

Learning to Tag using Noisy Labels

Edith Law, Burr Settles, and Tom Mitchell

Machine Learning Department
Carnegie Mellon University
{elaw,bsettles,tom.mitchell}@cs.cmu.edu

Abstract. In order to organize and retrieve the ever growing collection of multimedia objects on the Web, many algorithms have been developed to automatically tag images, music and videos. One source of labeled data for training these algorithms are tags collected from the Web, via collaborative tagging websites (e.g., Flickr, Last.FM and YouTube) or crowdsourcing applications (e.g., human computation games and Mechanical Turk). A common approach is to use tags directly as labels for training algorithms in a supervised way. This approach is problematic, because the presence of synonyms and misspellings amongst the tags creates a label space that is overly fragmented, with a huge number of classes, many of which are sparse and semantically equivalent to one another. In this work, we investigate a method for training tagging algorithms using a reduced set of labels corresponding to topics derived from the tags. We show that our proposed method is comparable, in terms of annotation and retrieval performance, to the method of using tags directly as labels, while being more efficient to train (as there are fewer classes) and less wasteful (eliminating the need to discard tags that are associated with too few examples). We demonstrate our results using a dataset collected by a human computation game, called TagATune.

1 Introduction

Over the years, the Internet has become the largest database for multimedia objects and is organized in a rich and complex way through tagging activities. Consider music as a prime example of this phenomenon. There is now a proliferation of new applications developed to collect large number of tags for music over the Web. One example is collaborative tagging websites, such as Last.FM, which collects on the order of 2 million tags per month [20] from tens of thousands of users. Another example is human computation systems, where people contribute tags as a by-product of doing a task they are naturally motivated to perform, such as playing causal web games. For example, TagATune [22] collects tags for music by asking two players to describe their given music clip to each other, then guess whether the music clip given to them are the same or different.

In order to effectively organize and retrieve the ever growing collection of music over the Web, many automatic tag generation algorithms have been developed [3, 15, 41]. These so-called *music taggers* are useful for generating tags for songs that are rarely annotated by any Internet users, such as new music

that just emerged on the market, or existing music belonging to lesser-known artists. Once generated, these tags can be used to support music search and recommendation on a semantic level.

In previous work, the labels used to train music taggers are considered to be devoid of errors and belonging to a small fixed vocabulary, and hence, can be directly used for training. In contrast, the tags collected by collaborative tagging websites or human computation games are noisy, i.e., they can be misspelled, redundant (due to synonyms), irrelevant to content (e.g., for organizational purpose only), and unlimited in numbers. It is difficult, from a learning perspective, to know *what classes* to learn, or determine *when* the number of examples is sufficient for training a particular class. It is also computationally inefficient to train a classifier for each tag, as the vocabulary can grow to be in the tens of thousands, or millions.

In this work, we present a new technique for classifying multimedia objects by tags, that is scalable (i.e., makes full use of the huge number of noisy labels that are freely available over the Web) and efficient (i.e., the training time remains reasonably short as the tag vocabulary grows). The main idea of our technique is to organize noisy tags into well-behaved labels using topic modeling, and learn to predict tags accurately using a mixture of topic labels. Using the TagATune [22] dataset as a case study, we compare the tags generated by our proposed method (Topic Method) versus binary classification using tags directly as labels (Tag Method), both in terms of their relevance for each music clip, as well as their utility in facilitating the retrieval of relevant music by text. We also highlight a longstanding issue regarding the evaluation of music classifiers by ground truth set comparison, which is especially severe on open vocabulary tasks. Specifically, using the results from several Mechanical Turk studies, we show that human evaluations are essential for measuring the *true* performance of tag classifiers, which the traditional evaluation methods will consistently underestimate. In addition, tag diversity is found to be an important factor in human judgment of annotation quality not considered by most evaluation metrics or learning algorithms.

2 Background

2.1 Music Tagging and Retrieval

The ultimate goal of music tagging is to enable the automatic annotation of large collections of music, such that users can then browse, organize, and retrieve music in an semantic way. Although tag-based search query is arguably one of the most intuitive methods for retrieving music, until recently [3, 8, 15, 41], most retrieval methods focused on querying by metadata [46] (e.g., artist or album names), similarity [13], humming [10], beatboxing [19] and tapping [11], or using a small, fixed set of categories (e.g., genre [43, 44], mood [39], or instrumentation [14]) as keywords. The lack of focus on music retrieval by semantic tags is partly due to the lack of labeled data for training music classification algorithms.

There has been a diverse set of machine learning methods applied to the classification of music into tags, including Support Vector Machines [25, 26], Gaussian Mixture Models [40, 41], Boosting [3], Logistic Regression [2], and other probabilistic models [15]. All of these methods are trained on labels that are on the order of tens to few hundreds, as opposed to thousands to tens of thousands. For example, the Gaussian Mixture Model proposed in [40] is trained on a dataset collected from 66 paid volunteers, with 500 songs and a vocabulary size of 159 unique tags. Bertin-Mahieux et al. [3] retained only 360 of the most popular tags from Last.FM as labels for training a artist-level tag classifier. This is in contrast to the TagATune dataset used in this paper, which has over 30,000 clips, over 10,000 unique tags collected from tens of thousands of users.

2.2 Dealing with Noisy Labels

The broad problem that this work addresses is the problem of noise in datasets. Most previous work focuses on noise that is introduced when examples are misclassified into a different class [7, 36, 48], and suggest a variety of methods for discarding, correcting, or re-weighting instances that are deemed incorrectly labeled, in order to improve classification accuracy.

In our work, we address a different noise problem in datasets – the over-fragmentation of the label space due to synonyms, misspelling and compound phrases. This label noise problem is readily found in the tags produced by collaborative tagging websites (such as last.FM) [20] and human computation games such as TagATune [22], where an *open vocabulary* is allowed.

Source	Type	Example
last.FM	content irrelevant	albums I own, favorites, awesome
	synonyms	deutsch, german
	misspelling	harpsicord (harpsichord)
	compound	eclectic celtic, political hip-hop
TagATune	content irrelevant	hello, you're good too, yes agree
	synonyms	choir, choral, chorus, singing
	misspelling	chello (cello), ipano (piano) voin (violin)
	compound	country techno, guitar plucking

Table 1. Examples of Noisy Tags.

Table 1 shows examples of noisy tags from last.FM and TagATune by types. First, some tags are irrelevant to the audio characteristics of the music, and serve only the purpose of organization (e.g., “albums I own”), expression of opinions (e.g., “awesome”), or communication with the partner, in the case of games (e.g., “hello”). The second, and likely the most common, type of noise are synonyms and misspellings, which render music that should be in the same class to belong to different classes. Finally, a large portion of the tags are compound phrases with multiple descriptors. These tags tend to be highly specific, but

are associated with very few music clips. When used as labels to train a music tagger, compound tags result in classes that contain very few positive examples.

Some recent work focuses on mitigating the problem of noisy tags from collaborative tagging websites, by learning the distinction between content relevant versus content irrelevant tags [18], or by discovering higher level concepts using co-occurrence statistics in the tags [21, 24]. However, none of these work explored the use of these higher-level concepts as labels in training annotation and retrieval algorithms.

3 Problem Formulation

This section presents the music annotation and retrieval problem formally. All vector quantities are denoted in **bold**. In both problems, we are given as training data a set of N music clips $\mathcal{C} = \{c_1, \dots, c_N\}$ each of which has been annotated by humans using tags $\mathcal{T} = \{t_1, \dots, t_V\}$ from a vocabulary of size V . Each music clip $c_i = (\mathbf{a}_i, \mathbf{r}_i)$ is represented as a tuple, where $\mathbf{a}_i = \mathbb{Z}^V$ is the *ground truth tag* vector containing the frequency of each tag in \mathcal{T} that has been used to annotate the music clip by humans, and $\mathbf{r}_i = \mathbb{R}^M$ is a vector of M real-valued acoustic features, which describes the characteristics of the audio signal itself.

The goal of music annotation is to learn a function $\hat{f} : R \times T \rightarrow \mathbb{R}$, which maps the acoustic features of each music clip to a set of scores that indicate the relevance of each tag for that clip. Having learned this function, music clips can be retrieved for a search query q by rank ordering the distances between the query vector (which has value 1 at position j if the tag t_j is present in the search query, 0 otherwise) and the tag probability vector for each clip. Following [41], these distances are measured using KL divergence, which is a common measure of distance between two distributions. Note that the query vector is a valid multinomial distribution (i.e. sums to 1) for one-word queries, which are what we used to evaluate retrieval performance in this work.

4 Proposed Solution

As mention previously, most prior works train music taggers using the ground truth tags directly as labels. This training approach becomes infeasible when ground truth tags are collected by applications, such as collaborative tagging websites or human computation games, that do not enforce a controlled vocabulary. In this work, we propose an new method of generating tags, by first learning a mapping from audio features to a small set of topic labels that can cover all tags in the vocabulary, then using these high-level labels to recover the tags that are the most relevant for any music clip.

The inspiration of our approach comes from the work by Palatucci et al [34] on zero-shot learning, where the problem is to learn a classifier to predict a huge number of labels, many of which can be missing from the training set. The particular application they are interested in, is predicting the word that a person is thinking about (e.g., dog) from the fMRI image of that person's brain.

