Tenth International Conference on Spoken Language Processing - ICSLP / Ninth INTERSPEECH, Brisbane, Australia, September 2008

Using Latent Dirichlet Allocation to Incorporate Domain Knowledge For Topic Transition Detection

Xiaodan Zhu, Xuming He, Cosmin Munteanu, Gerald Penn

Department of Computer Science, University of Toronto, Toronto, Canada
{xzhu,hex,mcosmin,gpenn}@cs.toronto.edu

Abstract

This paper studies automatic detection of topic transitions for recorded presentations. This can be achieved by matching slide content with presentation transcripts directly with some similarity metrics. Such literal matching, however, misses domain-specific knowledge and is sensitive to speech recognition errors. In this paper, we incorporate relevant written materials, e.g., textbooks for lectures, which convey semantic relationships, in particular domain-specific relationships, between words. To this end, we train latent Dirichlet allocation (LDA) models on these materials and measure the similarity between slides and transcripts in the acquired hidden-topic space. This similarity is then combined with literal matchings. Experiments show that the proposed approach reduces the errors in slide transition detection by 17-41% on manual transcripts and 27-37% on automatic transcripts.

Index Terms: slides transition detection, boundary detection.

1. Introduction

Presentations delivered with slides are pervasive in many academic and business spheres. Therefore, it is no surprise that a large number of presentations have been and will be recorded, such as lectures, seminars and internal corporate presentations. Knowing the time stamps of lecture topic transitions is greatly beneficial to navigating these multimedia archives. Topic transitions are also the natural boundaries to index these archives for the purpose of searching. In addition, topic transitions have also proven useful in automatic summarization of presentations[1].

Topic transitions, however, are only directly accessible through occasional natural language cues such as "Turning now to ...," "Our next topic ..." etc., so much of the work that aspires to use them (including [1]) uses *slide transitions* instead.[1] These are the time stamps indicating when the lecturer changes the slide displayed on a projector. A straightforward way of acquiring slide transitions is to mark them during data recording, e.g., through recording certain keyboard or clicking events invoked by the lecturer. Such recordings may not be available in many presentation environments, nor in many older recorded archives, nor are they preferable in *passive* recording environments, as discussed in [3]. There can also be problems with other keyboard activity on the same device, such as running demos, not to mention speaker or device error (accidentally paging forward by too many slides), as well as intentional backwards navigation, in which the speaker reverses direction in the slides in order to answer a question or emphasize an earlier point.

As a result, some research has attempted to detect slide transitions automatically. Some do so by analyzing video recordings[4][5][6], i.e., by detecting the slide area on the video canvas and looking for changes in that area. Such approaches depend heavily on the recording set-up and video quality, the variety of presentation environments, and the positioning, panning and brightness adaptation of the camera.

Another way of detecting slide transitions is through the audio channel: matching slide content with presentation transcripts. Only a lapel or head-mounted microphone is typically required here. Previous work has studied the direct matching of slide content and presentation transcripts using certain similarity metrics[7][8]. Only slides and transcripts themselves, however, have been used to estimate the similarities.

In this paper, we explore the pragmatic possibility of more accurately guessing topic transitions using evidence not only from slide transitions, but also from slide content, attained through automatic speech recognition, as well from electronically available texts on related subject matter. This approach is particularly interesting because, in principle, it can be extended to obtain an even finer granularity of topics than a transition sequence — more of the structure of a table of contents, with both coarse and subtler transitions. Textbooks usually have these, and where lectures closely follow a textbook, some of this structure can be co-opted. Even on slides alone, this is occasionally reflected by "bulleting" main points that are covered in the lecture. Such structured multimedia archives provide a more detailed means of navigating the archives, and are also useful for presentation summarization.

Even where presentations are not based on or accompanied by supplementary reading material, auxiliary written sources obtained elsewhere on the same subject can be used to collect more accurate semantic co-occurrence statistics to drive a spectral dimensionality reduction. Such reductions are crucial to avoiding chance keyword paraphrases and ASR transcription errors between semantically related documents or sections of documents. The latter is a particularly acute problem as speaker-independent models in the lecture domain often have word error rates (WERs) of more than 40%.

