A Probabilistic Model of Meetings that Combines Words and Discourse Features

Mike Dowman[*1] (mike@sacral.c.u-tokyo.ac.jp), Virginia Savova[2] (savova@mit.edu),
Thomas L. Griffiths[3] (tom_griffiths@berkeley.edu), Konrad P. Körding[4] (konrad@koerding.com),
Joshua B. Tenenbaum[2] (jbt@mit.edu), Matthew Purver[5] (mpurver@stanford.edu)

[1]Department of General Systems Studies,
The University of Tokyo, 3-8-1 Komaba,
Meguro-ku, Tokyo 153-8902, Japan
Phone: +81354544364
Fax: +81354544315

[2]Deptartment of Brain and Cognitive Sciences, MIT,
Cambridge, MA 02139
Phone: +16174522010
Fax: +16172538335

[3]Department of Psychology,
University of California at
Berkeley, Berkeley, CA 94720
Phone: +15106425292
Fax: +15106425293

[4]Physical Medicine and Rehabilitation,
Northwestern University,
Chicago, IL 60611
Phone: +17737826327
Fax: +18474914928

[5]CSLI, Stanford University,
Stanford CA 94305
Phone: +16507232030
Fax: +16507252166

EDICS Classification: SLP-UNDE

Abstract

In order to determine the points at which meeting discourse changes from one topic to another, probabilistic models were used to approximate the process through which meeting transcripts were produced. Gibbs sampling was used to estimate the values of random variables in the models, including the locations of topic boundaries. The paper shows how discourse features were integrated into the Bayesian model, and reports empirical evaluations of the benefit obtained through the inclusion of each feature and of the suitability of alternative models of the placement of topic boundaries. It demonstrates how multiple cues to segmentation can be combined in a principled way, and empirical tests show a clear improvement over previous work.

I Introduction

Much work in computational linguistics attempts to discover latent structure in corpora of natural language texts through an inductive process. Probabilistic generative models provide a natural way of solving this problem, describing a hypothetical process by which texts are created. Bayesian inference can then be used to infer the latent structure in observed text. This paper provides an example of how probabilistic models describing structure in texts can be estimated using Markov chain Monte Carlo (MCMC), a sophisticated Monte Carlo technique, with the samples produced providing a way to reconstruct the process that generated the corpus. Our system inferred distributions over words that characterized topics discussed in meetings, and variation in the topics under discussion throughout the meetings. This exemplifies how MCMC can be used to perform probabilistic inference in complex generative models even when the training data has the full complexity of unrestricted natural language, as might be encountered in a conversation among a group of people in a meeting.

The methods of Bayesian statistics are coming to be applied more widely within the field of computational linguistics, providing tools for inferring latent structure from linguistic data. Several recent papers have used MCMC to sample from posterior distributions over random variables in probabilistic models of language. These papers have addressed a wide range of different aspects of language structure, and have typically been trained in an unsupervised way using large quantities of naturally occurring text (for example [1]). Reference [2] showed how a probabilistic model called Latent Dirichlet Allocation can be used to analyze texts as probability distributions over a number of topics, each topic being made up of a probability distribution over words. This approach captures the fact that the topic of a document plays a large role in determining the frequency with which different word types occur in it, while allowing for the possibility that a document might be better modeled as a blend of different topics, rather than as a single topic. We follow [3], [4] in using MCMC to sample from the posterior distribution over random variables in a Latent Dirichlet Allocation model, and extend [5] which used this approach to find topical structure in transcripts of meetings.

We used the automatic speech recognition transcript of the ICSI meeting corpus [6] and evaluated our algorithm's segmentation performance on the 25 out of the 75 meetings for which a manual segmentation was provided by [7]. The ICSI corpus was concatenated into one long transcript, and the generative model was then used to infer the locations of the topic boundaries. (Boundaries between meetings were not marked in the training data, so these were treated as boundaries between topics in the same way as the within-meeting topic boundaries.) When the discussion in the meetings switched to a different topic, we could expect the distribution over topics which best modeled the utterances from that point onwards to change significantly. Such changes thus give an indication of the points in the transcript at which the topic of the meeting may have changed, and is one of the cues that was used to infer topic boundaries.

We expect discourse close to topic boundaries to differ from discourse in the middle of topics in more ways than just the vocabulary used; we therefore included various other discourse features in the set of observed variables in the probabilistic model. Each such feature was associated with an utterance (rather than with an individual word). One of these features, *cue phrases*, took an integer value corresponding to the number of a predetermined set of cue phrases occurring at the beginning of an utterance within a window around the target utterance. The set of eleven cue phrases was taken from [7], and they were extracted via correlation with topic boundaries in the same corpus; they include phrases such as "okay", "anyway", and "so". The other five features were all real valued. The *silence* feature corresponded to the proportion of non-speaking time in the window about the current utterance; *speaker overlap* corresponded to the proportion of time in the window in which the speech of different speakers overlapped; *speaker activity* measured changes in which speakers were talking in the current window compared to the previous window; *average* and *median segment length* were the mean and median length of all utterances which were partly or wholly in the current window. We also included the LCSeg lexical cohesion measure [7] as a feature: this is a measure of cohesion between utterance windows based on the presence (or otherwise) of lexical chains (sequences of repeated word stems). Features such as these have been shown to improve segmentation performance in discriminative segmentation approaches [7], [8].