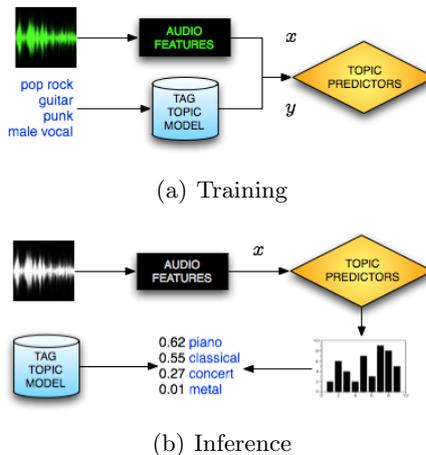


Fig. 1. The training and inference phase of the proposed model

To train such a classifier using supervised learning, one would need to create a dataset containing multiple fMRI images corresponding to each word in the English language, which would be too costly. Instead, the authors advocate an alternative method of mapping image features to a set of semantic codes that can cover all words in the English language (e.g., a boolean vector indicating the answers to questions such as “Does it breathe under water?”, “Is it slow moving?”, “Is it furry?”, “Is it carnivorous?” etc). Given a new fMRI image, the classifier can predict the semantic code of that image, then find the word in the knowledge base whose semantic code is closest to the prediction [34].

In this section, we will describe in detail the training and inference phase of our proposed method, as depicted in Figure 1.

4.1 Training Phase

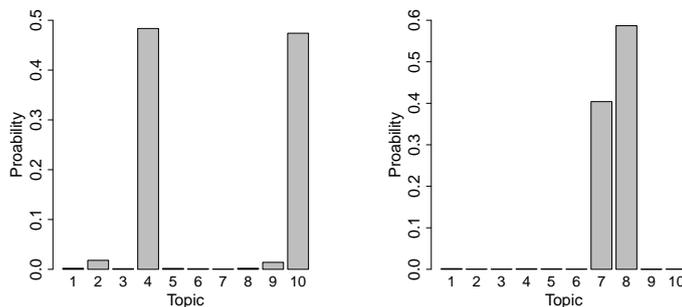
Our training phase (Figure 1(a)) is a two stage process. In the first stage, we induce a topic model using the ground truth tags associated with each music clip in the training set. This topic model allows us to infer the topic distribution of each music clip in the training set, and use these inferred topic distributions as new labels. The second stage involves training a classifier to predict topic distributions from audio features.

Stage 1: Topic Modeling using LDA

A topic model [6, 38] is a hierarchical probabilistic model that describes the process for generating the constituents of an entity (e.g., words of an article [12], musical notes in a score [17], or pixels in an image) from a set of latent topics. In the first stage of the training phase, our goal is to drastically reduce the size of

1	electronic beat fast drums synth dance beats jazz
2	male choir man vocal male_vocal vocals choral singing
3	indian drums sitar eastern drum tribal oriental middle_eastern
4	classical violin strings cello violins classic slow orchestra
5	guitar slow strings classical country harp solo soft
6	classical harpsichord fast solo strings harpsichord classic harp
7	flute classical flutes slow oboe classic clarinet wind
8	ambient slow quiet synth new_age soft electronic weird
9	rock guitar loud metal drums hard_rock male fast
10	opera female woman vocal female_vocal singing female_voice vocals

(a) Topic Model



(b) woman, classical, classssical, opera, male, violen, violin, voice, singing, strings, italian

(c) chimes, new age, spooky, flute, quiet, whsitling, whistle, fluety, ambient, chime, snare, soft, high pitch, bells

Fig. 2. An example of a topic model learned over music tags, and the representation of two music clips by topic distribution.

the label space, from thousands of *tag* labels to tens of *topic* labels, by learning a set of topics over the ground truth music tags that were collected by TagATune.

In our topic model, each topic is a distribution over music tags, and each music clip is associated with a set of topics with different probabilities. Figure 2(a) shows an example of a topic model (with 10 topics) learned over the music tags collected by TagATune. Figure 2(b) and Figure 2(c) show the topic distributions for two very distinct music clips and the ground truth tags associated with them (in the caption). The music clip represented by Figure 2(b) is associated with topic 4 (the “classical violin” topic) and topic 10 (the “female opera singer” topic), and the music clip represented by Figure 2(c) is associated with topic 7 (the “flute” topic) and topic 8 (the “quiet ambient music” topic).

In this work, we adopt a widely used method in topic modeling called the Latent Dirichlet Allocation (LDA) [6], as depicted in Figure 3. Given N music clips, V unique tags, and K topics, LDA is a probabilistic *latent variable model*, where the observed variables (shaded in grey) are $a_{i,j}$, the ground truth tags associated with music clip c_i , and the hidden variables to be inferred (circled in

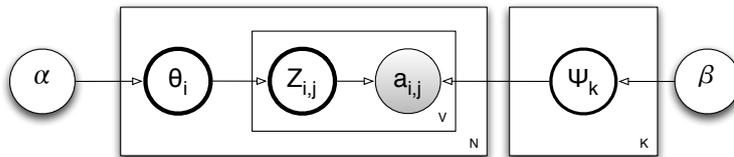


Fig. 3. Latent Dirichlet Allocation Model.

bold) are: (i) θ_i , the topic distribution for each music clip c_i , (ii) Ψ_k , the probability of each ground truth tag a_j in topic k , and (iii) $Z_{i,j}$ the topic responsible for generating the ground truth tag $a_{i,j}$ for music clip c_i , where $i = 1, \dots, N$ and $j = 1, \dots, V$, and $k = 1, \dots, K$.

The central innovation in LDA, over other topic model formulations such as Probabilistic Latent Semantic Indexing (pLSI) [16], is the use of a Dirichlet prior on the topic distribution θ_i (with hyperparameters $\alpha = \alpha_1 = \dots = \alpha_K$) and on the tag distribution Ψ_k for each topic (with hyperparameter β). These hyperparameters are fixed.

Together, LDA specifies a joint distribution over observed and hidden variables. The inference problem, then, is to learn the parameters of the posterior probability distribution of the hidden variables $(\theta_i, \Psi_k, Z_{i,j})$ conditioned on the observed data $(a_{i,j})$ and the hyperparameters (α, β) . Because it is intractable to learn this posterior distribution exactly, approximate methods (e.g., Mean Field Variational Inference [4], Gibbs Sampling [38]) have been used to solve LDA. The particular implementation used in this work is provided by the Mallet toolkit [30], which uses the Gibbs Sampling method specified in Steyvers et al [38].

LDA provides an interesting generative story about how players of TagATune might have generated the tags for the music clips they are listening to. According to the model, each player of TagATune would have a topic structure in mind when describing music. Given a music clip, the player first selects a topic according to the topic distribution for that clip, then generates a tag according to the tag distribution of the chosen topic. Under this interpretation, our goal in building a topic model over tags is to discover the topic structure that the players used to generate tags for music, so that we can leverage a similar topic structure to automatically tag new music.

Stage 2: Topic Distribution Classification by Maximum Entropy

The topic model derived in stage 1 of the training phase can be used to assign a *ground truth topic distribution* to each music clip. In the second stage, our goal is to learn a function g that maps audio features to topic distributions, using the ground truth topic distributions as labels for training. Our classifier of choice is Maxent (maximum entropy classifier) [9], which has been used extensively in text classification [1, 32], but to our knowledge, rarely for music tagging. The particular implementation we adopted is from the Mallet toolkit [30], which uses

Limited Memory BFGS [33] to maximize the likelihood of the parameters, and a slight modification of the optimization procedure provided by Yao et al [47] to enable the use of topic distributions, instead of a single topic, as labels for training the classifier.

4.2 Inference Phase

Figure 1(b) depicts the process of generating tags for new music clips. For an unseen music clip c' and given only its audio features, we can use the function g learned in stage 2 of the training phase to infer the topic distribution of that clip. Given this predicted topic distribution, each tag can be given a relevance score for the music clip c' , by multiplying the probability of that tag in each topic and the probability of that topic in c' , summing over all topics, i.e.

$$p(t_j|r_i) = \sum_k^K p(t_j|z = k) \cdot p(z = k|r_i)$$

where $j = 1, \dots, V$, $i = 1, \dots, N$ and $k = 1, \dots, K$. In reality, there are many different ways to generate tags from a topic model. For example, one can add a restriction that says that the generated tags can only come from the top Q topics, where $Q \ll K$. In future work, we may experiment with different inference schemes, and compare their effectiveness in generating relevant tags for music.

5 Experiment

Our goal is to compare our proposed method (Topic Method) against the methods of generating tags using binary classification (Tag Method) or at random (Random Method), using 5-fold cross validation. The experiments are guided by five central questions:

- Feasibility* Given a set of noisy music tags, is it possible to learn a reduced representation of the tag space that is (i) semantically meaningful, and (ii) predictable by content-based features (e.g., timbre, rhythm etc) of the music?
- Annotation* How accurate are the generated tags?
- Retrieval* How well do the generated tags facilitate music retrieval?
- Efficiency* How do the training times compare between methods?
- Evaluation* To what extent are the evaluations a reflection of the true performance of the tag classifiers?

