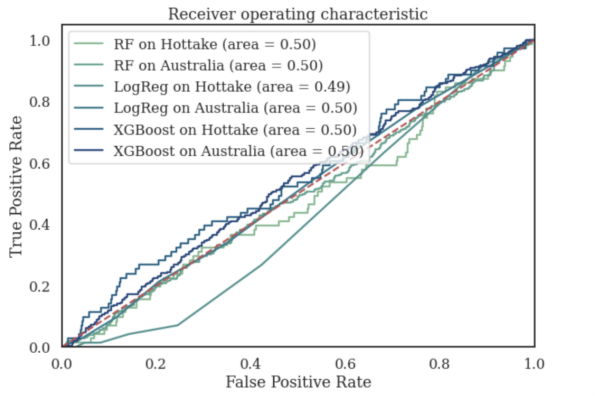

Figure 3: ROC Curve of DWT based models

From the result, we observe that text based models tend to perform better than audio based models. This may be due to the fact that text is more invariant to speakers, while audio data is quite sensitive to speakers, pitch, and background noise. In terms of feature engineering, the text-based model have far fewer features compared to audio-based ones, signaling that further regularization and tuning may be appropriate.

We also observe that current audio based models do not generalize well to different test episodes or topic domains. When performing the train-test splits within the same podcast episode, it yields much higher precision (e.g. 0.4 - 0.5) than train-test splits across episodes. With a new episode that the model has never see before, it cannot effectively predict disagreement.

One potential future improvement is pre-training a large model over a general dataset, and then fine tuning over the first few minutes of the test episode to help orient the model to the particular speaker style and genre in order to obtain better results.

In retrospect, the severe class imbalance also makes classification extremely difficult (only 4% of either test set is positive class), even with upsampling, balanced class weights and augmentation.

Another challenge is that these models treat audio segments as independent and identically distributed; however, time series of audio data has sequential information and its order matters, constraining model performance. This is the motivation for using an LSTM model to try and capture context. This gives promise for future research as we see it performs best compared to the other audio-based models.

6 Challenges and recommendations

Given the modeling challenges outlined previously (class imbalance, poor generalizability), we hypothesize a couple fruitful directions of model refinement. Our first idea is creating a feature to indicate speaker changes. Most cases of annotated disagreement occur when there is a speaker change. We hypothesize that narrowing down the possible candidates for disagreement segments can greatly enhance precision. Second, we believe performing data augmentation can reduce sensitivity to speaker characteristics. While time and frequency masking are utilized in CNN spectrogram modeling, adding colored noise and pitch shift may increase recall by encouraging models to learn disagreement-specific rather than speaker-specific features. Third, considering longer contexts and autoregressive approaches will help to keep valuable long-term information such as changes in topic or responses to previous statements. Including disagreement in previous time windows as a feature may allow models to capture the fact that disagreement may be more likely when nearby time segments also contain disagreement. Finally, the prior probability of disagreement is unlikely to be constant across a podcast episode (and particularly unlikely at podcast start). Including time within a podcast as a feature can be beneficial for predicting disagreements within a podcast.

7 Impact Statement

The deployment of a podcast disagreement detection model by a streaming service can have potential positive or negative downstream effects. The intended use of such a model is meant to benefit the users by providing more tailored podcast recommendations that can better meet their preferences; an example would be providing podcast recommendations for an individual who wants to hear a lively debate about the Silk Road. A disagreement detection model can also benefit podcast creators by driving higher audience traffic and engagement for their specific podcasts as well as provide more detailed information about their podcast listeners. This cycle benefits the streaming service as well since it helps build the company's base by producing more dedicated users and creators. Additionally, the streaming service can utilize a disagreement detection model for content moderation support to help identify podcasts that may break their code of conduct; this will in turn keep the platform safer and more inclusive.