As we are positing the existence of hidden but well-defined topics within lectures, we train latent Dirichlet allocation (LDA) models on relevant written materials and measure slide-transcript similarities in the acquired hidden-topic space. These are then combined with literal word-level matching that is calculated directly between slides and transcripts. Our experiments show that the proposed approach reduces the errors in topic transition detection by 17-41% on manual transcripts and

[1] A notable exception is the TextTiling method[2], although this has not enjoyed widespread usage on spoken language transcripts, nor does it avail itself of extra-transcriptional sources of evidence such as slides or related texts.

27-37% on automatic transcripts. We also analyze the situations in our test data where the method produces large errors.

2. Problem formulation

2.1. Alignment framework

Research on finding correspondences in parallel texts pervades natural language processing (NLP). In statistical machine translation[10], words or phrases from each bilingual sentence pair need to be aligned in order to train translation models. In automatic text summarization, the correspondence between human-written summaries and their original texts has been studied. Some research [9], for example, has decomposed sentences of human-written summaries to decide whether and where the texts are cut and pasted from the original documents.

In keeping with much of this work, we formulate the transition detection problem in an HMM framework. We are given a sequence of slides $S = s_1, s_2, ..., s_m$, and corresponding transcript $T = t_1, t_2, ..., t_n$, where each t_i is a window of words starting from the i^{th} word. The window size is adjustable and hence t_i can contain just one word or a sequence of words. In the latter case, t_i shares words with some windows before and after it. A slide s_i corresponds to a hidden state, and t_i corresponds to an output symbol. For a given output sequence, i.e., transcript $T = t_1, t_2, ..., t_n$, once the optimum hidden state sequence is decided, the correspondence between slides and transcripts is indicated and hence the slide transition points are discovered. These in turn approximate topic transition points.

The output probabilities $p(t_j|s_i)$ are estimated using normalized similarities between slides and transcript windows: $p(t_j|s_i) = sim(t_j, s_i)/\sum_k sim(t_k, s_i)$. In our baseline and experimental methods, we employ several common distance metrics to directly measure the similarities between slides and transcript windows: L1 (Manhattan) distance, L2 (Euclidean) distance, KL divergence, and cosine distance. The state transition probabilities $p(s_j|s_i)$ are set to ensure that a slide can only transit to itself (with probability λ) or to the next slide (with probability $1 - \lambda$).[2] With this assumption, the transition probabilities have only one parameter λ, as shown in the formula below, which is easy to estimate with limited data.

$$p(s_j|s_i) = \begin{cases} \lambda & : j = i \\ 1 - \lambda & : j = i + 1 \\ 0 & : otherwise \end{cases}$$

We use $\lambda = 0.9$ for our experiments, which was determined on a development set. Since almost all presentations start from the first slide, the initial state probability can be set as: $p(s_i) = 1$ if $i = 1$ and 0 otherwise. Once all the parameters above are estimated, a standard decoding algorithm can be applied to determine the hidden state (slide) sequence.

2.2. Topic models

Relevant written materials provide semantic, i.e., domain-specific knowledge for understanding presentation content. For our task, we incorporate this auxiliary information to improve the similarity measurements between slides and transcripts. To this end, we adopt a well-known topic model, Latent Dirichlet Allocation (LDA) [11]. LDA is a generative model for modelling documents, in which each document is regarded as a bag of words and generated by taking a mixture of hidden topics. For example, a document on prototyping evaluation methods in computer science is likely to be a mixture of words from the topic of EVALUATION and the topic of PROTOTYPING. Each topic itself is represented by a distribution over words, and this distribution is obtained through training LDA models over a collection of documents. Once the models are obtained, a document can be represented by its distribution over the topics in LDA. We can then calculate the similarity of two documents based on this new representation. The domain-specific semantic knowledge, which is evident through word collocations, is naturally considered in this new similarity measure. For example, a slide that mentions *FSA* (finite state automata) but not *automata* can have a non-zero similarity score with the corresponding part of a transcript that mentions *automata* but not FSA, since both *FSA* and *automata* appear in textbooks on the same topic. The new measure can be naturally combined with one of the baseline similarity measures obtained through matching slides and transcripts directly, to estimate the output probabilities $P(T|S)$ of the HMM. In this paper, we use a linear combination of the baseline $P(T|S)$ and topic-based $P(T|S)$, each derived from its own normalized similarity computations.

