A Model of Extended Paragraph Vector for Document Categorization and Trend Analysis

Pengfei Liu
Department of Systems Engineering and Engineering Management
The Chinese University of Hong Kong
Email: pfliu@se.cuhk.edu.hk

King Keung Wu
Department of Mechanical and Automation Engineering
The Chinese University of Hong Kong
Email: kkwu@mae.cuhk.edu.hk

Helen Meng
Department of Systems Engineering and Engineering Management
The Chinese University of Hong Kong
Email: hmmeng@se.cuhk.edu.hk

***Abstract*—The increasing number of academic papers published each year has led to a growing demand of organizing the papers into different categories according to their topics and analyzing topic trends over time. Domain knowledge such as journal categories and conference sessions are potentially useful for categorizing papers and obtaining trends easily interpretable to users. In this paper, we aim to organize a collection of papers into journal categories which describe research areas of a field, and then analyze the trend of each research area. Conference sessions are used to link with journal categories assuming that papers from the same session are put into the same category. Sessions are also adopted to reflect the trend of a category over years as they are derived by domain experts to describe each year's topics. First, we present a model of *extended paragraph vector* to model the hierarchical structure of sessions, papers and words, and capture their semantics with distributed vector representations in the same space. Then, we propose a *two-stage approach* for document categorization, which first chooses a subset of journal categories covering the major research areas in the corpus and then associates each session with its most similar category based on session vectors. Finally, we present the research trend of a category through its matching sessions ordered in time and showing the most similar words of each session.**

I. Introduction

With more and more academic papers available each year, there is a growing demand to categorize these papers according to their topics and analyze the temporal trends covered in these papers. Probabilistic topic models have been widely applied to discover topics in papers and analyze their trends. For example, Griffiths and Steyvers [1] adopted the latent Dirichlet allocation (LDA) model to find scientific topics in a corpus of paper abstracts; Hall et al. [2] studied the development of ideas in a scientific field by post-processing LDA results; Blei et al. [3] analyzed the dynamic evolution of topics in large document collections using a dynamic topic model. However, the obtained topics usually need to be labeled or interpreted manually by a domain expert and the labeling is subjective and time-consuming. Besides, the topics may not align very well with human-derived categories for a collection of documents [4]. In addition, the topics and their trends are not easily consumable to users without additional domain knowledge.

Domain knowledge like journal categories are potentially useful for categorizing papers according to their subjects and analyzing temporal trends from the perspective of expert-derived categories for easily interpretable, user-oriented results. For example, Table I shows some EDICS (Editors Information Classification Scheme) categories from the journal of *IEEE/ACM Transactions on Audio, Speech, and Language Processing* (TASLP). The EDICS categories describe the major research areas in the field of *Audio, Speech, and Language Processing*, and are typically stable in their titles, although with minor updates in category descriptions over time. These categories are potentially good candidates to organize an accumulating collection of papers over time.

TABLE I: Example EDICS categories from *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Category Title	Description
Speech Production	Physical models of the vocal production system; bioacoustics and medical acoustics; singing and properties of the musical voice.
Speech Perception and Psychoacoustics	Models of Speech Perception; hearing and psychoacoustics; physiological models and applications thereof; audiology applications.
Speech Analysis	Spectral and other time-frequency analysis techniques; segmental and suprasegmental analysis; distortion measures; extraction of non-linguistic information (e.g., gender, stress, etc); voice/speech disorders; speaker localization (space) (e.g., in meetings); speaker diarization (time) (e.g., in meetings); speaker clustering (e.g., in Broadcast news).
Speech Synthesis and Generation	Segmental-level and/or concatenative synthesis; signal processing/statistical model for synthesis; articulatory synthesis; parametric synthesis; prosody, emotional, and expressive synthesis; text-to-phoneme conversion; voice quality/morphing; audio/visual speech synthesis; multilingual synthesis; quality assessent/evaluation metrics in synthesis; tools and data for speech synthesis; text processing for speech synthesis (text normalization, syntactic and semantic analysis).

Similarly, there are also conference sessions which organize papers according to their topics. A session title consists of a few words or phrases to describe a specific topic, e.g., a session of *deep neural networks for speech synthesis* from Interspeech. As session titles are derived by domain experts in each year's conference, they may naturally reflect the trend of a topic (e.g., speech synthesis) over time, and provide an interesting perspective to analyze research trends.

We then raise such a motivation question: *How to assign the sessions together with their papers into the EDICS categories and investigate the temporal trends of each category?* This question essentially involves two tasks: *document categorization* and *trend analysis*, i.e., assigning papers into the EDICS categories and analyzing temporal trend of each category.

The first task of document categorization in our setting is very challenging due to a few reasons. *First*, we need to identify which EDICS categories are suitable for the Interspeech papers as there are some EDICS categories defined for other particular research areas such as audio or language processing.

978-1-5090-6182-2/17/$31.00 ©2017 IEEE

We may ask a speech expert to specify the categories, which however is subjective and time-consuming. Instead, we choose categories by *asking the data* to make our approach automatic and generalizable to other datasets as well. *Second*, an EDICS category is typically described with a few words and phrases, while a session contains a larger number of words from more than 10 papers on average. It might be biased to compare directly the words of a session and a category description. Instead, we represent a category with its most similar sessions and compare the similarity between a session and a category indirectly through these sessions. However, this leads to the *third* challenge of how to measure the semantic similarities between sessions. And the second task of trend analysis is highly dependent on the results of the first task.

In this paper, we aim to categorize a large collection of academic papers into a set of expert-defined journal categories based on their semantic similarities and analyze temporal trends of each category. Our contributions are three folds: (1) propose a model of *extended paragraph vector* to capture the semantics of sessions, papers and words in the same vector space; (2) present an iterative semantic-based two-stage approach for document categorization; (3) conduct trend analysis for each category based on its matching sessions and the most similar words of each session.