Deploying a model that can directly affect podcast users and creators has potential negative consequences that must be taken into consideration. The podcasts used in this paper span from January 1, 2019 to March 1, 2020, meaning that conclusions reached may not directly apply to the present day podcast landscape. Related to possible generalization concerns, the eleven podcasts examined encompass a small subset of the different types and genres of podcasts that can be found on streaming service platforms. It is necessary to complete a broader study in order to better understand how models may perform on a larger corpus. If the model that is deployed is not effective or contains potential biases, this will be detrimental for podcast users and creators as well as the streaming service itself. In this vein, a disagreement detection model could potentially benefit more professional podcasts that provide clearer audio and text compared to more amateur ones; this can disadvantage individuals in lower socioeconomic situations who may not have the funds for better quality sound equipment. Another potentially disadvantaged group may be certain individuals or genres where the model wrongly predicts disagreement, potentially exhibiting the podcast and its creators in a negative light. For example, some forms of comedy that are boisterous, or individuals that have certain tones or pitches that are more likely to make them sound disagreeable, may flag these podcasts disagreement. This demonstrates that any deployment of a model that can directly affect end users must have ample testing to make sure it is effective and does not cause unnecessary harm.

8 Appendix

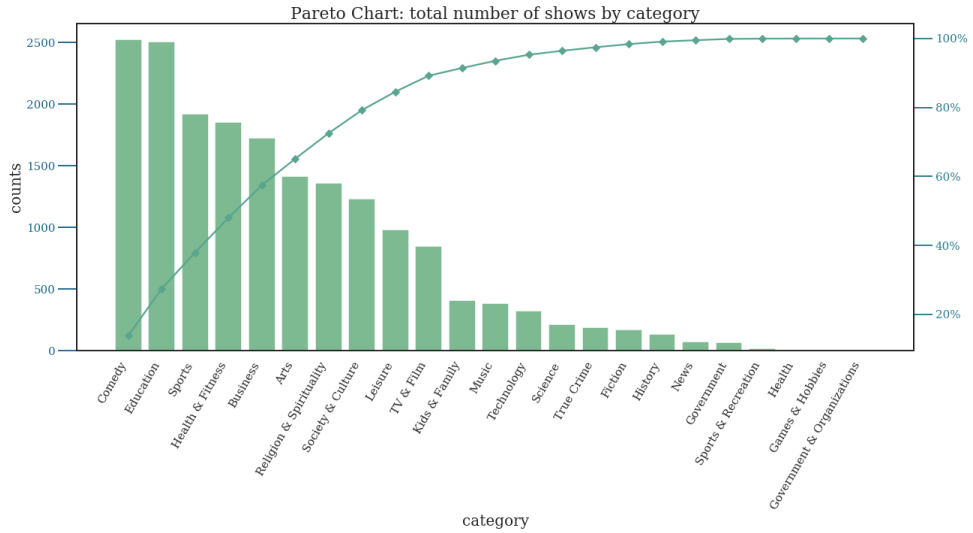

Figure 4: Pareto chart of total number of shows by category

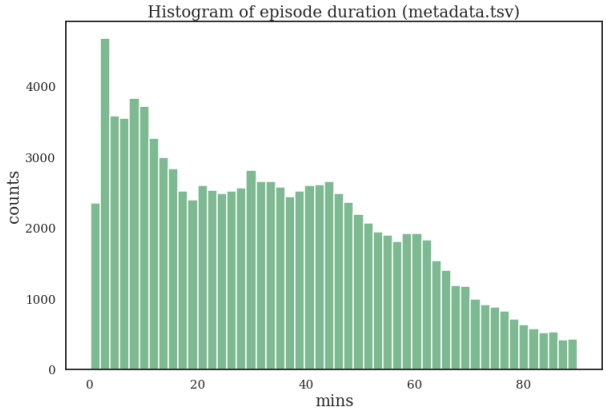

Figure 5: Histogram of episode duration (only episodes shorter than 90 minutes are included for better visualization)

Timestamp	Duration	Utterance
41:10-41:13	3s	It is. It is not
41:33-41:35	2s	Ah. Yes, they are.
42:11-42:13	2s	I understand.
42:21-42:22	1s	Definitely not
43:38-43:39	1s	It's okay.
43:45-43:47	2s	It's not right because
43:50-43:51	1s	No.
44:10-44:12	2s	Wow, that's uncalled for.