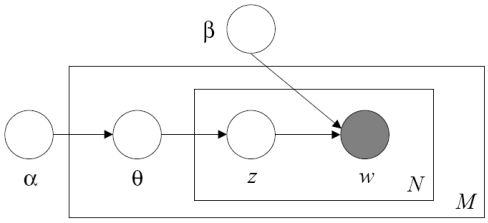

Figure 1: *Latent Dirichlet Allocation.*

Compared to singular value decomposition, a widely used dimensionality reduction method, LDA provides a more sophisticated model of word count distribution. Although the probabilistic analogue of SVD (pLSI) uses a similar model assumption, it is not fully generative. In particular, it is difficult to estimate the probability of a new document not appearing in the training data. This is critical for our task, among others — we train the models on relevant written documents (textbooks) and then need to assign a probability to a slide and transcript window, which are not part of the training data.

A graphical representation of LDA for a corpus is shown in Figure 1. It is a three-level hierarchical Bayesian model. Each document is represented as a set of N words (the inner plate), and the corpus has M documents (the outer plate). Each word w in a document is generated from a topic distribution β_z, which is a multinomial distribution over words. The topic indicator z of the word w is assumed to have a multinomial distribution θ over topics, which in turn has a Dirichlet prior with parameter α. The parameters of the LDA model can be estimated by maximizing the data likelihood of training documents. We set the hyperparameter α as in [11]. Then we integrate out θ and learn β using the EM algorithm. The E step is based on a Gibbs sampling of topic indicators z, and the M step only needs to calculate the sufficient statistics for β. For our task, we train LDA models on textbooks, in which a subsection, as defined by its table of contents, is treated as a document. Once the model is trained, we can map a slide or transcript window into the hidden topic space by computing its θ. This is given by an EM procedure that treats θ as a parameter with z missing.

[2] One can change the state transition matrix to allow for more flexible models, e.g., those permitting transitions to previous slides or skipping slides, but we do not discuss these here.

3. Experiment set-up

We use a corpus of lectures recorded at a large research university. Only the lecturer's voice is recorded, using a head-mounted microphone. The lectures that we have used in our experiments are from two undergraduate computer science courses: a second year introductory course and a fourth-year advanced course, each with a different instructor. The former course is an introduction to Unix and several programming environments. We use five lectures for which we have both manual and automatic transcripts. The average length of a class is 45 minutes, while the average number of slides is approximately 13 per class. The course is based on four textbooks, which contain an aggregate of 868 subsections. We treat each subsection as an individual document and use them to train an LDA model. One lecture is held out as the development set to tune undecided parameters, such as the number of hidden topics (300), the size of the transcript windows (between 0.2 and 0.6 times the number of words in a lecture's transcript divided by the number of slides) and the λ in the HMM transition model (0.9). Stop-words are removed and stemming is applied to the textbooks before training.

The fourth-year course is a human-computer interaction (HCI) course. We have four recorded classes with both manual and automatic transcripts. The average length of a class is 45 minutes, with 28 slides per lecture — approximately twice as many as for the introductory course. The difference is due to the fact that introductory course's lectures often involve many example programs, and more interaction with students. The advanced course uses only one textbook, which has 186 subsections, resulting in only 100 hidden topics being trained.[3]

The evaluation metric of our task is straightforward — automatically acquired transitions are compared against the gold standard to calculate a collection of offsets measured in number of words. The offsets are averaged over all transitions to evaluate the transition detection performance on the whole corpus. We call these offsets *transition errors*. Our gold standard for topic transitions was obtained through manual annotation. The annotator was given the lecture video, transcripts, and slides to decide topic transitions. Note that topic/content transitions may not happen at exactly the same time as the instructor changes the slides. For example, right after the instructor of a lecture switches slides, he may receive questions from students on the previous slide and therefore continues to talk about the previous slide's material even though the new slide is being displayed. In such cases, the annotator marks real content transitions.

4. Experimental results

4.1. Detection performance

Table 1 shows the experimental results obtained using manual transcripts. The first row counts baseline transition errors without using the LDA models trained on textbooks. In this and the following row, we report the average word offset score per transition, the sum being obtainable by multiplying these numbers by computing the total number of transitions in the corpus (172).[4] Incorporating an LDA model trained on textbooks reduces transition errors with all four standard distance metrics. The relative error reductions range between 17% - 41%.