II. Related Work

Word Embeddings Word embeddings, also known as distributed representations of words in a vector space, have been successfully applied in various natural language processing tasks, such as neural network language modeling[5], part-of-speech tagging and named entity recognition[6], machine translation[7], sentiment analysis[8] and so on. Mikolov et al. [9] proposed the continuous bag-of-words (CBOW) model and the continuous skip-gram model for computing distributed vector representations of words from very large data sets. Moving beyond word-level representations, distributed representations for compositional semantics have also received a lot of attentions, such as representing the meaning of word combinations by vector composition in terms of additive and multiplicative functions by Mitchell et al. [10], the semantic compositionality through recursive matrix-vector spaces by Socher et al. [11], as well as the recursive neural tensor network for modeling the parse tree of a sentence by Socher et al. [12] and so on. The idea of learning a joint vector space has shown considerable success in several works. Weston et al. [13] learns a low-dimensional joint embedding space for both images and annotations. Srivastava et al. [14] proposes a deep Boltzmann machine to extract a representation of multimodal data from the joint space of image and text inputs. Recently, Le and Mikolov [15] proposed a model named *paragraph vector* to learn continuous distributed vector representations for texts of variable length, called *paragraph*.

Document Clustering Document clustering, aiming to organize similar documents into groups, has been a very important task for document organization, browsing, summarization, classification and retrieval [16]. Aggarwal et al. [17] presented a survey of document clustering algorithms, including probabilistic document clustering by topic models, distance-based clustering like agglomerative and hierarchical clustering, partition-based clustering like K-means, word and phrase based clustering (e.g., based on frequent word patterns), online clustering with text streams as well as semi-supervised clustering with labeled data to guide the clustering process. Remarkably, Lu et al. [4] showed empirically that using the most likely topic in topic models as the cluster is not as accurate as traditional clustering baseline such as K-means.

Trend Analysis Trend analysis from unstructured text is a challenging problem. There are keyword-based approaches [18], [19], [20], [21] and topic modeling based approaches [3], [22], [23], [24]. For example, Bollen et al. [21] performed a quantitative trend analysis of *D-Lib Magazine*'s text content from 1995 to 2004, by mining shifting patterns of how words co-occur in documents over time and using these patterns to pinpoint some trends in the community. Blei et al. [3] conducted trend analysis by a dynamic topic model, which captures the trajectory of the posterior probabilities of the words for each topic. Wang et al. [22] introduced a non-Markov continuous-time model named topics over time (TOT) for capturing topical trends. Wei et al. [23] proposed a dynamic mixture model (DMM), assuming that the document-specific topic mixture proportions are dependent on the mixture proportions of the previous timestamp following a Dirichlet distribution. Ahmed et al. [24] developed a dynamic hierarchical Dirichlet process model named infinite dynamic topic models (iDTM) to learn the number of latent topics from data together with capturing topic dynamics.

III. Methodology

In this section, we first describe the model of *extended paragraph vector* to capture the semantics of sessions, papers and words in the same vector space, and then present an iterative semantic-based two-stage approach for document categorization, and finally describe our methods to conduct category-specific trend analysis.

A. The Model of Extended Paragraph Vector

We propose a novel neural network architecture to learn distributed representations of sessions, papers and words in the same vector space. As shown in Figure 1, this architecture extends the model of *paragraph vector* [15] by adding an additional matrix $S \in \mathbb{R}^{L \times N}$ for L sessions, and $P \in \mathbb{R}^{M \times N}$ and $W \in \mathbb{R}^{V \times N}$ are the matrices for M papers and V unique words respectively. In the input layer, papers from the same session share the same row vector from S and words from the same paper share the same row vector from P. All the words share the same word matrix W. Each session vector provides the first-level context and each paper vector provides the second-level context. For a context window of C words from paper p in session s, the session vector S_s, the paper vector P_p and the $C-1$ word vectors are averaged in the projection layer to predict the target word in the output layer.

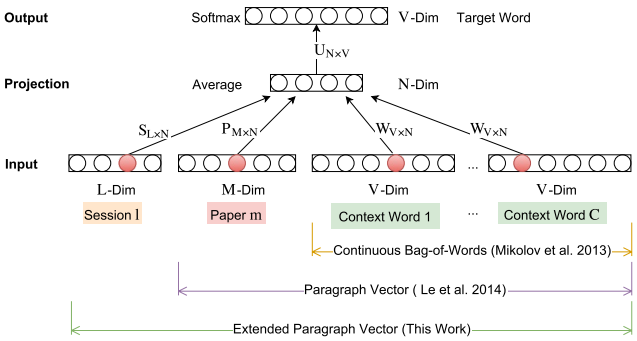

Fig. 1: The neural network for embedding sessions, papers and words in the same N-*dimensional* vector space.

Formally, given a sequence of words $w_1, w_2, \ldots, w_T$ from the whole corpus, the objective of the CBOW model [9] is to maximize the average log probability:

$$L = \frac{1}{T}\sum_{t=k+1}^{T-k} \log p(w_t | w_{t-k}, \ldots, w_{t+k}) = \frac{1}{T}\sum_{t=k+1}^{T-k} \log \frac{\exp(y_{w_t})}{\sum_i \exp(y_i)} \quad (1)$$

$$y = b + U^{\mathrm{T}} h\left(w_{t-k}, \ldots, w_{t+k}; W\right) = b + U^{\mathrm{T}} \sum_{i=t-k}^{t+k} W_i \quad (2)$$

where we assume a sliding window of $w_{t-k}, \ldots, w_{t+k}$, in which w_t is the target word and all other words are context words ($C = 2k$) and each word in the input layer is represented by a one-hot column vector x of the vocabulary size $V, x \in \mathbb{R}^V$ and $b \in \mathbb{R}^V$ is a bias. $W_i = W^{\mathrm{T}} x_i$ represents the ith row vector of the weight matrix W, and is called the *embedding* of the ith word in the vocabulary. To learn vector representations for text of variable length, the model of *paragraph vector* [15] introduces a new matrix D as defined in (3), where a row vector of D_d acts as a memory of the topic of the paragraph d.