Table 3: Utterances missed in transcripts for dog-related clip example

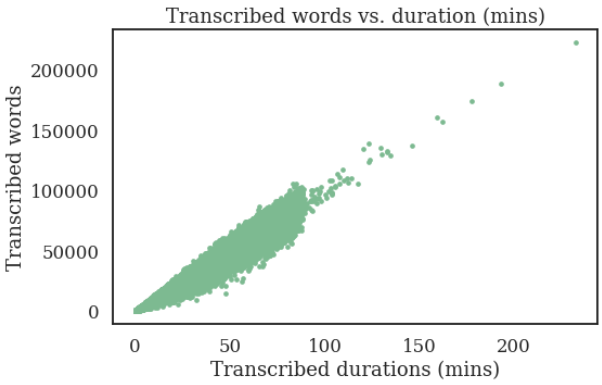

Figure 6: Episode transcribed duration vs. word count

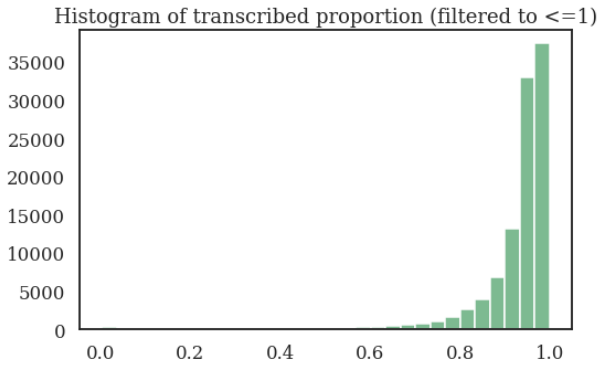

Figure 7: Distribution of transcribed proportion

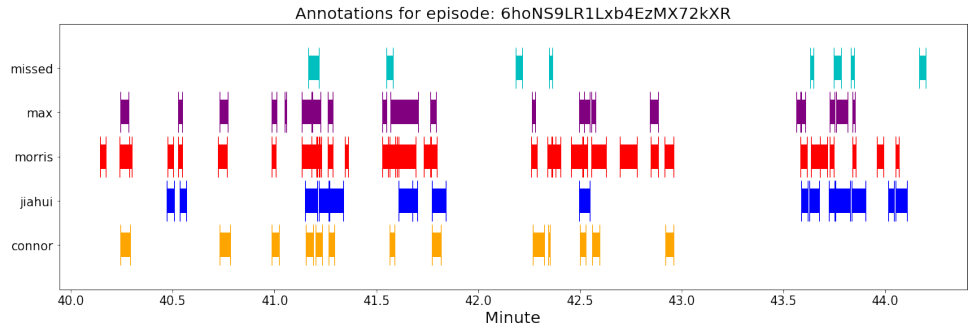

Figure 8: Comparison of utterances missed in transcripts vs. annotations

Category	Name	Input Features	Model Class	Input Data
Baseline	Naive	Random coin flip	NA	NA
Text	Word count	Negation word counts	Logistic Regression	LOO (leave one out)
	Word2vec	Avg. word2vec embedding	Logistic Regression	LOO
Audio	Spectrogram	Mel spectrogram	CNN	LOO
	Wavelet - Logreg	Discrete Wavelet Transform (DWT)	Logistic Regression	LOO
	Wavelet -RF	DWT	Random Forest	LOO
	Wavelet -XGBoost	DWT	XGBoost	LOO
	Wavelet -LSTM	DWT	LSTM	LOO

