Topic Modeling for Conference Analytics

Pengfei Liu¹, Shoai Bameel¹, Wai Lam¹, Bin Ma², Helen Meng¹

¹Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Hong Kong SAR, China
²Human Language Technology Department, Institute for Infocomm Research, Singapore

{pfliu, msjameel, wlarm, hmmeng}@se.cuhk.edu.hk, mabin@i2r.a-star.edu.sg

Abstract

This work presents our attempt to understand the research topics that characterize the papers submitted to a conference, by using topic modeling and data visualization techniques. We infer the latent topics from the abstracts of all the papers submitted to Interspeech2014 by means of Latent Dirichlet Allocation. Per-topic word distributions thus obtained are visualized through word clouds. We also compare the automatically inferred topics against the expert-defined topics (also known as tracks for Interspeech2014). The comparison is based on an information retrieval framework, where we use each latent topic as a query and each track as a document. For each latent topic, we retrieve a ranked list of tracks scored by the degree of word overlap. Each latent topic is associated with the top-scoring track. This analytic procedure was applied to all submissions to Interspeech2014 and sheds some interesting light in terms of providing an overview of topic categorization in the conference, popular versus unpopular topics, emerging topics and topic compositions. Such insights are potentially valuable for understanding the technical content of a field and planning the future development of its conference(s).

Index Terms: topic modeling, conference analytics, information retrieval

1. Introduction

Academic conferences such as INTERSPEECH usually call for papers with a list of expert-designed research tracks and sub-tracks. These tracks are useful for the organization of a technical conference where authors can submit their papers to their preferred tracks, reviewers can choose papers to review from their preferred tracks, and readers can search for papers by filtering tracks. Conference organizers may be interested in questions such as, Are the list of tracks is representative and diverse enough to cover all major research topics of the field? Are the tracks described well? Do the tracks match the research topics in the submitted papers as they vary from year to year? Are authors submitting their papers to the relevant tracks? If not, then are the track names ambiguous? Alternatively, are the track descriptions confusing?

To answer these questions properly, we present a new task named topic-track matching, which matches latent topics inferred from the content (e.g., abstract) of submitted papers with a list of expert-designed conference tracks. The task can be helpful to the conference organizers in the following ways: (1) Help the organizers monitor whether authors are submitting their papers to the appropriate track; (2) Help the organizers monitor whether a track is ambiguously described; (3) Analyze the popularity of the tracks and merge less popular tracks with other related tracks for future conferences; (4) Help the organizers observe the popularity of topics in their field, which may engender new tracks for emerging and popular topics.

We propose to apply topic modeling techniques for the topic-track matching task. Probabilistic topic models [1, 2] such as Latent Dirichlet Allocation (LDA) [3] are statistical models that find patterns of words or underlying latent topics from a large collection of documents, which have been widely used in the past to study academic conferences [4, 5, 2]. We investigate the task under an information retrieval framework [6], with each latent topic as a query and each track as a document for retrieval. We apply the LDA model on abstracts of all the submitted papers from the conference organizers of Interspeech2014 to infer K latent topics and match each topic with the most similar track, which has the highest F-score calculated by counting the overlapping words between top-ranking words based on decreasing probability of a topic and keywords of a track. The outputs of the task are matches between topics and tracks, word cloud visualization of topics and distribution analysis of matching results to give suggestions for future conference organization. Our experimental results show that the topic-track matching task enables the detection of mismatches between inferred latent topics and expert-designed tracks. This can help suggest which track descriptions may be revised. Our results can also facilitate the analysis of the popularity of tracks for better future conference organization.

2. Approach

2.1. Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) is a generative probabilistic topic model for collections of discrete data such as text corpora. Following the notations in [3], given the hyper-parameters α and β, LDA defines the probability of a corpus D with M documents, as illustrated in formula (1), where θd is a topic mixture for document d, wdn is the nth word from document d, and zd,n is the latent topic assignment for the word wdn given θd.

\[ p(D|α, β) = \prod_{d=1}^{M} \prod_{z_{d,n}} p(θ_d|α) \left( \frac{1}{N} \sum_{n=1}^{N} p(z_{d,n}|θ_d)p(w_{d,n}|z_{d,n}, β) \right) dθ_d \]  

(1)