$$y = b + U^{\mathrm{T}} h\left(w_{t-k}, \ldots, w_{t+k}, D_t; W, D\right) = b + U^{\mathrm{T}} \left(\sum_{i=t-k}^{t+k} W_i + D_d\right) \quad (3)$$

To model the three-level hierarchy among sessions, papers and words, we introduce an additional context matrix S for sessions together with the paper matrix P (similar to D in the paragraph vector). For a sliding window of $w_{t-k}, \ldots, w_{t+k}$ obtained from a paper p under a particular session s, we have the following formulation:

$$y = b + U^{\mathrm{T}} h\left(w_{t-k}, \ldots, w_{t+k}, p, s; W, P, S\right) = b + U^{\mathrm{T}} \left(\sum_{i=t-k}^{t+k} W_i + P_p + S_s\right) \quad (4)$$

Thus, we embed sessions, papers and words in the same vector space, which also has the advantage of supporting simple vector arithmetics among them (See Section IV-C3 and IV-D).

B. The Two-Stage Approach for Document Categorization

We propose an iterative semantic-based two-stage approach for document categorization. The two-stage approach consists of the first stage to choose a subset of the EDICS categories and their initial representative sessions from the corpus, and the second stage to assign each remaining session into its most similar category in an iterative and incremental procedure, as shown in Figure 2.

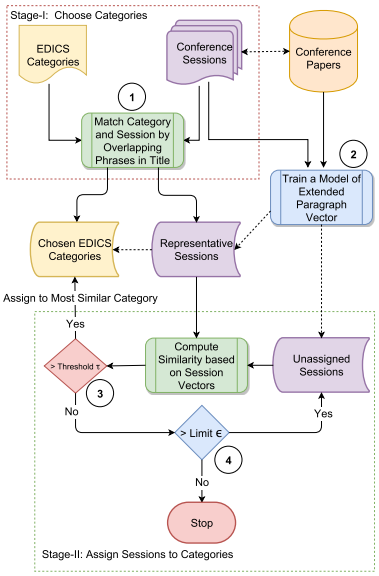

Fig. 2: The iterative semantic-based two-stage approach for document categorization.

In the first stage, we aim to choose a suitable subset from all the EDICS categories and meanwhile find out the representative sessions for each chosen category. We define that a category and a session matches if and only if the title of one is a sub-string of the title of the other. Such a strict matching criteria ensures that the category and the session refers to the same research area, and thereby the session is a good representative for the category. On the other hand, conference organizers may use different terms to define sessions, the exact matching with a category may indicate that the category (described by widely accepted terms) is well covered in the corpus. Therefore, we claim that the first stage enables us to choose the categories well-covered in the corpus and their corresponding representative sessions (①, See Section IV-C1 for the examples.).

In Stage-II, we aim to group each remaining session into its most similar category. The similarity is calculated based on the model of *extended paragraph vector* (②, See Section III-A). As the order of sessions affects the final categorization results, we assign the most similar sessions to their categories in the earliest iterations by introducing the similarity threshold τ

(③), which is decreased by a decay rate $\eta, \tau = \tau \times \eta$ after each iteration. Similar to K-means, we define a centroid for each category based on its already assigned sessions. This centroid is used to calculate the similarity between a session with a category. Different with K-means, we update the centroid for a category even in the same iteration as long as a new session has been assigned to the category, and Stage-II stops if the similarity is lower than the pre-specified similarity limit ϵ (④). This ensures all previously assigned sessions of a category can vote for any new candidate session, instead of keeping the centroids fixed in each iteration.

C. Category-Specific Trend Analysis

Based on the results of assigning conference sessions into journal categories, we sorted the matching sessions of each category in ascending order of time. The session titles naturally reflect the temporal trend of a category, e.g., the technology changes of *speech synthesis*, as explained in Section IV-D. We also obtained the most semantically similar words with each matching session of a category each year. The most similar word w is formally defined as $\text{argmax}_w \text{sim}(w, s)$ for a given session s, where $\text{sim} = \frac{w \cdot s}{\|w\| \|s\|}$ is the cosine similarity between the two vectors of w and s.

IV. Experiments

A. Corpus

We collected all the Interspeech papers between 2000 to 2015 from the ISCA (International Speech Communication Association) online archive[1], and parsed the HTML pages to obtain the title, abstract and the corresponding session of each paper. An example paper is shown in Table II. The basic corpus statistics are shown in Figure 3, where the total number of papers is 12,220 and the total number of sessions is 1,060.

TABLE II: An example paper in the Interspeech corpus.

Title	An Investigation of Recurrent Neural Network Architectures for Statistical Parametric Speech Synthesis
Abstract	In this paper, we investigate two different recurrent neural network (RNN) architectures: Elman RNN and recently proposed clockwork RNN for statistical parametric speech synthesis (SPSS). Of late, deep neural networks are being used for SPSS which involve predicting every frame independent of the previous predictions, and hence requires post-processing for ensuring smooth evolution of speech parameters. RNNs, on the other hand, are intuitively better suited for the task as they inherently model temporal dependencies, but were restricted in use because of the difficulty in training. Lately, techniques such as sparse initialization, Nesterov's accelerated gradient, gradient clipping and leaky integration (LI) have been shown to overcome this difficulty. We study the utility of these techniques for SPSS task. In addition, we show that clockwork RNN is equivalent to an Elman RNN with a particular form of LI. This perspective enables us to understand the reason why a simple Elman RNN with LI units performs well on sequential tasks.
Session	Deep Neural Networks for Speech Synthesis

B. Effectivenss of Extended Paragraph Vector

Based on the proposed neural network architecture, we embed sessions, papers and words in the same vector space, which captures some notion of semantics and enables simple vector operations among them. We first demonstrate the shared vector space with an example session and its most similar papers and words in Figure 4. Similar with probabilistic topic models where the most probable words are used to represent a latent topic, the most close (similar) words to a session in the vector space may represent the semantics of a session. We also show that the shared vector space enables effective information retrieval using simple vector arithmetics, such as creating a query based on the vectors of sessions and words to find the most relevant sessions. For example, we may sum the word vectors of *deep*, *neural* and *network* as a query and retrieve the top 10 most similar sessions, which is shown in Table III.