Table 4: Summary of the models

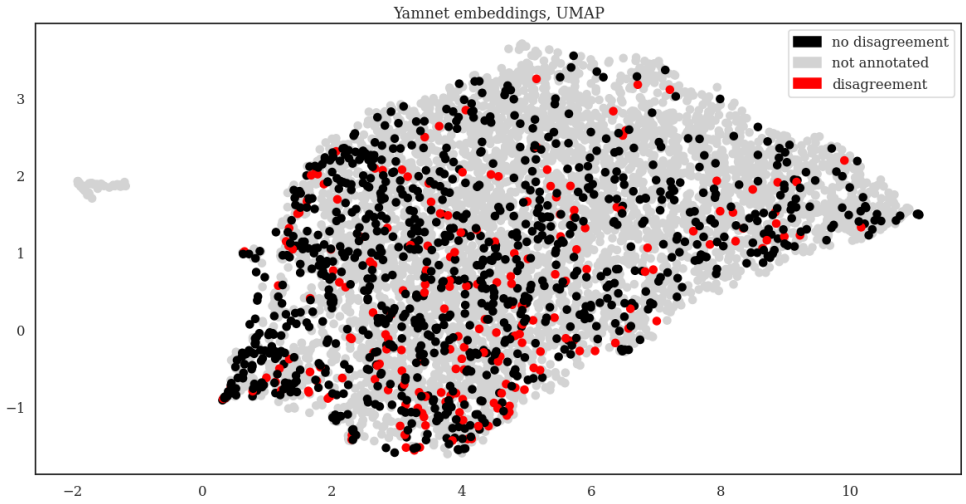

Figure 9: YAMNet Embeddings (UMAP Projection) for single annotated episode

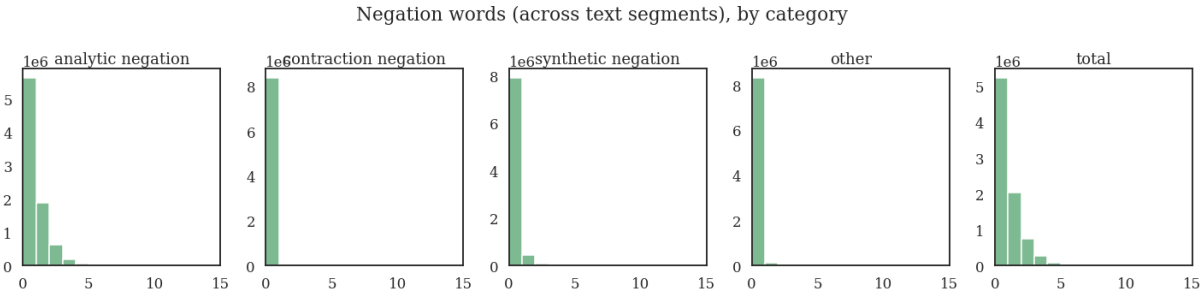

Figure 10: Distribution of negation words by category

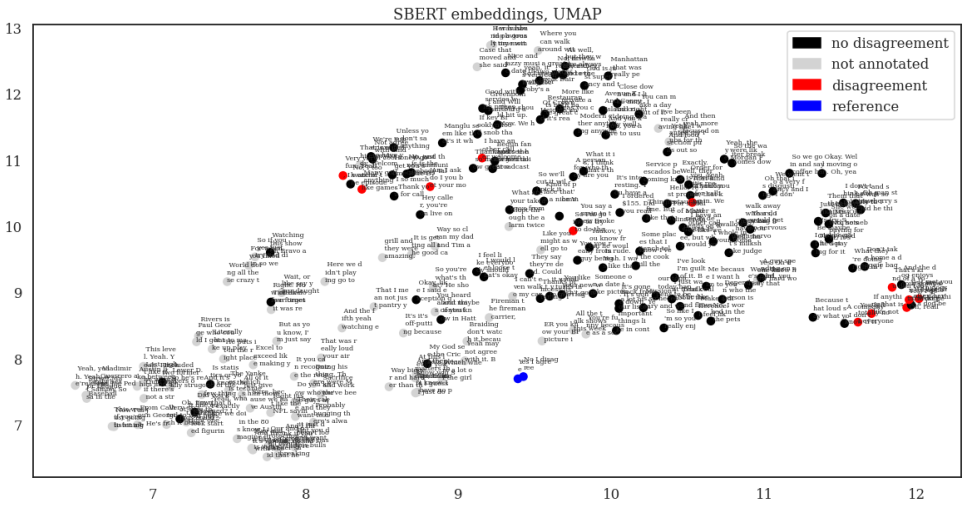

Figure 11: SBERT embeddings (UMAP projection), train and test episode

Below are additional explorations of models which we explored on a subset of the full annotated dataset (at an early stage when only 2 podcast episodes were annotated / available). Moreover, the inputs and labels for these early models do not use a sliding window approach (there is no overlap between consecutive inputs). We note that the model results are considerably higher due to the test set being an excerpt from the same podcast episode as the training set. Nonetheless, we include this section as it highlights modeling approaches which may be of interest for evaluation in future work.

8.0.1 YAMNet Embedding

We explore the YAMNet embeddings for the same podcast episode to see if there is any association between our manual disagreement annotations and these vector representations of audio segments. YAMNet is a deep neural network that employs the MobileNetV1 and predicts 521 audio event classes (trained on the AudioSet-YouTube corpus). We utilize pretrained embeddings available through Spotify; there is an embedding of dimension 1024 for each 0.48 seconds chunk. These chunks are visualized via UMAP in Figure 9 together with our annotations. There is no obvious separability based on the scatterplot when visualized in two dimensions, which we hypothesize is due to: (1) 0.48 seconds being too short a window to meaningfully