5.1 Dataset

The data is collected via a two-player online game called TagATune [22]. Figure 4 shows the interface of TagATune. In this game, two players are given either the same or different music clips, and are asked to describe their given music clip. Upon reviewing each other’s description, they must guess if the music clips are the same or different. Since its deployment in May 2008, TagATune has collected over a million tags from tens of thousands of users.



Fig. 4. TagATune

There exist several human computation games [28, 42] that collect tags for music that are based on the *output-agreement mechanism* (a.k.a. the ESP Game [45] mechanism), where two players must match on a tag in order for the tag to become a valid label for a music clip. In our previous work [22], we have shown that output-agreement games, although effective for image annotation, are restrictive for music data: there are so many ways to describe music and sounds that players often have a difficult time agreeing on any tags. In TagATune, the problem of agreement is alleviated by allowing players to communicate with each other. Furthermore, by requiring that the players guess whether the music are the same or different based on each other’s tags, the quality and validity of the tags are ensured. The downside of opening up the communication between players is that the tags entered are more noisy.

Figure 5 shows the characteristics of the TagATune dataset, in terms of how many ground truth tags each music clip has, and how many music clips are available to each tag as training examples. Figure 5(a) is a rank frequency plot showing the number of music clips (y-axis) that have a certain number of ground truth tags (x-axis). The plot reveals that a majority of the music clips (> 1500) have under 10 ground truth tags, with around 1300 music clips with only 1 or 2 ground truth tags, and very few music clips that have a large number (e.g.,

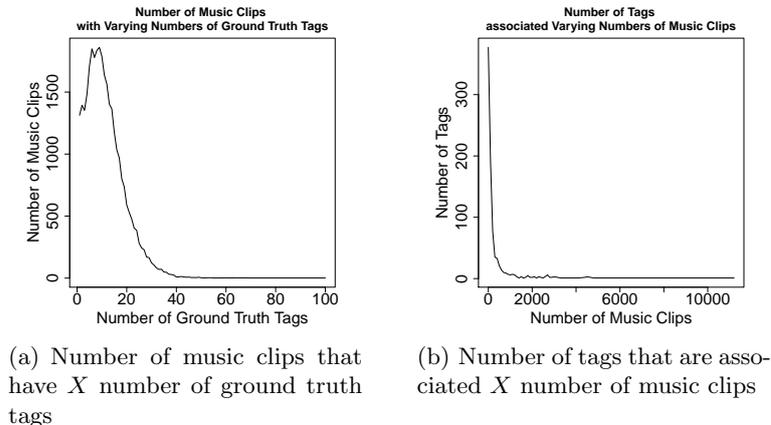


Fig. 5. Characteristics of the TagATune Dataset

> 100) of ground truth tags. This disparity in the number of ground truth tags creates a problem in our evaluation – many of the generated tags will not be found amongst the ground truth tags, and therefore will be considered incorrect when they are in fact correct. Figure 5(b) is a rank frequency plot showing the number of tags that have a certain number of music clips available to them as training examples. The plot shows that the vast majority of the tags have few music clips to use as training examples, while a small number of tags are endowed with a large number of examples. This highlights the aforementioned sparsity problem that emerges when tags are used directly as labels, a problem that is addressed by our proposed method.

We did a small amount of pre-processing on a subset of the data collected by TagATune until April 2009, tokenizing tags, removing punctuation and four extremely common tags that are not related to the content of the music, i.e. “yes”, “no”, “same”, “diff”. These tags are natural consequences of the game, since players communicate with each other in other ways beyond just describing the music [22], such as saying “yes” or “no” to confirm whether the partner’s tags also describe one’s own music clip, or “same” or “diff” to notify the partner of the player’s current guess of whether the music is the same or different.

We also eliminated tags that have fewer than 20 music clips available as training examples, in order to conduct a comparison against the Tag Method, which requires sufficient amount of training examples for each binary classification task. This reduces the number of music clips from 31867 to 31251, and the total number of ground truth tags from 949,138 to 699,440, and the number of *unique* ground truth tags from 14506 to 854. For the purpose of comparison, this reduced set of ground truth tags is used in both the Topic Method and the Tag Method. Note that we are throwing away a substantial amount of tag data when we require that each tag be associated with a minimum number of examples. A motivation for using topic models to generate tags is that we do not need to

throw away any tags at all. Rare tags, i.e. tags that are associated with only one or two music clips, can still be grouped into a topic, and used in the annotation and retrieval process.

Each of the 31251 music clips is 29 seconds in duration, and is represented by a set of ground truth tags collected via TagATune, as well as a set of content-based (spectral and temporal) features extracted using the technique described in [27]. Spectral features consist of summary statistics (mean and covariance) of a clip’s Mel-Frequency Cepstral Coefficients (MFCC), which describe the power spectrum of an audio signal on a scale composed of frequencies that are meaningful to human hearing. Temporal features describe the total magnitude of different frequency levels over time. The detail of this feature extraction scheme is available in [27].

5.2 Experiment 1: Feasibility

Table 2 shows the top 10 words of each topic learned by LDA using the tags collected via TagATune with the number of topics fixed at 10, 20 and 30. In general, the topics are able to capture meaningful grouping of tags, e.g., synonyms (e.g., {“choir”, “choral”, “chorus”}, or {“male”, “man”, “male_vocal”, “male_voice”}), misspellings (e.g., {“harpsichord”, “harpsicord”} or {“cello”, “chello”}), or associations (e.g., {“indian”, “drums”, “sitar”, “eastern”, “tribal”, “oriental”} or {“rock”, “guitar”, “loud”, “metal”}). As we increase the number of topics, there emerge new topics that are not captured by topic models with fewer number of topics. For example, in the topic model with 20 topics, topic 3 (which describes soft classical music), topic 13 (which describes jazz), topic 17 (which describes rap, hip-hop and reggae) are new topics that are not evident in the topic model with 10 topics. We also observe some repetition (or refinement) of topics as the number of topic increases (e.g., topics 8, 25 and 27 in the 30-topic model all describe female vocal music, but are slightly different in terms of genre).

It is difficult to know exactly how many topics can succinctly capture the concepts underlying the music in our dataset. For all our experiments, we empirically tested how well topic distribution and the best topic can be predicted using audio features, fixing the number of topics at 10, 20, 30, 40, and 50 topics. Figure 6 summarizes the results. We evaluated performance using several metrics, including accuracy and average rank of the most relevant topic, as well as the KL divergence between the ground truth and the predicted topic distribution. Although we see a degradation of performance as the number of topics increases, all models (under the accuracy, average rank, KL divergence metrics) significantly outperform the random baseline, which uses random distributions as labels for training. Moreover, even with 50 topics, the average rank of the most relevant topic is still around 3, which suggests that the classifier is capable of predicting the most relevant topic well. This is crucial, as the most appropriate tags for a music clip are likely to be found in the most relevant topics for that clip.

10 Topics	
1	electronic beat fast drums synth dance beats jazz electro modern
2	male choir man vocal male_vocal vocals choral singing male_voice pop
3	indian drums sitar eastern drum tribal oriental middle_eastern foreign fast
4	classical violin strings cello violins classic slow orchestra string solo
5	guitar slow strings classical country harp solo soft quiet acoustic
6	classical harpsichord fast solo strings harpsicord classic harp baroque organ
7	flute classical flutes slow oboe classic clarinet wind pipe soft
8	ambient slow quiet synth new_age soft electronic weird dark low
9	rock guitar loud metal drums hard_rock male fast heavy male_vocal
10	opera female woman vocal female_vocal singing female_voice vocals female_vocals voice

20 Topics	
1	indian sitar eastern oriental strings middle_eastern foreign guitar arabic india
2	flute classical flutes oboe slow classic pipe wind woodwind horn
3	slow quiet soft classical solo silence low calm silent very_quiet
4	male male_vocal man vocal male_voice pop vocals singing male_vocals guitar
5	cello violin classical strings solo slow classic string violins viola
6	opera female woman classical vocal singing female_opera female_vocal female_voice operatic
7	female woman vocal female_vocal singing female_voice vocals female_vocals pop voice
8	guitar country blues folk irish banjo fiddle celtic harmonica fast
9	guitar slow classical strings harp solo classical_guitar soft acoustic spanish
10	electronic synth beat electro ambient weird new_age drums electric slow
11	drums drum beat beats tribal percussion indian fast jungle bongos
12	fast beat electronic dance drums beats synth electro trance loud
13	jazz jazzy drums sax bass funky guitar funk trumpet clapping
14	ambient slow synth new_age electronic weird quiet soft dark drone
15	classical violin strings violins classic orchestra slow string fast cello
16	harpsichord classical harpsicord strings baroque harp classic fast medieval harps
17	rap talking hip_hop voice reggae male male_voice man speaking voices
18	classical fast solo organ classic slow soft quick upbeat light
19	choir choral opera chant chorus vocal vocals singing voices chanting
20	rock guitar loud metal hard_rock drums fast heavy electric_guitar heavy_metal