[1]http://www.isca-speech.org/archive

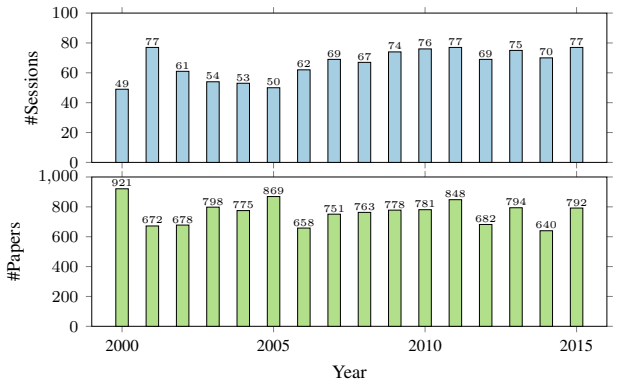

Fig. 3: Number of papers and sessions in Interspeech from 2000 to 2015.

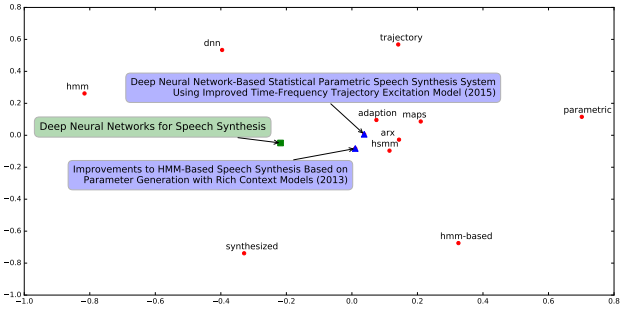

Fig. 4: The example session of *deep neural networks for speech synthesis* (in green) from Interspeech-2015, its top 2 most similar papers (in blue) and top 10 most similar words in the same vector space.

TABLE III: The top 10 most similar sessions for the query of the sum vector of *deep* + *neural* + *network*.

Session ID	Session Title	Similarity
2012Sess002	asr: deep neural networks i, ii	0.708
2015Sess084	neural networks: novel architectures for lvcsr	0.683
2014Sess011	dnn architectures and robust recognition	0.625
2014Sess032	dnn for asr	0.556
2015Sess012	deep neural networks in language and accent recognition	0.524
2015Sess002	feature extraction and modeling with neural networks	0.519
2015Sess040	fast efficient and scalable computing for neural nets	0.486
2013Sess005	asr - neural networks	0.478
2014Sess056	dnn learning	0.470
2015Sess065	robust speech recognition: features, far-field and reverberation	0.451

C. Categorization Results of the Two-Stage Approach

1) Stage-I: In Stage-I, we choose a subset of the EDICS categories if a category has at least one matching session in the corpus. Table IV shows the chosen categories, one example matching session and the number of matching sessions for each category. There are in total 18 categories, covering the major research areas in Interspeech.

TABLE IV: The chosen categories and their matching sessions in Stage-I.

Category ID	Category Title	Example Matching Session	#Sessions
HLT-DIAL	discourse and dialog	discourse and dialogue	6
HLT-LACL	language acquisition and learning	language acquisition	2
HLT-LANG	language modelling	language modelling	2
HLT-LRES	language resources and systems	resources	1
HLT-SDTM	spoken document retrieval and text mining	spoken document retrieval	1
HLT-UNDE	spoken language understanding and computational semantics	spoken language understanding	6
INT-PHON	phonetics, phonology, and prosody	phonetics	10
SPE-ADAP	speech adaptation normalization	adaptation	1
SPE-ANLS	speech analysis	speech analysis and processing i-iii	17
SPE-CODI	speech coding	speech coding and quality assessment	17
SPE-ENHA	speech enhancement	single channel speech enhancement	29
SPE-GASR	general topics in speech recognition	topics in speech recognition	2
SPE-RECO	acoustic modeling for automatic speech recognition	acoustic modeling	3
SPE-ROBU	robust speech recognition	robust speech recognition on aurora	12
SPE-SPER	speech perception and psychoacoustics	speech perception	5
SPE-SPKR	speaker recognition and characterization	speaker recognition	1
SPE-SPRD	speech production	speech production and physiology	28
SPE-SYNT	speech synthesis and generation	speech synthesis	3

2) Stage-II: The second stage assigns the remaining sessions into their most similar categories iteratively and incrementally. We obtained the results of session assignments by setting the starting similarity threshold $\tau = 0.98$, the decay rate $\eta = 0.95$ and the similarity limit $\epsilon = 0.4$. We plot the vector representations of the sessions using the tool t-SNE [25] in Figure 5, with different colors indicating different categories. For example, the three categories of *9: SPE-ANLS (speech analysis)*, *10: SPE-CODI (speech coding)* and *11: SPE-ENHA (speech enhancement)* are very close with each other in the space because they are semantically very similar. On the other hand, *16: SPE-SPKR (speaker recognition and characterization)* and *18: SPE-SYNT (speech synthesis and generation)* are relatively isolated and they are far from each other due to different research foci.