In the generative process of LDA, each document is generated by first sampling a document-specific topic proportion θd
from a Dirichlet distribution, and then drawing each word from a topic-specific Multinomial distribution \( p(w_{dn}|z_{dn}, \beta) \). The model generates a low-dimensional representation of data, consisting of a word distribution of \( P(w|z) \), which states the probability of a word \( w \) belonging to a topic \( z \) and a topic distribution in a document \( P(z|d) \), which specifies the mixture of topics in a document \( d \). Our interest is on \( P(w|z) \) as we will match the words in topics with the words in expert-designed tracks of an academic conference. The LDA model can be estimated by several algorithms such as the variational Bayes algorithm by Blei et al. [3], the expectation propagation algorithm by Minka et al. [7] and the collapsed Gibbs sampling algorithm by Griffiths and Steyvers in [4] and so on.

### 2.2. Topic-Track Matching

Conceptually, we assume a 1:N relationship between track and topic, and we define a match as a pair between a topic and its top-one similar track based on F-score. We tackle the topic-track matching problem under an information retrieval framework, with each latent topic as a query and each track as a document. The latent topics are obtained by applying LDA on the paper abstracts and we represent each topic by choosing its top \( N = 200 \) words based on the decreasing probability of each word. This parameter is set empirically to cover approximately 80% of the probability space of the words in each topic. Each document typically consists of 20-50 words pre-processed from the corresponding track description.

Our topic-track matching system is illustrated in Figure 1. We first applied the same pre-processing step to both Conference Tracks and Paper Abstracts. Then, we applied LDA to get the latent topics with a list of top words in descending order of probability \( P(w|z) \), which are queries for retrieving the tracks represented with a set of key words.

For each query (topic), we match it with the document (track) which has the highest F-score obtained by calculating their overlapping words. As the F-score measure used in Text::Similarity\(^3\) for pair-wise similarity of files or strings, we calculated F-score by first counting the number of matching words between the key words of a track \( (W_t) \) and the top words \( (W_i) \) of a topic, and then computing Precision, Recall and F-score accordingly, as shown in formulas (2), (3) and (4).

\[ \text{Precision} = \frac{|W_k \cap W_t|}{|W_t|} \quad (2) \]
\[ \text{Recall} = \frac{|W_k \cap W_t|}{|W_i|} \quad (3) \]
\[ \text{F-score} = \frac{|W_k \cap W_t|}{\frac{|W_k|+|W_t|}{2}} \quad (4) \]

### 2.3. System Implementation

We developed our matching system in Java which includes dataset pre-processing, F-score calculation and topic-track matching, as well as a LDA component from the MALLET [8] toolkit which implements the collapsed Gibbs sampling algorithm for model inference. Specifically, we used the Java class ParallelTopicModel in MALLET, a parallel threaded implementation of LDA, whose detailed algorithm is described in [9, 10].

### 3. Experiments

#### 3.1. Corpus

We conducted experiments on the paper abstracts of 12 main tracks set by the conference organizers of Interspeech2014\(^4\). In the dataset, the total number of paper abstracts is 1,078, and the total number of words is 101,312. The average number of words in a paper abstract is about 94. After stopword removal and stemming, the vocabulary contains 5,732 unique words.

#### 3.2. Experimental Settings

**Preprocessing:** For the step of pre-processing, we kept only content words and stemmed each word by morphology (i.e., computing the base form of English words by removing inflections such as noun plurals, pronoun case and verb endings). We lowercased all words and removed stop words.

**Hyperparameters of \( \alpha \) and \( \beta \):** We then applied the LDA model on the pre-processed dataset, with the hyper-parameters optimized using the hyper-parameter sampling algorithm implemented in MALLET.

**Number of Topics:** We empirically set the number of latent topics as 20, which is higher than the number of tracks (12) because there may exist some topics not described in tracks by conference organizers but included in a number of papers. This number is close to 22 given by the hierarchical Dirichlet Process (HDP) model [11, 12], which takes a nonparametric Bayesian approach to find the number of latent topics automatically [11, 13, 2]. We also verified the number of topics by showing the average matching F-score under different number of topics for Interspeech2014, in Figure 2a, which shows that the number of topics should not be too high to lead to a low average F-score.