30 Topics	
1	choir choral opera chant chorus vocal male chanting vocals singing
2	classical solo classic oboe fast slow clarinet horns soft flute
3	rap organ talking hip_hop voice speaking man male_voice male_man_talking
4	rock metal loud guitar hard_rock heavy fast heavy_metal male punk
5	guitar classical slow strings solo classical_guitar acoustic soft harp spanish
6	cello violin classical strings solo slow classic string violins chello
7	violin classical strings violins classic slow cello string orchestra baroque
8*	female woman female_vocal vocal female_voice pop singing female_vocals vocals voice
9	bells chimes bell whistling xylophone whistle chime weird high_pitch gong
10	ambient slow synth new_age electronic soft spacey instrumental quiet airy
11	rock guitar drums loud electric_guitar fast pop guitars electric bass
12	slow soft quiet solo classical sad calm mellow very_slow low
13	water birds ambient rain nature ocean waves new_age wind slow
14	irish violin fiddle celtic folk strings clapping medieval country violins
15	electronic synth beat electro weird electric drums ambient modern fast
16	indian sitar eastern middle_eastern oriental strings arabic guitar india foreign
17	drums drum beat beats tribal percussion indian fast jungle bongos
18	classical strings violin orchestra violins classic orchestral string baroque fast
19	quiet slow soft classical silence low very_quiet silent calm solo
20	flute classical flutes slow wind woodwind classic soft wind_instrument violin
21	guitar country blues banjo folk harmonica bluegrass acoustic twangy fast
22	male man male_vocal vocal male_voice pop singing vocals male_vocals voice
23	jazz jazzy drums sax funky funk bass guitar trumpet reggae
24	harp strings guitar dulcimer classical sitar slow string oriental plucking
25*	vocal vocals singing foreign female voices women woman voice choir
26	fast loud upbeat quick fast_paced very_fast happy fast_tempo fast_beat faster
27*	opera female woman vocal classical singing female_opera female_voice female_vocal operatic
28	ambient slow dark weird drone low quiet synth electronic eerie
29	harpsichord classical harpsicord baroque strings classic harp medieval harps guitar
30	beat fast electronic dance drums beats synth electro trance upbeat

Table 2. Topic Model with 10, 20, and 30 topics. The topics in bold in the 20-topic model are examples of new topics that emerge when the number of topics is increased from 10 to 20. The topics marked by * in the 30-topic model are examples of repeated or refined topics that emerge as the number of topics is increased.

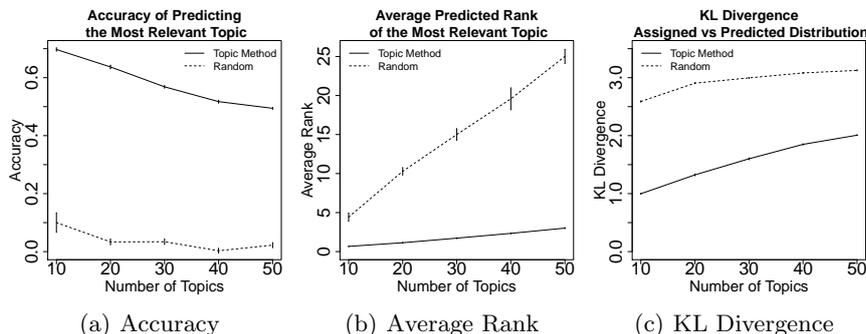


Fig. 6. Results showing how well topic distributions or the best topic can be predicted from audio features. The metrics include accuracy and average rank of the most relevant topic, and KL divergence between the assigned and predicted topic distribution.

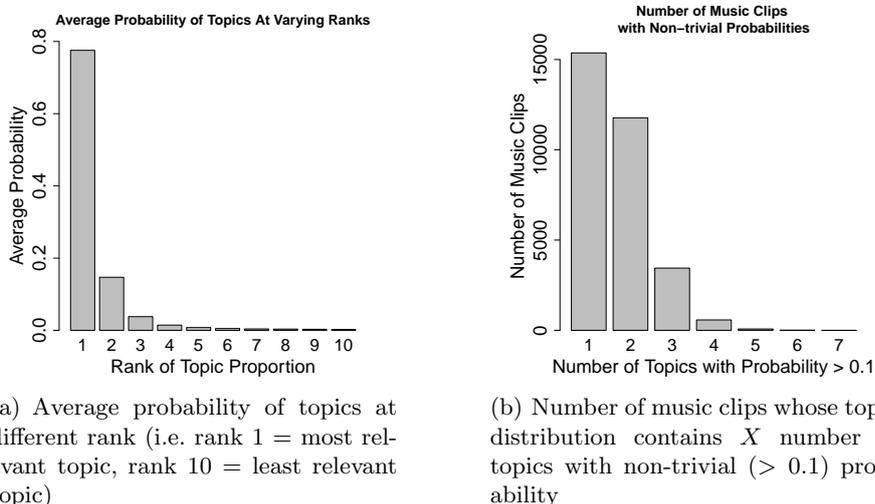


Fig. 7. Most music clips are assigned only 1 or 2 topics with non-trivial probabilities.

We also experimented with using the most relevant topic as the label to train the maximum entropy classifier, and observe that it produced the same results as using the topic distribution as a label for training. There are two possible explanations. First, Yao et al [47] reported a similar observation, that the “output of the topic proportion classifier is often overly concentrated on the single largest topic”. Therefore, this phenomenon can be an artifact of the particular classifier and optimization method we used. Second, we observe that for most music clips in the TagATune dataset, the topic model assigns very high probabilities to only a few topics, and low probabilities for all other topics.

Figure 7(a) shows the probability of topics at different rank (rank 1= most relevant topic, rank 10 = least relevant topic), averaged over all music clips. It reveals that the most relevant topic has average probability of approximately 0.7, followed by the second ranking topic with probability < 0.2 , and the third ranking topic with probability < 0.05 , and the rest of the topics with very small probabilities. Figure 7(b) is a rank frequency plot showing the number of music clips whose topic distribution have X number of topics with non-trivial (> 0.1) probability. It is evident that for the majority of music clips in the TagATune dataset, their topic distributions contain only 1 or 2 topics with non-trivial probabilities.

5.3 Experiment 2: Annotation Performance

Following [15], we evaluate the accuracy of the top 10 tags for each music clip, under three different metrics: per-clip metric, per-tag metric and omission-penalizing per-tag metric.

Per-Clip Metric

The per-clip precision@ N metric measures the proportion of correct tags (according to agreement with the ground truth set) amongst the N tags that have the highest inferred probabilities for each clip, averaged over all the clips in the test set. The results are presented in Figure 8.

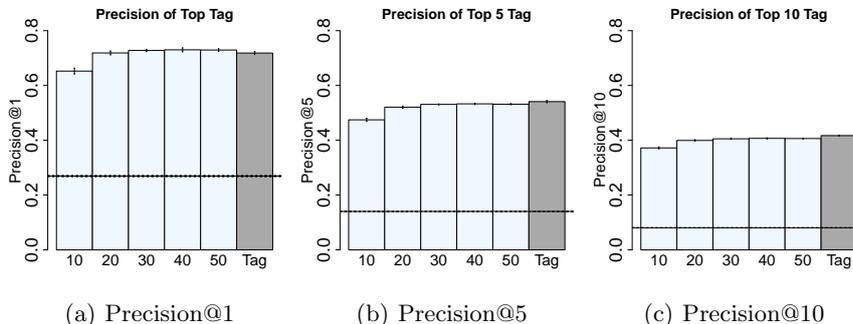


Fig. 8. Per-clip Metrics. The light-colored bars represent Topic Method with 10, 20, 30, 40 and 50 topics. The dark-colored bar represents the Tag Method. The horizontal line represent the random baseline, and the dotted lines represent its standard deviation.

Topic Model (using 50 topics) and the Tag Method are almost indistinguishable under this metric.

Per-Tag Metric

Alternatively, we can evaluate annotation performance by computing the precision, recall and F-1 measures for each tag, averaged over all the tags that are outputted by the algorithm (i.e. if the music tagger does not output a tag, the scores for that tag are simply ignored). Specifically, given a tag t , its precision P_t , recall R_t and F-1 measure F_t can be computed as follows:

$$P_t = \frac{c_t}{a_t} \quad R_t = \frac{c_t}{g_t} \quad F_t = 2 \cdot \frac{P_t \cdot R_t}{P_t + R_t}$$

where g_t is the number of music clips that has the tag t in their ground truth sets, a_t is the number of clips that are annotated with the tag t by the tagger, and c_t is the number of clips that has been *correctly* annotated with the tag t by the tagger, according to the ground truth set. The overall per-tag precision, recall and F-1 scores for a test set are P_t , R_t and F_t for each tag t , averaged over all tags in the vocabulary.