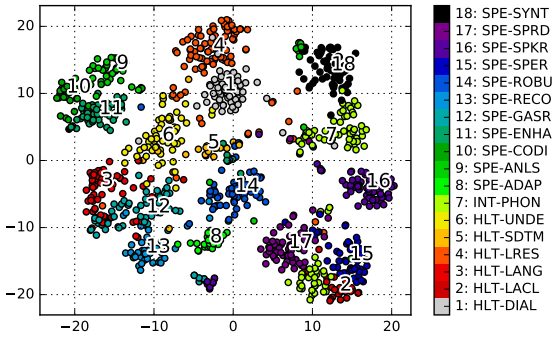

Fig. 5: Visualization of the chosen categories and their matching sessions using t-SNE: semantically similar sessions are embeded closely.

3) Analysis of the Approach: We analyze the behavior of the approach empirically. In the following, we show the effectiveness of the extended paragraph vector model in capturing the semantics of sessions, papers and words, explain the issue of session orders in the assignment procedure, illustrate the effect of the similarity limit, and demonstrate the advantage of incremental assignment for obtaining more stable results.

Iterative Decay of Similarity Threshold We design an iterative procedure to assign the most similar sessions into their categories first. This is implemented by introducing the similarity threshold τ which controls only sessions above the threshold can be assigned in a particular iteration and the threshold is decayed after each iteration. The procedure stops when the threshold is lower than a pre-specified limit ϵ. To show our approach generally assigns the more similar sessions in earlier iterations, we plot how the similarity changes in the assigning order of each session to its most similar category. Figure 6 presents the decreasing trends of the similarities between the assigned sessions and their categories along the iterations. Note that the similarities for *HLT-SDTM (spoken document retrieval and text mining)* has high fluctuation due to low matching sessions in the first iterations, which is also explained in Figure 7.

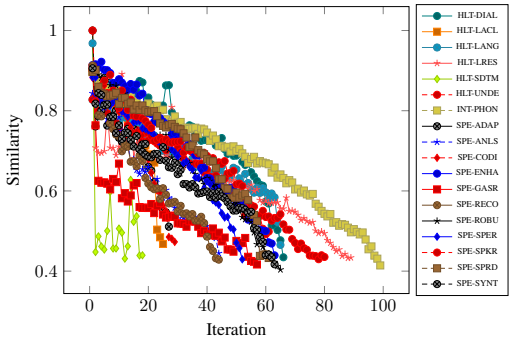

Fig. 6: The decreasing trends of the similarities between the assigned sessions and their categories. The similarity limit ϵ is set as 0.4 empirically.

Effect of the Similarity Limit As there are some sessions not belonging to any of the chosen categories, we set a similarity limit (i.e., ϵ) to discard the sessions having low similarities with all the categories, and decide when the procedure stops. A larger ϵ will discard more sessions but keep the resulting categories more coherent. We can empirically choose ϵ to reach a balance between correct session assignments and wide session coverage. Figure 7 shows the categorization results with different values of ϵ. We can see that all categories obtains a good number of matching sessions when $\epsilon = 0.4$. However, decreasing ϵ to a lower value like 0.5 leads that some categories (e.g., HLT-DIAL, SPE-CODI, SPE-SYNT) have lower number of matching sessions. This means some sessions that are previously assigned to these categories with higher similarities are now assigned with other categories with lower similarities. We should avoid this and thereby choose $\epsilon = 0.4$, below which all categories show increasing number of matching sessions.

Effect of Incremental Assignment The proposed iterative procedure also enables us to assign sessions into categories incrementally. This actually solves another issue of different

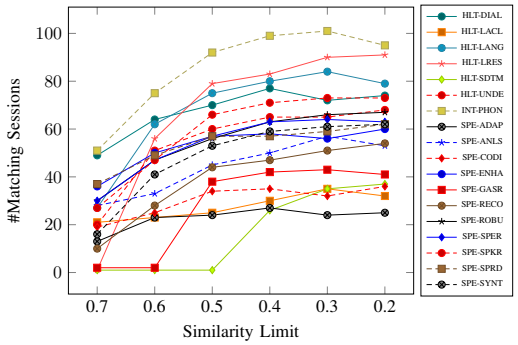

Fig. 7: The different number of matching sessions for each category under different values of the similarity limit ϵ.

categorization results by different orders of the sessions to be assigned. The reason is that the centroid of a category is updated after a session is assigned to the category. This affects the similarity between other unassigned sessions and the centroid and thus leads to different categorization results. To demonstrate the effect of incremental assignment, we run the assignment procedure 10 times, with the sessions randomly ordered. Figure 8 shows the standard deviation (Std.) of the number of matching sessions for each category in 10 different runs, with and without incremental assignment. We can see that without incremental assignment, there are large variances in the number of matching sessions, such the categories of HLT-DIAL, HLT-LACL, HLT-LANG and so on. However, the method of incremental assignment has smaller variances in most categories and thus leads to more stable results.

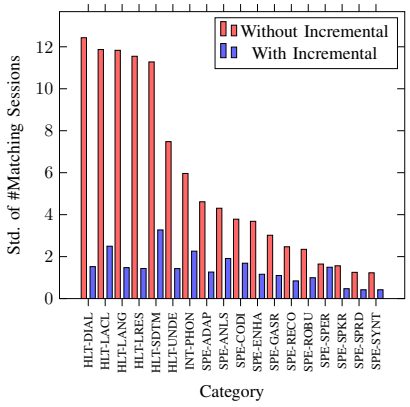

Fig. 8: The standard deviation of the number of matching sessions for each category with and without incremental assignment.

D. Results of Category-Specific Trend Analysis

For each category, we sorted its matching sessions in ascending order of time. The session titles naturally reflect the trend of each category and are easily consumable to users because the titles are derived by domain experts with widely accepted technical terms. As an example, Table V presents the matching sessions for the category of *SPE-SYNT: speech synthesis and generation*, where we can see the major research shifts on *speech synthesis and generation*: concatenation $\rightarrow$ unit selection $\rightarrow$ statistical parametric $\rightarrow$ hmm-based speech synthesis $\rightarrow$ deep neural networks for speech synthesis. Table V also presents the most similar words of each session based on the cosine similarities between a session vector and word vectors. There are some interesting abbreviations related with speech synthesis such as ar (Auto-Regressive), arx (Auto-Regressive with Exogenous Input), hsmm (hidden-semi Markov model), mbrola (a famous multilingual speech synthesizer), pesq (Perceptual Evaluation of Speech Quality), gv (Global Variance), umeda (the first text-to-speech system for English by Umeda et al.), Klatt (a synthesizer), evc (eigenvoice conversion), bn (Bayesian Networks), and so on.