expect differences in audio characteristics, (2) only a subset of YAMNet event classes may be relevant for disagreement, and even then only in rare cases (e.g. shouting would be expected to be limited to only the most extreme of disagreements), and (3) there may be higher dimensional structure to the data that is not discernible from the figure.

8.0.2 F0 (fundamental frequency) model

Intuitively, one might expect that disagreement in human speech is associated with large pitch fluctuations: that is, high variance or a high average estimated pitch may be hypothesized to be associated with disagreement. We therefore first consider a model using as features the mean and variance of the estimated fundamental frequencies in each audio segment.

Fundamental frequencies (F0) are estimated using the probabilistic YIN algorithm (Mauch and Dixon, 2014), with frame length and hop length of 2048 and 512, respectively. At a sampling rate of 22050 samples/second, this corresponds to obtaining 1 scalar F0 estimate for every $512/22050 \approx 23$ ms. The mean and variance of F0 is computed for each 5 second window, standardized (separately for train and test), and used as features to a logistic regression model. Train/test splitting is random and shuffled. These features are visualized for a sample episode excerpt in Figure 12.

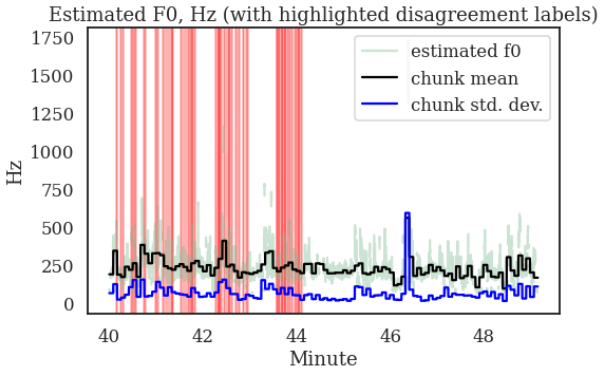

Figure 12: Estimated F0 windowed mean and std. dev. (dog disagreement episode)

Table 5 details the results. We observe that this simple F0-based model with only 2 features outperforms the precision of a random classifier by approximately 10 percentage points: the proportion of true disagreement labels in the test set is 0.32.