Table 1: *Transition errors on manual transcripts*

	L1	L2	KL	COS
No textbook models	18	24	24	32
Using LDA models	15	19	20	19
Reduction	17%	21%	17%	41%

Table 2 presents the experimental results on automatically generated transcripts. The WER of the transcripts is 45% on average. The transcripts were generated with the SONIC toolkit [13]. The acoustic model was trained on 30 hours of the Wall Street Journal Dictation Corpus. The language model was trained on corpora obtained from the Web through searching the words appearing on slides as suggested by Munteanu et al. [14]. Table 2 reveals that for all but the cosine distance metric, larger error reductions are achieved on automatic transcripts than on manual transcripts. This can be observed by comparing the third rows of Table 1 and Table 2. Focussing on column L1 in these two tables, we can see that without using the LDA models trained on the textbooks, the transition errors increase from 18 to 29 (61% relative increase) due to the speech recognition errors; after incorporating textbooks, the transition errors rise from 15 to just 21 words (40% relative increase). This means that with L1 distance, the use of textbooks makes transition detection more robust to speech recognition errors. Actually, the usefulness of written documents in spoken document processing has also been observed in spoken document retrieval (SDR), where query and document expansion using written documents is also very effective[15].

Table 2: *Transition errors on automatic transcripts*

	L1	L2	KL	COS
No textbook models	29	32	30	51
Using LDA models	21	22	22	32
Reduction	28%	31%	27%	37%

4.2. Error analysis

We conduct a further analysis to understand the detailed distribution of transition errors. As shown in Table 3, we group transitions by their baseline errors (no textbooks, L1 distance, manual transcripts). Transitions with more baseline errors are more likely to be improved with the use of textbooks. For example, among transitions with baseline errors less than 5 words, using textbooks only improve 37.2% of them, but for transitions with baseline errors over 20 words, using textbooks helps 51.2% of them. This agrees with our intuition: when a baseline system makes small errors, it means literal matching works well, i.e., there are enough words overlapping between slides and transcripts. In such cases, domain-specific semantic knowledge does not provide much more additional information.

Figure 2 depicts the absolute error reductions (below the x-axis) and increases (above the x-axis) for each slide transition. The x-axis contains slides sorted by their (L1) baseline errors

[3] Considering that this course's discussion of its sole textbook is more detailed than the introductory course's, we trained an alternative model on 300 hidden topics, using each of the textbook's 1186 paragraphs as documents rather than its subsections. The paragraph-level model had similar performance to the subsection-level model, so all the results reported here use subsection-level LDA models.

[4] The recall/precision metric is considered unsuitable for topic segmentation[12]. Furthermore, our approach is different from regular topic segmentation in that the number of segments obtained is same as in the gold standard, so we can measure offsets directly instead of using the more complicated metrics designed for general topic segmentation.

Table 3: *Improved transitions grouped by baseline errors*

Baseline errors	< 5	$5 - 19$	> 19
Improved transitions (%)	37.2	43.0	51.2

in increasing order. These are shown in grey. The reduction or increase that results from using textbooks, in number of words, is shown in black for each slide transition.

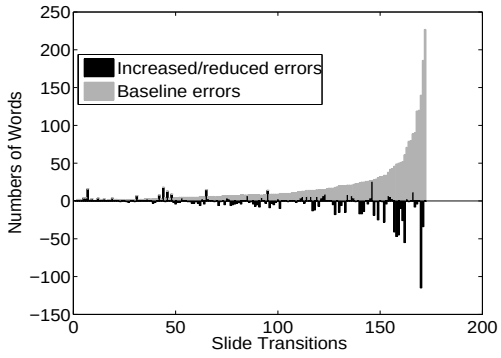

Figure 2: *Per-transition error deltas from using textbooks.*

It shows that while error reduction magnitudes roughly increase in correlation to the baseline number of errors (correlation coefficient: -0.4462), error increases do not (correlation: 0.0429) and are smaller overall. Thus the improvement comes more from the transitions with worse baseline performance.