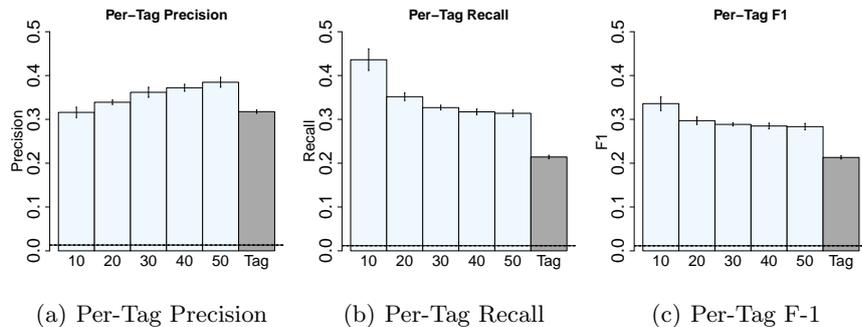
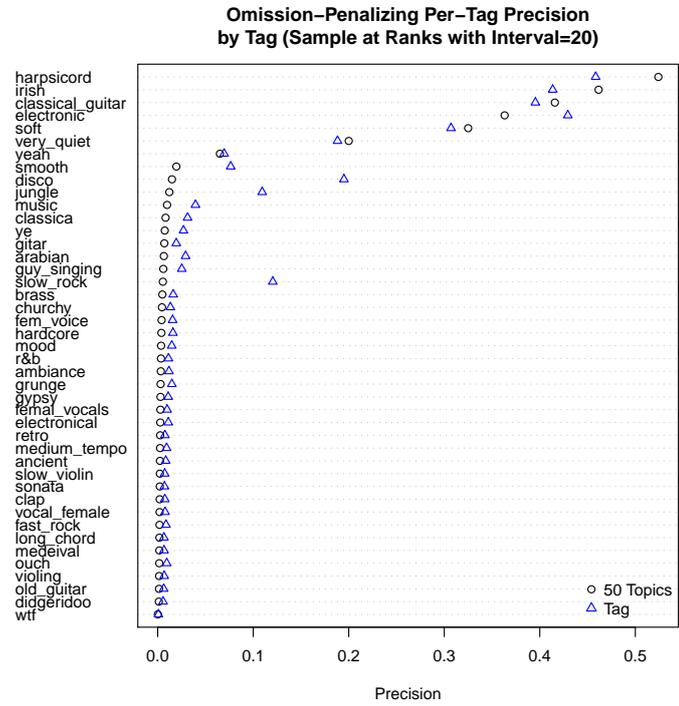
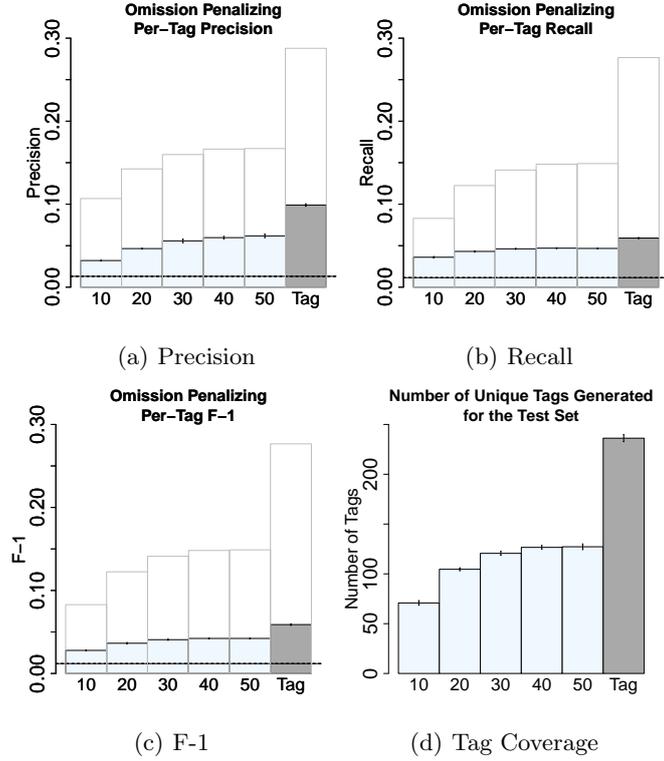


Fig. 9. Per-tag Metrics. The light-colored bars represent Topic Method with 10, 20, 30, 40 and 50 topics. The dark-colored bar represents the Tag Method. The horizontal line represent the random baseline, and the dotted lines represent its standard deviation.

Results (in Figure 9) show that the Topic Method significantly outperforms the Tag Method under this set of metrics.

Per-Tag Metric (Omission Penalizing)

Although informative, two of the metrics – per-clip precision@N and per-tag precision – are problematic in that a system can output the most common tags, leaving out the rare ones, and still perform reasonably well under these metric [41]. In response to this criticism, several previous work [3, 15, 41] has adopted a set of *per-tag* metrics that penalizes algorithms for omitting tags that could have been used to annotate music clips in the test set.



(e) Precision by Tag

Fig. 10. Omission-Penalizing Per-tag Metrics. The light-colored bars represent Topic Method with 10, 20, 30, 40 and 50 topics. The dark-colored bar represents the Tag Method. The horizontal line represent the random baseline, and the dotted lines represent its standard deviation. Figure (e) shows the precision of individual tags at rank 1, 21, 41, \dots , 854 etc. It is evident that the Topic Method loses in precision by failing to output many of the rarer tags.

Following [15, 41], the omission-penalizing per-tag precision and recall can be computed as follows:

$$P_t = \begin{cases} \frac{c_t}{a_t} & \text{if present} \\ E_t & \text{if omitted} \end{cases} \quad R_t = \begin{cases} \frac{c_t}{g_t} & \text{if present} \\ 0 & \text{if omitted} \end{cases}$$

where E_t is the empirical frequency of the tag t in the test set. This specification penalizes classifiers that leave out tags, especially ones that are rare. Note that these metrics are upper bounded by a quantity that depends on the number of tags outputted by the algorithm. This quantity can be computed empirically by setting the precision and recall to 1 when the tag are present, and E_t and 0 when a tag is omitted.

Results (Figure 10 (a)–(d)) shows that for the Topic Method, performance increases with more topics, but reaches a plateau as the number of topics approaches 50. We investigated additional models with 60, 70, 80, 90, and 100 topics, and found that this plateau persists in these models. In particular, Figure 11(a) shows that under the per-tag metric, precision keeps increasing when we increase the number of topics, but recall hits a plateau. The same performance plateau is observed under the omission-penalizing per-tag metric (Figure 11(b)).

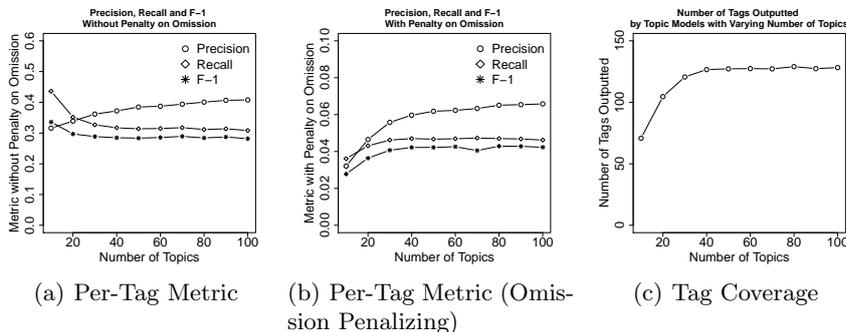


Fig. 11. How performance varies as the number of topics increases.

The performance plateau can be attributed to the fact that the number of tags outputted by the topic models plateau at around 127 (Figure 11(c)). This is a somewhat expected, and problematic, behavior of the Topic Method, where common tags (e.g., classical) tend to be ranked higher in any given topic, and therefore, are more likely to be generated. The plateau also explains why the Tag Method outperforms the Topic Method under this metric – it generated roughly twice the number of unique tags (Figure 10(d)).