V. Conclusion

This paper aims to categorize a large collection of academic papers into different research areas represented by expert-defined journal categories and analyze the trend of each research area. We exploited the expert-derived conference sessions to facilitate the categorization process because papers under the same session can be assigned to the same category. To capture the semantics of sessions, we proposed a model of extended paragraph vector which learns vector representations of sessions, papers and words jointly in the same space. The obtained session vectors allow us to link sessions with journal categories using an iterative two-stage approach, which first chooses a subset of journal categories covering the major research areas in the corpus and then links sessions with their most similar categories based on session vectors. We further showed the trend of a category by sorting its matching sessions in ascending order of time and obtaining the most similar words of each session based on the cosine similarities between word vectors and session vectors.

Acknowledgment

The authors would like to thank Prof. Xunying Liu, Dr. Kun Li and the three anonymous reviewers for helpful discussions and comments.

References

[1] T. L. Griffiths and M. Steyvers, "Finding scientific topics," *Proceedings of the National academy of Sciences*, vol. 101, no. suppl 1, pp. 5228–5235, 2004.

[2] D. Hall, D. Jurafsky, and C. D. Manning, "Studying the history of ideas using topic models," in *Proceedings of the conference on empirical methods in natural language processing*. Association for Computational Linguistics, 2008, pp. 363–371.

[3] D. M. Blei and J. D. Lafferty, "Dynamic topic models," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 113–120.

[4] Y. Lu, Q. Mei, and C. Zhai, "Investigating task performance of probabilistic topic models: an empirical study of plsa and lda," *Information Retrieval*, vol. 14, no. 2, pp. 178–203, 2011.

[5] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *journal of machine learning research*, vol. 3, no. Feb, pp. 1137–1155, 2003.

[6] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *Journal of Machine Learning Research*, vol. 12, no. Aug, pp. 2493–2537, 2011.

TABLE V: The matching sessions and their corresponding most similar words in descending order of similarity for the category of *SPE-SYNT: speech synthesis and generation*.