Even our best configuration (manual transcripts, L1 distance, with textbooks) fails in several transitions with very large offsets. Figure 3 is the histogram of errors made by this configuration. The x-coordinate is transition errors (beginning with zero) and the y-coordinate counts the number of transitions with that number of transition errors. From the figure, we can observe that there are only 11 transitions (6% of the total number of transitions) with transition errors over 50 words. The offsets on these 11 transitions, however, account for 40% of the total sum. Otherwise, the automatic detection performs well on most transitions — on over 60% of the transitions, the transition errors are smaller than 10 words. Nine of these 11 transitions, furthermore, are adjacent to a slide with very little text on it — these slides either contain mainly images or contain example programming code, and so they provide little information to match with transcripts. The remaining two transitions are between slides that differ only slightly from each other. In such cases, transitions are difficult to decide on, too.

5. Conclusions and future work

This paper studies the automatic detection of topic transitions for recorded presentations. Our experimental results show that incorporating textbooks with the topic model LDA improves the performance of transition detection on both manual and automatic transcripts over a baseline that uses slides alone. Incorporating textbooks also makes the detection task more robust to speech recognition errors on most distance metrics. The approach we use cannot handle a few transitions well, such as those adjacent to slides with little textual content, or little textual differentiation.

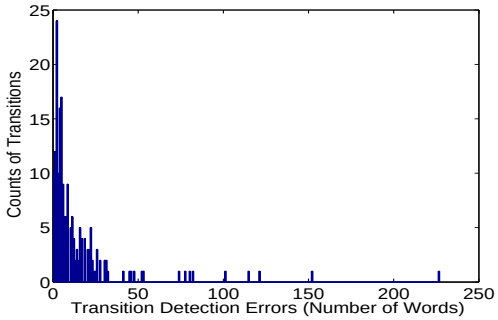

Figure 3: *Histogram of transition errors.*

A direct follow on to this study would be to extend the approach to align slide content with presentation transcripts on a finer level, e.g., the slide "bullet" level. This would not only provide a more detailed means of navigating recordings, but can be useful for other tasks such as summarization and automatic slide generation. In addition, comparing LDA with other models such as latent semantic analysis may render a further understanding of these tasks.

6. References

[1] He, L., Sanocki, E., Gupta, A., and Grudin, J., Auto-summarization of audio-video presentations, Proc. 7th ACM Multimedia, 1999

[2] Hearst, M. A., Multi-Paragraph Segmentation of Expository Text, Proc. 32nd ACL, 1994.

[3] Mukhopadhyay S. and Smith B., Passive capture and structuring of lectures, Proc. 7th ACM Multimedia, 1999

[4] Liu T., Hjelsvold R., and Kender J.R., Analysis and enhancement of videos of electronic slide presentations, Proc. IEEE ICME, 2002.

[5] Fan Q., Barnard K., Amir A., Efrat A. and Lin M., Matching slides to presentation videos using SIFT and scene background, Proc. 8th ACM Int'l Workshop on Multimedia Information Retrieval.

[6] Wang F., Ngo C.W., and Pong T.C., Synchronization of Lecture Videos and Electronic Slides by Video Text Analysis, Proc. 11th ACM Multimedia, 2003.

[7] Chen Y. and Heng W.J., Automatic synchronization of speech transcript and slides in Presentation, Proc. Int'l Symposium on Circuits and Systems, 2003.

[8] Ruddarraju R., Indexing Presentations Using Multiple Media Streams, M.S. Thesis, Georgia Institute of Technology, 2006.

[9] Jing H., Using Hidden Markov Modeling to Decompose Human-Written Summaries, Computational Linguistics 28(4):527-543 (2002).

[10] Och F.O. and Ney H., A systematic comparison of various statistical alignment models. Computational Linguistics, 29(1):19-51 (2003).

[11] Blei, D. M., Ng, A. Y., and Jordan, M. I., Latent Dirichlet allocation. Journal of Machine Learning Research 3:993-1022 (2003).

[12] Beeferman D., Berger A., Lafferty J., Statistical Models for Text Segmentation. Machine Learning, 34(1-3):177-210 (1999).

[13] Pellom B. L., SONIC: The University of Colorado continuous speech recognizer, Tech. Rep. TR-CSLR-2001-01, University of Colorado, 2001.

[14] Munteanu, C., Penn, G., Baecker, R., Web-Based Language Modelling for Automatic Lecture Transcription. In Proc. 8th INTERSPEECH, 2007.

[15] Singhal, A., Choi, J., Hindle, D., Hirschberg, J., Pereira, F., Whittaker, S., AT&T at TREC-7 SDR Track, Proc. TREC, 1997.