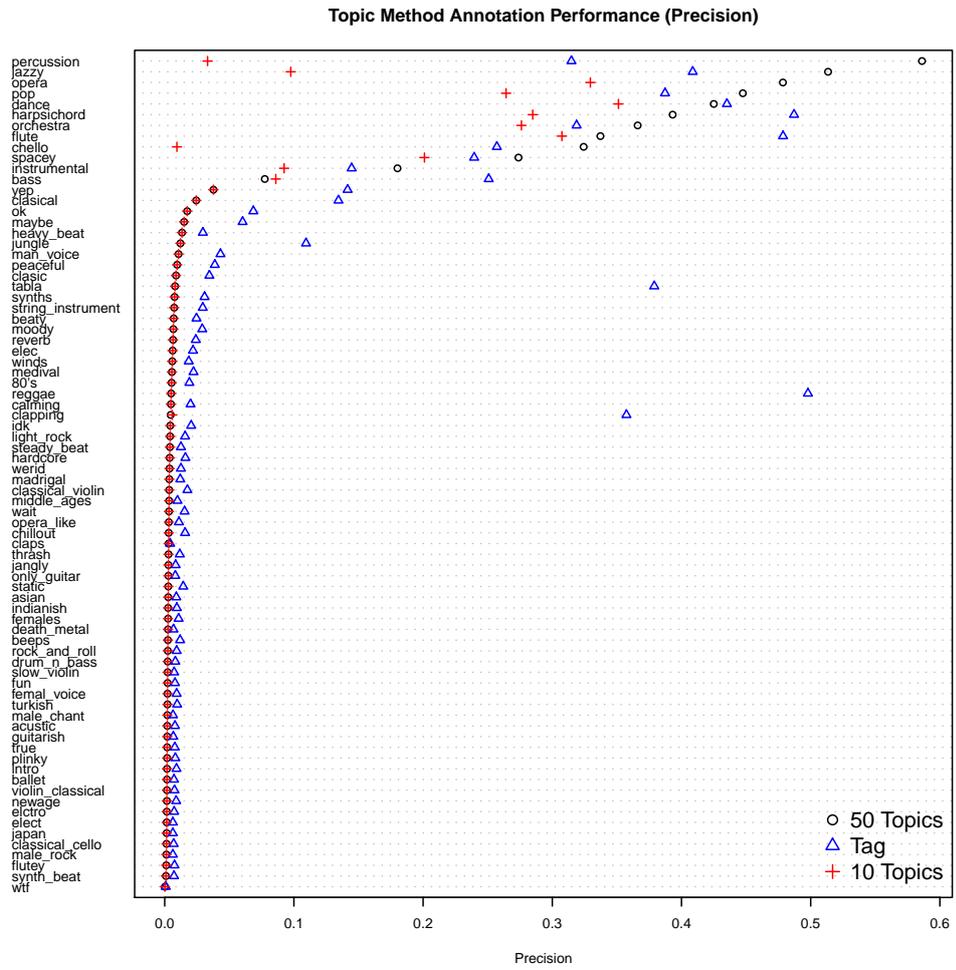


Fig. 12. Precision

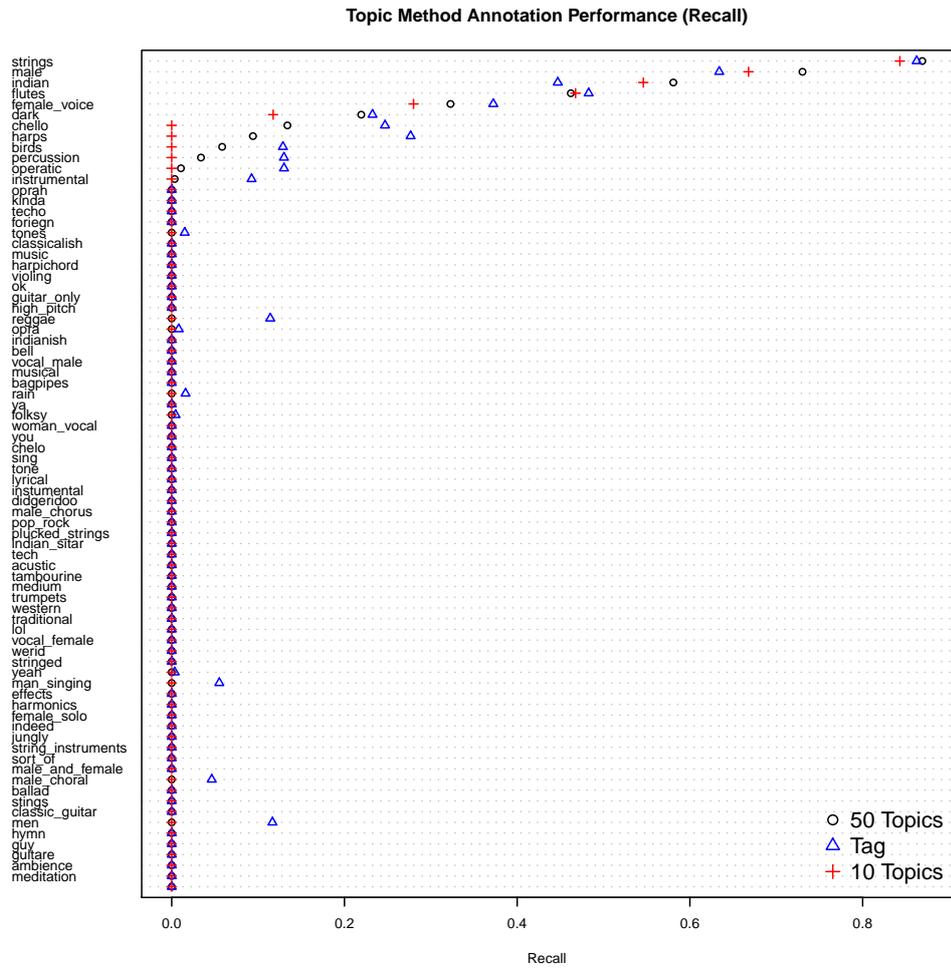


Fig. 13. Recall

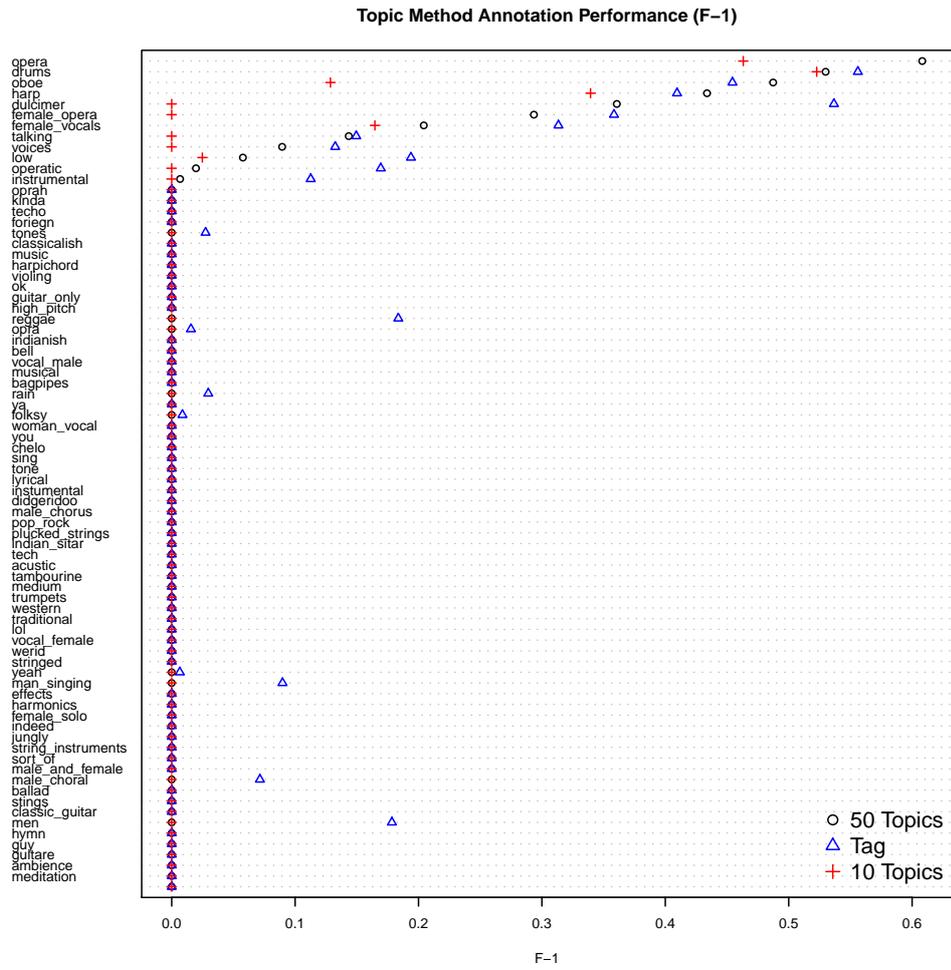


Fig. 14. F-1

Figure 12, 13 and 14 shows the detailed performance of the Topic Method (with 10 and 50 topics) and Tag Method for classifying individual tags, confirming our intuition about why the Tag Method is performing better than the Topic Method under the omission-penalizing metrics. Results shows that the Tag Method omits much fewer tags than the Topic Method, therefore, gaining precision scores for rarer tags (such as “meditation”, “male_and_female” etc), for which the Topic Method receives zeros. The plots also show that the Topic Method and Tag Method are very similar in their precision, recall and F-1 performance for more common tags (e.g., “opera”, “drums”, “strings” etc), and that the model with more topics (i.e. 50) generally outperforms that with fewer topics (i.e. 10) on the same tags.

5.4 Experiment 3: Retrieval Performance

The tags generated by a music tagger can be used to facilitate retrieval. Given a search query, music clips can be ranked order by the KL divergence between the query tag distribution and the tag probability distribution for each clip. We measure retrieval performance using the mean average precision (MAP) [29] metric, which computes precision (the number of retrieved music clips whose ground truth tags include the search query) while placing more weight on the higher ranked clips.

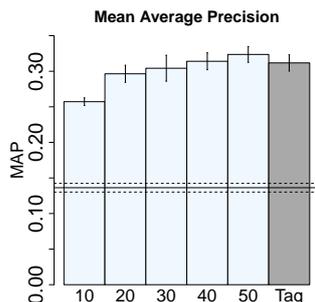


Fig. 15. Retrieval Performance, in terms of average mean precision

Figure 15 shows the retrieval performance of the three methods under this metric. The retrieval performance of the Topic Method (with 50 topics) is indistinguishable from the Tag method, and both methods significantly outperform the random baseline.

5.5 Experiment 4: Efficiency

One of the main motivation behind using the Topic Method to generate tags is efficiency, i.e., it is much faster to train a classifier to predict 50 topic classes than 834 tag classes.

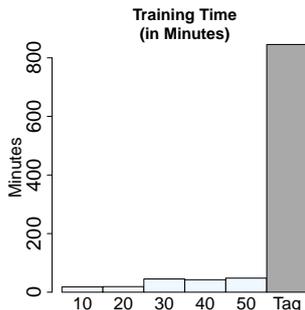


Fig. 16. Comparison of efficiency in terms of training time

Figure 16 shows a rough estimate of the training time (averaged over folds) of the different models. While the training time does increase as the number of topics increases, the training time plateaus and are very similar for topic models with 30, 40 or 50 topics. The most important observation is that the Topic Method is approximately 94% times faster to train than the Tag Method, which confirms our belief that our proposed method will be significantly more scalable as the size of the tag vocabulary grows.