**Number of Iterations:** We tune the number of iterations by comparing the training perplexity [3] of the LDA model on the whole dataset under different number of iterations, as shown in Figure 2b. We can see that 10,000 iterations is enough to lead the perplexity to be stable. Therefore, we conducted all the experiments with the number of iterations as 10,000, which is also feasible for a relatively small dataset like Interspeech2014.


\(^4\)We also tried to apply LDA on the tracks directly to infer their top covered topics, which are however not distinguishable among tracks.

\( \text{Precision} = \frac{|W_k \cap W_t|}{|W_t|} \quad (2) \)
\( \text{Recall} = \frac{|W_k \cap W_t|}{|W_i|} \quad (3) \)
\( \text{F-score} = \frac{|W_k \cap W_t|}{\frac{|W_k|+|W_t|}{2}} \quad (4) \)

---

**Figure 1:** The Topic-Track Matching System.
3.3. Topical Words

In Table 1, we analyzed the LDA results by presenting the top 10 words sorted by decreasing probability for 8 topics out of 20 topics inferred from the dataset. We chose the top 10 words because these words can provide sufficient detail to convey the subject of a topic, and distinguish one topic from another [14]. We also named each topic with a title manually in the table header which helps us gain an overview of the research topics in Interspeech2014, such as Speech Recognition, Speech Synthesis, Language Modeling, etc.

3.4. Visualization of Topic-Track Matching

We present two illustrations of topic-track matching (Again, topics are automatically derived and tracks are expert defined.). The first example is between the topic of Neural Network (See Table 1) and Track 7 (See Table 2), to which there are 177 submitted papers. Figure 3 shows a word cloud of the topic Neural Network by the tool Wordle\(^4\) and the matching words with Track 7, where words with larger font have higher probability in the topic while the layout and color are randomly set for visualization. We observe in Figure 3 that deep neural network has become a popular topic for Track 7 in speech recognition.

The other matching example is between the topic Speech Synthesis (See Table 1) and Track 6 (See Table 2), which has 105 submitted papers. Figure 4 shows a word cloud of the topic and its matching words with Track 6, which presents the areas of speech synthesis and voice conversion and the HMM statistical model.

3.5. Analysis of Matches between Latent Topics and Tracks

Figure 5(a) shows the paper submissions across the 12 Interspeech2014 tracks, as selected by the authors. Figure 5(b) shows the number of latent topics matched to each track accord-

---

\(^4\)http://www.wordle.net/
Table 2: Descriptions of Track 6 and Track 7 from Interspeech2014.

<table>
<thead>
<tr>
<th>Track 6</th>
<th>Track 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1 Grapheme-to-phoneme conversion for synthesis</td>
<td>7.1 Feature extraction and low-level feature modeling for ASR</td>
</tr>
<tr>
<td>6.2 Text processing for speech synthesis (text normalization, syntactic and semantic analysis)</td>
<td>7.2 Prosodic features and models</td>
</tr>
<tr>
<td>6.3 Segmental-level and/or concatenative synthesis</td>
<td>7.3 Robustness against noise, reverberation</td>
</tr>
<tr>
<td>6.4 Signal processing/statistical model for synthesis</td>
<td>7.4 Far field and microphone array speech recognition</td>
</tr>
<tr>
<td>6.5 Speech synthesis paradigms and methods, silence speech, articulatory synthesis, parametric synthesis etc.</td>
<td>7.5 Speaker normalization (e.g., VTLN)</td>
</tr>
<tr>
<td>6.6 Prosody modeling and generation</td>
<td>7.6 Deep neural network</td>
</tr>
</tbody>
</table>

Figure 5: Distribution of Submitted Papers, and Matches between Latent Topics and Tracks in Interspeech2014.

In this paper, we propose a new task of topic-track matching which applied LDA on paper abstracts and matched inferred latent topics with expert-designed conference tracks. We investigated the task under an information retrieval framework, with each topic as a query and each track as a document. For each topic, we retrieved a ranked list of tracks by calculating their word overlapping score and chose the top-scoring track as the matching document. Finally, we analyzed the matches to obtain trends which may be helpful for future conference organization.

4. Conclusions and Future Work

In this work, we were interested in hierarchical topic structures from conference papers, which not only help researchers know current research topics but also facilitate future conference organizers to derive better tracks and sub-tracks.

5. Acknowledgements

This work is affiliated with the Stanley Ho Big Data Decision Analytics Research Center of The Chinese University of Hong Kong.
6. References