Year	Session Title	Most Similar Words
2000	generation and synthesis of spoken language 1, 2	concatenation, concatenative, generate, syllable-level, mary, prosody, diphone, tying, high-quality, corpus-based, contours, emotive, sub-syllable, expressive, synthesiser
	generation and synthesis of spoken language 3	morphing, excited, smooth, prevent, module, symmetry, adopts, high-quality, modification, synthesiser, labor, synthesizer, waveform, product, sliding
	generation and synthesis of spoken language (poster)	greek, title, generating, rules, generate, mark-up, phrases, peculiar, corpus-based, domain-specific, text, generator, solve, grammar, text-to-speech
2001	systems and prosody	phrases, tags, intonational, prosody, phrasing, grammatical, characters, accentual, generate, disambiguation, generator, rules, interrogative, clinically, annotate
	concatenation	concatenation, inventory, concatenative, encapsulating, realization, realizations, discontinuities, critical, animated, inventories, circle, zeros, vocalic, choice, natural-sounding
	prosody	intonation, discourse, prosody, linguistic, syntactic, prosodic, basic, annotate, intonational, contours, symbolic, phrasing, interrogative, adjusted, stem-ml
	miscellaneous	concatenative, high-quality, concatenation, counting, modified, diphone, templates, festival, mixed-language, contours, generate, singing, ar, waveform, synthesiser
2002	speech synthesis	concatenative, corpus-based, prosody, high-quality, synthesis, concatenation, unit, syllable-level, synthesized, generate, title, dependency, inventory, annotate, generating
	speech synthesis: alternative views	expressive, animated, peculiar, synthesised, corpus-based, festival, educational, polyglot, synthetic, synthesized, prosody, intuitively, text-to-speech, creation, imitating
	speech synthesis: prosody	prosody, contours, intonation, contour, templates, arranged, text-to-speech, interrogative, dependency, expressive, phrase, diphone, intonational, peculiar, phrasing
	speech synthesis: **unit selection**	join, sagittal, concatenation, re-synthesized, dissimilarity, similarity, definition, synthesized, forcing, reconstructed, cost, tractable, objective, optimizing, informal
2003	**speech synthesis: unit selection 1, 2**	concatenative, unit, concatenation, join, diphone, inventory, circle, text-to-speech, expressive, generate, selection, contours, annotate, festival, retains
	towards synthesizing expressive speech	expressive, controlling, creation, singing, prosody, heads, expression, animated, psychological, text-to-speech, educational, actor, obligatory, non-verbal, expressing
	speech synthesis: miscellaneous 1, 2	singing, grapheme-to-phoneme, voices, illustrative, clinically, vibrato, direct, discontinuities, phrase-final, laryngectomees, intonation, high-quality, contours, tense, possesses
	speech synthesis: voice conversion and miscellaneous topics	high-quality, retains, modifying, synthesizer, waveform, parametric, cross-sectional, shapes, grapheme-to-phoneme, functions, straight, pleasant, direct, singing, laryngeal
2004	**spoken language generation and synthesis**	generate, generating, generation, high-quality, grapheme-to-phoneme, synthesize, rules, text-to-speech, concatenative, polyglot, retains, prosody, compound, annotate, diphone
	processing of prosody by humans and machines	expressive, sm, breaks, interrogative, expressivity, relation, phrase, discourse, avatar, phrasing, devise, prosody, juncture, hand, traits
2005	the blizzard challenge 2005	festival, text-to-speech, creation, concatenative, animated, tts, expressive, home, capt, singing, experienced, educational, controlling, platforms
	multilingual tts	corpus-based, tts, festival, text-to-speech, polyglot, concatenative, synthesize, mary, expressive, nagoya, greek, ssml, writing, building, under-resourced
	text-to-speech i, ii	generate, generating, parametric, singing, generated, proper, expressive, diphone, synthesis, pronunciations, clinically, lexicon, baseform, synthesize, diphone-based
	tts inventory	concatenative, high-quality, inventory, diphone, encapsulating, synthesis, discontinuities, graphics, join, generate, unit, animated, expressive, concatenation, subtractive
2006	**text-to-speech i, ii**	concatenative, inventory, polyglot, unit, concatenation, diphone, manipulate, singing, festival, generating, corpus-based, retains, rules, mary, diphone-based
	corpus-based synthesis	aperiodicity, modifying, singing, discontinuities, synthesize, concatenative, diphone, noise-only, festival, clinically, modified, selection, segment, modifies, reconstruct
	voice morphing	evc, eigenvoice, one-to-many, gmm-based, singer, pleasant, modifying, converted, grapheme-to-phoneme, retains, hsmm, conversion, pre-stored, tract, singing
	speech synthesis	creating, editing, creation, facilitate, singing, expressive, author, animated, electrolarynx, practically, pre-stored, playback, talking, preserving, generate
2007	prosodic modeling i, ii	phrase, prosodic, prosody, phrasing, sentence, intonation, phrasal, generating, characters, break, generation, annotate, predicting, assigns, breaks
	speech synthesis i, ii	concatenative, synthesis, high-quality, annotate, text-to-speech, polyglot, creation, mixed-language, unnatural, expressive, inventory, annotation, unit, tts, clinically
	unreviewed papers for special sessions	singing, controlling, api, laryngeal, mbrola, assistive, air, tracked, voices, electrical, animation, educational, industry, hands, control
2008	speech synthesis methods i, ii	concatenative, generate, generated, singing, expressive, synthesize, electrolarynx, modifying, diphone-based, generation, selection, polyglot, modifies, retains, high-quality
	speech synthesis: prosody and emotion i, ii	contours, intonation, contour, fujisaki, explicit, retains, prosody, interrogative, phrasing, rising-falling, mathematical, straight, assimilation, adjusted, mary
2009	**statistical parametric synthesis i**	parametric, concatenative, trajectory, hsmm, hmm-based, vocoder, mixed-language, hmmbased, high-quality, unit-selection, resynthesis, naturalness, polyglot, singing, text-to-speech
	prosody, text analysis, and multilingual models	prosody, intonation, phrasing, contours, explicit, mary, manipulate, expressive, three-layer, assimilation, fujisaki, vietnamese, predicting, inadequate, abbreviation
	unit-selection synthesis	covariances, selection, circle, optimizing, heuristic, join, optimize, hand-crafted, letter-to-sound, trajectory, summing, handcrafted, product, arranged, concatenation
	voice transformation i, ii	retains, gmm-based, eigenvoice, singing, emotive, evc, singer, converted, one-to-many, excitation, conversion, modifying, timbre, waveform, transforming
	statistical parametric synthesis ii	parametric, hmm-based, concatenative, hsmm, trajectory, synthesis, hmmbased, mge, mixed-language, vocoder, polyglot, synthesizer, resynthesis, retains, mary
	speech synthesis methods	concatenative, high-quality, expressive, synthesis, synthesizer, festival, synthesized, polyglot, synthesizing, generating, resynthesis, govern, synthesize, waveform, generation
2010	speech synthesis: unit selection and others	concatenation, annotate, selection, mixed-language, generate, concatenative, chunking, synthesized, unit, vocoded, naturalness, inventory, join, circle, diphone
	speech synthesis: **hmm-based speech synthesis i, ii**	hsmm, polyglot, synthetic, topology, synthesized, trajectory, festival, expressive, naturalness, retains, unit-selection, hmmbased, hmm-based, cluster, synthesis
	speech synthesis: miscellaneous topics	affective, facilitate, natural-sounding, spoken-language, clinicians, intonation, attempt, prosody, innovation, baldi, non-verbal, appealing, narrative, inform, stringent
	voice conversion	effectively, eigenvoice, emotive, non-parallel, variational, smoothed, converted, gmm-based, singer, nonnegative, modifying, apply, transforming, magnitudes, arbitrary
	voice conversion and speech synthesis	singing, emotive, creaky, context-sensitive, qualities, modifying, controlling, natural-sounding, manipulate, mimic, timbre, modal, intuitively, arbitrary, silverman
2011	**hmm-based speech synthesis i, ii**	trajectory, parametric, topology, resynthesis, re-estimation, cluster, multi-form, hsmm, autoregressive, straight, expressive, arx, hmm-based, three-layer, high-quality
	speech synthesis - unit selection and hybrid approaches	animated, text-to-speech, mixed-language, judgments, concatenative, expressive, formal, vocoded, klatt, informal, naturalness, modified, templates, library, prosody
	speech synthesis - selected topics	rich, annotate, prosody, syntactical, tags, phrasing, automating, crawling, freely, text, selecting, mixed-language, part-of-speech, retrieved, generate
	voice conversion and speech synthesis	synthesizing, emotive, mimic, singing, synthesize, donor, synthesized, eigenvoice, qualities, creaky, conversion, converted, natural-sounding, synthetic, eam
2012	speech synthesis: prosody	prosody, contours, intonation, expressive, superpositional, fujisaki, mary, contour, emotive, klatt, text-to-speech, ar, command-response, generate, re-estimation
	speech synthesis: intelligibility	naturalness, synthesized, synthetic, intelligibility, quality, re-synthesized, appropriateness, pesq, vocoded, reconstructed, perceived, informal, vocoder, electrolaryngeal, distortion
	speech synthesis: adaptation	emotive, synthetic, hsmm, expressive, converted, expressiveness, timbre, synthesize, singing, naturalness, polyglot, neutral, donor, modifying, synthesized
	hmm synthesis i, ii	trajectory, parametric, straight, vocoder, lp, sinusoidal, smoothed, pitch-synchronous, modification, converted, envelope, over-smoothing, all-pole, ar, modified
	speech synthesis	singing, clinically, illustrative, voices, text-to-speech, animated, mbrola, non-modal, discontinuities, modal, actor, expressive, controlling, klatt, laryngectomees
	speech synthesis: selected topics	break, surface, bc, mixed-language, generate, phrasal, accentual, phrase, ambiguous, plausible, assimilation, merger, optional, phrasing, focal
2013	**speech synthesis i, ii**	singing, expressive, synthesized, waveform, synthesize, modifying, converted, emotive, high-quality, naturalness, retains, synthetic, modifies, hsmm, synthesizing
	speech synthesis - prosody and emotion	expressive, hsmm, appropriateness, prosody, synthesizer, rising-falling, expressiveness, animated, sample, emotive, festival, synthesiser, neutral, naturalness, labels
	speech synthesis - various topics	curve, mixed-language, singing, concatenation-based, naturalness, vocoder, singular, data-derived, cepstrum-based, retains, doubt, melp, concatenative, creaky, sounding
2014	prosody processing	segmental, filled, appropriateness, prosody, naturalness, dependency, passages, vibrato, el, closely, prosodic, sentence-final, text-to-speech, expressive, reconocimiento
	speech synthesis i-iii	emotive, expressive, synthesized, synthetic, high-quality, synthesizer, prosody, hsmm, singing, converted, lp, diphone-based, naturalness, synthesize, expressiveness
	statistical parametric speech synthesis	singing, **klatt**, retains, sounded, hsmm, synthesised, synthetic, phonatory, trajectory, depressed, copied, synthesized, festival, sharp, judgements
	deep neural networks for speech generation and synthesis	converted, smoothed, timbre, nn, folding, eigenvoice, routines, singing, stack, control, shape, rnn, ring, variances, parametric
2015	speech synthesis 1-3	singing, expressive, synthesize, emotive, voices, modifying, synthetic, polyglot, generated, festival, converted, concatenative, synthesized, prosody, retains
	deep neural networks for speech synthesis	trajectory, hmm-based, arx, hsmm, parametric, adaption, dnn, hmm, synthesized, maps, umeda, synthesis, fine-tuning, bn, ann
	statistical parametric speech synthesis	hmm-based, topology, parametric, hsmm, gv, trajectory, composition, unit-selection, synthetic, diphone-based, mge, hmmbased, concatenative, clean-speech, hmm
	prosody modeling for speech synthesis	arx, fujisaki, intonation, explicit, prosody, contours, all-pole, dependency, concatenation, contour, generate, refining, ar, segmental, polynomial
	evaluation of speech synthesis	informal, synthesized, opinion, synthetic, preference, objective, hsmm, voices, naturalness, subjective, listening, re-synthesized, blizzard, ratings, assessed