Experiment 5: Evaluation

The performance metrics we have used so far can only *approximate* the quality of the generated tags. The reason is that the ground truth tags collected by TagATune, which we are using as if they were gold standards, can never be fully complete. When deciding if a generated tag is accurate, comparison against the ground truth set will systematically under-estimate performance, due to missing tags or vocabulary mismatch.

Consider the examples in Table 3. Table 3(a) shows examples of music clips that have only one or two ground truth tags (in **bold**). In this case, generated tags that cannot be found amongst ground truth tags are counted as wrong, when in fact they are correct. For example, the tags “india”, “oriental”, “middle eastern” (example i), or “guitar”, “loud”, “drums” (example ii), or “vocal”, “chorus”, “chant” (example iii) are not considered correct tags, even though they are either equivalent in meaning to the ground truth tags, or highly correlated and likely to be correct (e.g., in the case of “drums” and “rock”). Figure 3(b) show examples where the ground truth set tags *do* provide sufficient coverage, but because of vocabulary mismatch, there are again many false negatives. Examples of vocabulary mismatch include “electro” versus “electronic”, “beats” versus “beat” (example i), or “female voice” versus “female vocals”, “pop” versus “popish” (example ii), or “celtic” versus “irish”, “medival” versus “medieval”, or “strings” versus “string” (example iii).

(a) Missing Ground Truth Annotation

(i) sitar eastern TP: indian sitar eastern guitar oriental strings middle_eastern slow drums arabic TG: indian sitar guitar eastern slow drums oriental india strings solo
(ii) rock TP: rock guitar male male_vocal pop loud man vocal metal drums TG: rock male male_vocals male_vocal guitar male_voice pop loud man vocals
(iii) singing TP: choir choral opera vocal vocals chorus chant singing female classical TG: choir choral vocal opera singing chorus vocals female voices woman

(b) Vocabulary Mismatch

(i) faster jazzy beat fast disco guitar dance pop cymbals drums rock 80s upbeat electro TP: electronic drums synth rock beat fast guitar dance electro beats TG: electronic beat drums synth electro fast electric beats guitar rock
(ii) woman popish female_voice pop female vocal female_singer synth TP: female woman vocal female_vocal female_voice singing pop vocals female_vocals voice TG: female woman pop female_vocal singing vocals female_voice vocal guitar female_vocals
(iii) celtic classic violins violin medieval strings TP: classical violin guitar strings slow irish harp classic violins country TG: classical strings violin classic guitar fiddle violins string baroque medieval

Table 3. In **bold** are the ground truth tags. TP and TG refers to the Topic Method and Tag Method respectively.

Annotation Performance

In order to compare the true merit of the competing approaches, we conducted a Mechanical Turk experiment where we ask humans to evaluate the tags generated by the Topic Method (with 50 topics), Tag Method and Random Method. We randomly selected a set of 100 music clips (20 in each fold) and solicited evaluations from 10 unique turkers for each music clip. For each clip, the turker is given three lists of tags, generated by the Topic Method, Tag Method, and the Random Method respectively. The order of the three lists are randomized to eliminate any presentation bias. The turkers are asked to (1) click the checkbox beside a tag if that tag is appropriate for the music clip (i.e. describes the music well), and (2) rank order the three lists based on how well they describe the music clip overall. Figure 17 shows the interface for the Mechanical Turk annotation experiment.

Figure 18 shows the per-tag precision, recall and F-1 scores as well as the per-clip precision scores of the three methods, when we evaluate tags by comparing to the ground truth set versus using human evaluation. Results show that when tags are judged based on whether they are found amongst the ground truth tags, the performance of the tagger is grossly underestimated under any metrics. In fact, of the tags (generated by either Topic Method or Tag Method) that the turkers considered as “appropriate” for any music clip, on average, approximately 50% of them are not found in the ground truth sets.

While the performance of Topic Method and Tag Method are similar in this experiment, when asked which *list* of tags the human user prefers the most, second most and the least, the average numbers of votes (out of 10) are 6.20 for

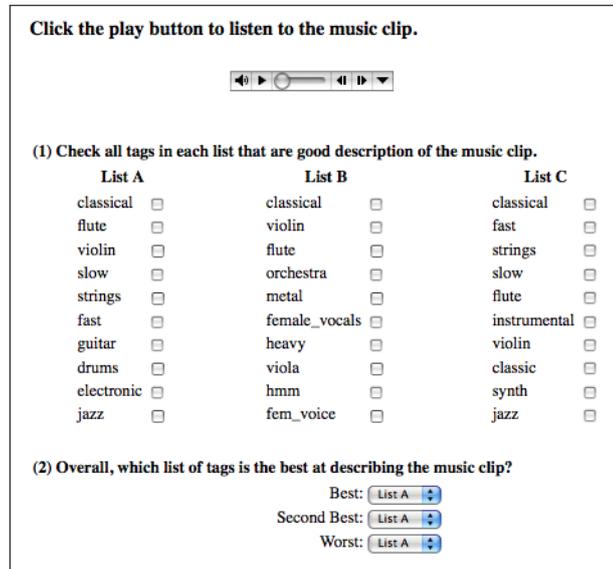


Fig. 17. Mechanical Turk Annotation Experiment: Interface

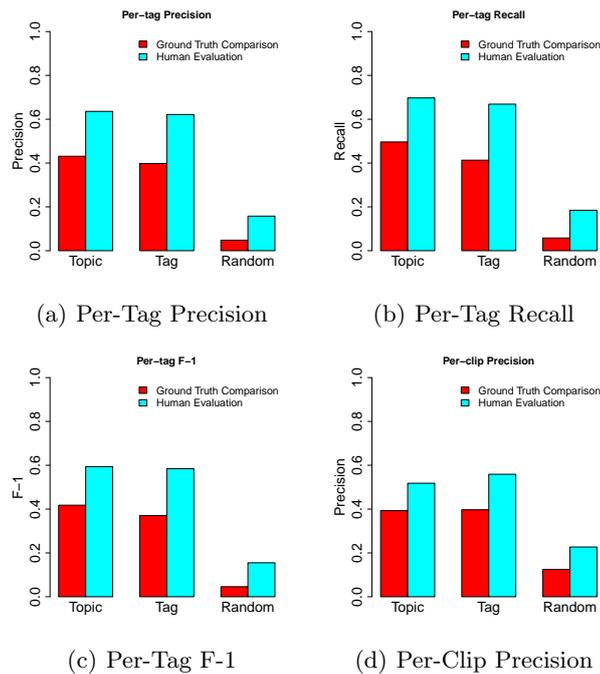


Fig. 18. Mechanical Turk Annotation Experiment: Results

the Tag Method, 3.34 for the Topic Method, and 0.46 for the Random Method, in strong favor of the Tag Method. Our hypothesis is that people actually prefer the Tag Method because its tag coverage is better than the Topic Method. This observation has interesting implications for how tags should be evaluated, individually versus as a whole.

Retrieval Performance

We conducted a similar experiment for evaluating retrieval performance. Figure 19 shows the interface for the Mechanical Turk annotation experiment.

Your search query is "man"

(1) Listen to each music clip in each list, and check those that are relevant to the search query.

List A	List B	List C
▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢
▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢
▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢
▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢
▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢
▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢
▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢
▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢
▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢
▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢	▶ ◀ ◁ ▷ ▢

(2) Which list has the most relevant songs for the search query?

Best:

Second Best:

Worst:

Fig. 19. Mechanical Turk Retrieval Experiment: Interface

Similar to the annotation task, our hypothesis is that the retrieval performance of the three methods, under the mean average precision metric, is underestimated because many of the retrieved music clips are false negatives if the search query cannot be found amongst their ground truth tags. To test this hypothesis, we ran a similar Mechanical Turk experiment, where we provide each turker a search query and three lists of music clips retrieved for that search query by the Tag Method, Topic Method and Random Method. Again, the order of the lists is randomized to prevent presentation bias. There are 100 one-word queries in total, and 3 users for evaluating the music clips retrieved for each query. Users are asked to check the checkbox of each music clip that they consider “relevant”

for the query. In addition, they are asked to rank order the three lists in terms of their overall relevance to the query.

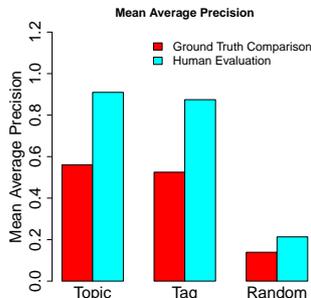


Fig. 20. Mechanical Turk Retrieval Experiment: Results

Figure 20 shows the mean average precision, when the ground truth tags versus human judgment is used to evaluate the relevance of each music clip in the retrieved set. Results show that when humans evaluate the retrieved list, the mean average precision of all methods are significantly higher than if we use the ground truth tags as the judge.

Finally, when asked which list of music clips the turker prefers the most, second most and the least, the average numbers of votes (out of 3) are 1.17 for the Tag Method, 1.78 for the Topic Method, and 0.56 for Random Method, in favor for the Topic Method.