8.0.3 Embedding cosine similarity model

Rather than using a fixed set of words as with the count-based model, we may hypothesize that transcript segments that are similar to expressions of disagreement in some dense embedding space are more likely to contain disagreement. This forms a natural motivation for converting each transcript segment to a

Model	Precision	Recall	F1
Agreement	0.77	0.67	0.71
Disagreement	0.44	0.57	0.50

Table 5: F0 model, dog disagreement test set

vector, and classifying as disagreement based on cosine similarity to a user-defined query (we utilize "No I disagree"): we refer to this as the 'cosine similarity' model.

To convert each text transcript segment to a vector representation, we utilize the all-MiniLM-L6-v2 version of SBERT (Reimers and Gurevych, 2019), a pretrained transformer model which maps sentences paragraphs to a 384 dimensional dense vector space. The pretrained model comes tuned on a dataset of 1 billion sentence pairs (from sources such as Reddit, Stack Exchange, and Yahoo Answers) with a contrastive learning objective: predicting which of a set of randomly sampled sentences was paired with the given sentence in the dataset. By default, SBERT truncates to 128 word pieces; to convert longer text transcript segements to vectors, we take the weighted average of subsegment embeddings.

For classification, we classify a given text segment as 'disagreement' if the cosine similarity between its embedding and the embedding of 'No I disagree' exceeds 0.2. Table **??** displays the results. Although the cosine similarity model achieves higher F1 score than the lexicon reduced and full models, we observe that this is due to higher recall, which partly reflects the low cosine similarity threshold chosen.

In Figure 11, we visualize the 2D UMAP projection of the SBERT embeddings. We hypothesize that one reason for the modest performance of the cosine similarity model is that the embeddings may capture similarity of general areas of discussion rather than speakers' stances towards the areas of discussion. For example, the bottom right cloud of red points (corresponding to annotated disagreement segments) corresponds to discussion about dogs, and therefore embedding similarity may be capturing this common topic rather than a high incidence of disagreement about the topic.

References

Ann Clifton, Aasish Pappu, Sravana Reddy, Yongze Yu, Jussi Karlgren, Ben Carterette, and Rosie Jones. 2020a. The spotify podcast dataset.

Ann Clifton, Sravana Reddy, Yongze Yu, Aasish Pappu, Rezvaneh Rezapour, Hamed Bonab, Maria Eskevich, Gareth Jones, Jussi Karlgren, Ben Carterette, and Rosie Jones. 2020b. 100,000 podcasts: A spoken English document corpus. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5903–5917, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Perry Cook George Tzanetakis, Georg Essl. 2001. Audio analysis using the discrete wavelet transform.

Ajda Gokcen and Marie-Catherine de Marneffe. 2015. I do not disagree: leveraging monolingual alignment to detect disagreement in dialogue. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 94–99, Beijing, China. Association for Computational Linguistics.

Dustin Hillard, Mari Ostendorf, and Elizabeth Shriberg. 2003. Detection of agreement vs. disagreement in meetings: Training with unlabeled data. In *Companion Volume of the Proceedings of HLT-NAACL 2003 - Short Papers*, pages 34–36.

Athanasios Lykartsis and Margarita Kotti. 2019. Prediction of user emotion and dialogue success using audio spectrograms and convolutional neural networks. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 336–344, Stockholm, Sweden. Association for Computational Linguistics.

Matthias Mauch and Simon Dixon. 2014. PYIN: A fundamental frequency estimator using probabilistic threshold distributions. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2014, Florence, Italy, May 4-9, 2014*, pages 659–663. IEEE.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Lu Wang and Claire Cardie. 2014. Improving agreement and disagreement identification in online discussions with a socially-tuned sentiment lexicon. In *Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 97–106, Baltimore, Maryland. Association for Computational Linguistics.

Wen Wang, Sibel Yaman, Kristin Precoda, Colleen Richey, and Geoffrey Raymond. 2011. Detection of agreement and disagreement in broadcast conversations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 374–378, Portland, Oregon, USA. Association for Computational Linguistics.

Chang Xu, Cecile Paris, Surya Nepal, and Ross Sparks. 2019. Recognising agreement and disagreement between stances with reason comparing networks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4665–4671, Florence, Italy. Association for Computational Linguistics.