[7] T. Mikolov, Q. V. Le, and I. Sutskever, "Exploiting similarities among languages for machine translation," *arXiv preprint arXiv:1309.4168*, 2013.

[8] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 2011, pp. 142–150.

[9] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[10] J. Mitchell and M. Lapata, "Composition in distributional models of semantics," *Cognitive science*, vol. 34, no. 8, pp. 1388–1429, 2010.

[11] R. Socher, B. Huval, C. D. Manning, and A. Y. Ng, "Semantic compositionality through recursive matrix-vector spaces," in *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Association for Computational Linguistics, 2012, pp. 1201–1211.

[12] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, vol. 1631. Association for Computational Linguistics, 2013, p. 1642.

[13] J. Weston, S. Bengio, and N. Usunier, "Large scale image annotation: learning to rank with joint word-image embeddings," *Machine learning*, vol. 81, no. 1, pp. 21–35, 2010.

[14] N. Srivastava and R. R. Salakhutdinov, "Multimodal learning with deep boltzmann machines," in *Advances in neural information processing systems*, 2012, pp. 2222–2230.

[15] Q. V. Le and T. Mikolov, "Distributed representations of sentences and documents." in *ICML*, vol. 14, 2014, pp. 1188–1196.

[16] P. Xie and E. P. Xing, "Integrating document clustering and topic modeling," *Proceedings of the 30th Conference on Uncertainty in Artificial Intelligence*, 2013.

[17] C. C. Aggarwal and C. Zhai, "A survey of text clustering algorithms," in *Mining text data*. Springer, 2012, pp. 77–128.

[18] R. Feldman and I. Dagan, "Knowledge discovery in textual databases (kdt)." in *KDD*, vol. 95, 1995, pp. 112–117.

[19] B. Lent, R. Agrawal, and R. Srikant, "Discovering trends in text databases." in *KDD*, vol. 97, 1997, pp. 227–230.

[20] K. Rajaraman and A.-H. Tan, "Topic detection, tracking, and trend analysis using self-organizing neural networks," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2001, pp. 102–107.

[21] J. Bollen, M. L. Nelson, G. Manepalli, G. Nandigam, and S. Manepalli, "Trend analysis of the digital library community," *D-Lib Magazine*, vol. 11, no. 1, pp. 1082–9873, 2005.

[22] X. Wang and A. McCallum, "Topics over time: a non-Markov continuous-time model of topical trends," in *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2006, pp. 424–433.

[23] X. Wei, J. Sun, and X. Wang, "Dynamic mixture models for multiple time-series." in *IJCAI*, vol. 7, 2007, pp. 2909–2914.

[24] A. Ahmed and E. P. Xing, "Timeline: A dynamic hierarchical dirichlet process model for recovering birth/death and evolution of topics in text stream," *arXiv preprint arXiv:1203.3463*, 2012.

[25] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008.