6 Conclusion and Future Work

The purpose of this work is to show how classification algorithms can be trained, in an efficient way, to recognize the characteristics of some objects (e.g., music) when the training data consists of a huge number of noisy labels. Focusing on music tagging as the domain of interest, we showed that topic models can be used to define a reduced set of labels, using which the task of mapping from audio features to tags is made more efficient. Our proposed method opens up the opportunity to leverage the large number of tags freely available on the Web for training classification algorithms.

We compared the Topic Method and Tag Method on five criteria: feasibility, annotation performance, retrieval performance, computational efficiency, and annotation and retrieval performance as judged by human evaluators. Our main results show that our proposed method is feasible and both data-efficient (i.e., can utilize an arbitrary open vocabulary of tags) and time-efficient (i.e., reduces training time by 94% compared to learning from tag labels directly), achieves comparable performance for annotation and superior performance for retrieval.

An interesting finding is that despite the comparable annotation performance, human evaluators preferred the Tag Method, which we believe can be attributed to its superior tag coverage.

Future Work

The training phase of our method is currently a two-step process: we first learn a topic model over tags, then learn a mapping from audio features to topic distributions. In the future, we may investigate a training procedure that combines the two steps into one. For example, there has been recent work on topic models that are learned from not only text, but other metadata associated with the documents, such as sLDA [5] and DMR [31]. Another class of methods to investigate are semi-supervised techniques for performing factorization and classification simultaneously, such as Support Vector Decomposition Machine (SVDm) [35] or Collective Matrix Factorization [37].

Our work exposes the problem of evaluating tags when the ground truth sets are noisy or incomplete. Following the lines of [23], an interesting direction would be to build a human computation game that is suited specifically for evaluating tags, and which can become a service for evaluating any music taggers.

Finally, an exciting application area for this work is birdsong classification. To date, there are not many, if any, databases that would allow a birdsong to be retrieved by text, e.g., using arbitrary tags such as “high-pitched”, “cheep cheep chirp”, “black-throated sparrow”. Given the *vast* number of descriptions for any particular birdsong, it would be difficult to train a classification algorithm to map from audio features to tags directly, as most tags may be associated with only one or two birdsongs as examples. In collaboration with Cornell’s Lab of Ornithology, our plan is to use TagATune to collect tags for birdsongs from the tens of thousands of citizen scientists, and attempt to re-apply the technique here to train a classifier for automatically tagging birdsongs, so that they can be retrieved easily by semantic queries.

References

1. A. Berger, S. Della Pietra, and V. Della Pietra. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71, 1996.
2. J. Bergstra, A. Lacoste, and D. Eck. Predicting genre labels for artists using freedb. In *ISMIR*, pages 85–88, 2006.
3. T. Bertin-Mahieux, D. Eck, F. Maillet, and P. Lamere. Autotagger: a model for predicting social tags from acoustic features on large music databases. *TASLP*, 37(2):115–135, 2008.
4. D. Blei and J. Lafferty. Topic models. In A. Srivastava and M. Sahami, editors, *Text Mining: Theory and Applications*. Taylor and Francis, 2009.
5. D. Blei and J.D. McAuliffe. Supervised topic models. In *NIPS*, 2007.
6. D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.
7. C. Brodley and M. Friedl. Identifying mislabeled training data. *JAIR*, pages 131–167, 1999.

8. L. Chen, P. Wright, and W. Nejdl. Improving music genre classification using collaborative tagging data. In *WSDM*, pages 84–93, 2009.
9. I. Csizsar. Maxent, mathematics, and information theory. In K. Hanson and R. Silver, editors, *Maximum Entropy and Bayesian Methods*. Kluwer Academic Publishers, 1996.
10. R. B. Dannenberg and N. Hu. Understanding search performance in query-by-humming systems. In *ISMIR*, pages 41–50, 2004.
11. G. Eisenberg, J.M. Batke, and T. Sikora. Beatbank – an mpeg-7 compliant query by tapping system. In *Audio Engineering Society Convention*, page 6136, 2004.
12. E. Erosheva, S. Fienberg, and J. Lafferty. Mixed-membership models of scientific publications. *Proceedings of the National Academy of Sciences of the United States of America*, 101(Suppl 1):5220–5227, April 2004.
13. M. Goto and K. Hirata. Recent studies on music information processing. *Acoustic Science and Technology*, pages 419–425, 2004.
14. P. Herrera, G. Peeters, and S. Dubnov. Automatic classification of music instrument sounds. *Journal of New Music Research*, pages 3–21, 2003.
15. M. Hoffman, D. Blei, and P. Cook. Easy as cba: A simple probabilistic model for tagging music. In *ISMIR*, pages 369–374, 2009.
16. T. Hofmann. Probabilistic latent semantic analysis. In *UAI*, pages 50–57, 1999.
17. D. Hu and L. Saul. A probabilistic topic model for unsupervised learning of musical key-profiles. In *ISMIR*, pages 441–446, 2009.
18. T. Iwata, T. Yamada, and N. Ueda. Modeling social annotation data with content relevance using a topic model. In *NIPS*, 2009.
19. A. Kapur, M. Benning, and G. Tzanetakis. Query by beatboxing. In *ISMIR*, pages 170–178, 2004.
20. P. Lamere. Social tagging and music information retrieval. *Journal of New Music Research*, 37(2):101–114, 2008.
21. C. Laurier, M. Sordo, J. Serra, and P. Herrera. Music mood representations from social tags. In *ISMIR*, pages 381–386, 2009.
22. E. Law and L. von Ahn. Input-agreement: a new mechanism for collecting data using human computation games. In *CHI*, pages 1197–1206, 2009.
23. E. Law, K. West, M. Mandel, M. Bay, and S. Downie. Evaluation of algorithms using games: The case of music tagging. In *ISMIR*, pages 387–392, 2009.
24. M. Levy and M. Sandler. A semantic space for music derived from social tags. In *ISMIR*, 2007.
25. T. Li, M. Ogihara, and Q. Li. A comparative study on content-based music genre classification. In *SIGIR*, pages 282–289, 2003.
26. M. Mandel and D. Ellis. Song-level features and support vector machines for music classification. In *ISMIR*, 2005.
27. M. Mandel and D. Ellis. Labrosa’s audio classification submissions. www.music-ir.org/mirex/2008/abs/AA_AG_AT_MM_CC_mandel.pdf, 2009.
28. M. Mandel and D. Ellis. A web-based game for collecting music metadata. *Journal of New Music Research*, 37(2):151–165, 2009.
29. Performance metrics for information retrieval. http://en.wikipedia.org/wiki/Information_retrieval.
30. Andrew Kachites McCallum. Mallet: A machine learning for language toolkit. <http://mallet.cs.umass.edu>, 2002.
31. D. Mimno and A. McCallum. Topic models conditioned on arbitrary features with dirichlet-multinomial regression. In *UAI*, 2008.

32. K. Nigam, J. Lafferty, and A. McCallum. Using maximum entropy for text classification. In *IJCAI-99 Workshop on Machine Learning for Information Filtering*, pages 61–67, 1999.
33. J. Nocedal. Updating quasi-newton matrices with limited storage. *Mathematics of Computation*, pages 339–353, 1995.
34. M. Palatucci, D. Pomerleau, G. Hinton, and T. Mitchell. Zero-shot learning with semantic output codes. In *NIPS*, 2009.
35. F. Pereira and G. Gordon. The support vector decomposition machine. In *Proceedings of the 23rd International Conference on Machine Learning (ICML)*, pages 689–696, 2006. <http://portal.acm.org/citation.cfm?id=1143844.1143931>.
36. U. Rebbapragada and C. Brodley. Class noise mitigation through instance weighting. In *ECML*, pages 708–715, 2007.
37. A. Singh and G. Gordon. Relational learning via collective matrix factorization. In *KDD*, pages 650–658, 2008.
38. M. Steyvers and T. Griffiths. Probabilistic topic models. In T. Landauer, D.S. McNamara, S. Dennis, and W. Kintsch, editors, *Handbook of Latent Semantic Analysis*. Erlbaum, Hillsdale, NJ, 2007.
39. K. Trohidis, G. Tsoumakas, G. Kalliris, and I. Vlahavas. Multi-label classification of music emotions. In *ISMIR*, pages 325–330, 2008.
40. D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet. Towards musical query-by-semantic description using the CAL500 data set. *SIGIR*, pages 439–446, 2007.
41. D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet. Semantic annotation and retrieval of music and sound effects. *TASLP*, 16(2):467–476, February 2008.
42. D. Turnbull, R. Liu, L. Barrington, and G. Lanckriet. A game-based approach for collecting semantic annotations of music. In *ISMIR*, pages 535–538, 2007.
43. G. Tzanetakis and P. Cook. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302, 2002.
44. G. Tzanetakis, G. Essl, and P. Cook. Automatic music genre classification of audio signals. In *ISMIR*, 2001.
45. L. von Ahn and L. Dabbish. Labeling images with a computer game. In *CHI*, pages 319–326, 2004.
46. B. Whitman and P. Smaragdis. Combining musical and cultural features for intelligent style detection. In *ISMIR*, 2002.
47. L. Yao, D. Mimno, and A. McCallum. Efficient methods for topic model inference on streaming document collections. In *KDD*, pages 937–946, 2009.
48. X. Zhu, X. Wu, and S. Chen. Eliminating class noise in large datasets. In *ICML*, pages 920–927, 2003.