The plan of the paper is as follows. Section II overviews previous approaches to topical segmentation. Section III describes our basic generative model, and then Section IV reports how we estimated the posterior distribution over the variables in this model using Gibbs sampling. Section V describes how discourse features were integrated into the basic model, and Section VI describes a range of different models of the placement of segment boundaries. In Sections VII, VIII and IX we report the results of empirical testing, firstly evaluating the relative effectiveness of the different models of segment boundary placement, next the effectiveness of each feature, then the relative contribution of the text of the transcript and of the discourse features, and finally the effect of the hyperparameters. In Section X we summarize the results, and assess the main contributions of this work.

II Finding Topical Structure in Meetings

A number of previous approaches have been taken to the problem of segmentation of text and speech transcripts. Some of these approaches have been based only on differences in the distribution of words in parts of the text dealing with different topics [9]-[13], while others have focused on features that are indicative of topic boundaries [14]-[17]. Direct quantitative comparisons between these approaches are difficult, due to differences in corpora and evaluation procedures, but generally, the greatest success has been achieved by combining both kinds of cues into a single system [7], [8], [18], [19].

A slightly different approach is to make a generative model that represents the process by which the documents were produced [20], [21]. We can model a meeting transcript as having been generated as a concatenation of segments of text, each on a separate topic. Each segment can be modeled as a series of words, randomly sampled from a probability distribution over word types that defines the current topic. The assignment of words to topics and placement of segment boundaries can then be inferred by inverting this generative model, with the probability that the meeting was generated by a particular set of topics and segment boundaries corresponding to the joint probability of the transcript and the variable settings

defining that exact set of topics and boundaries. Such a model therefore allows both topics and the topical segmentation of a transcript to be learned simultaneously.

Reference [5] used this approach to model meeting transcripts as concatenations of sections of text on different topics, except that Latent Dirichlet Allocation was used to model each topical segment as a probability distribution over topics, with each word token being assigned to one particular topic. By making a direct comparison to the LCSeg system [7], it was possible to show that the system reported in [5] achieved similar performance to previous approaches to the same problem which also did not use discourse features (P_k = 0.32, WD = 0.36 for [7]; P_k = 0.33, WD = 0.35 for [5]). The best overall performance on this task appears to have been obtained by [8] and [22] who used supervised learning, unlike [5] whose system was unsupervised. Reference [22] used SVMs to classify each utterance either as a topic boundary or not, based on the words it contained. Reference [8] achieved a modest improvement in performance by using transductive SVMs to incorporate discourse features into the original system.

Maintaining the goal of unsupervised segmentation of meetings, our work extends [5] by integrating discourse features into the generative model. It also investigated the effect of changing how the length of each topical segment in the meetings was modeled. The results reported in [5] concern the manual transcription of the ICSI corpus, not the ASR transcription we used, and also used a different evaluation procedure (see Section VII), so those P_k and WD results are not directly comparable to ours. We focus on comparing the relative performance of extensions to our model, and then perform a separate evaluation to make a direct comparison to the best result reported in [8]. Our primary goal is to show how probabilistic models can be used to integrate multiple sources of information about text structure, an approach that we believe has the potential to be applied to a wide range of problems in computational linguistics.

III A Generative Model of Multi-topic Discourse

The generative model of meeting transcripts is summarized in Fig. 1, which shows which variables are dependent on which other variables. The corpus is a list of U utterances, and the uth utterance contains N_u

words. The total number of word types is W and $\mathbf{w}$ is the list of all utterances in the corpus in order. $\boldsymbol{w}_u$ is the words contained in the uth utterance. $w_{u,i}$ is the ith word in $\boldsymbol{w}_u$. Each utterance is modeled with a probability distribution over T topics, indicating how likely it is that the words in the utterance belong to each of those topics. (We should note that these topics are quite distinct from the topics that form the topical segments of the meetings.) Reference [5] investigated the effect of using 2, 5, 10 and 20 topics on the overall segmentation performance, and concluded that performance was hardly affected by the number of topics. Therefore we have followed Purver et al by using 10 topics throughout this paper.

A variable $\mathbf{c}$ records whether or not a new topical segment begins at each utterance, and so whether the distribution over topics is different to that for the previous utterance. If the value of this variable for the uth utterance c_u is 0, then the distribution over topics for this utterance ($\theta^{(u)}$) is the same as for the previous utterance ($\theta^{(u-1)}$), which is the case illustrated in the diagram. However, when c_u is 1, a completely new distribution over topics is sampled for the current utterance (and for all the following utterances up to the next utterance for which c_u is 1). The first utterance must start a topical segment, so c_1 is always 1. However, normally there will be many utterances in each topical segment, so for most utterances we would expect $\mathbf{c}$ to be zero. We let $P(c_u = 1) = \pi$, so that π effectively specifies the expected number of segments. Later a variety of methods are introduced for controlling the value of π, including setting it to a fixed value, but initially its value is sampled from a symmetric Beta distribution with parameter γ. The extension of the model to incorporate discourse features (f_u in Fig. 1) is introduced in Section V.

The distribution over topics $\theta^{(u)}$ specifies the probability that each word token in the utterance belongs to each of the T topics, the probability of it belonging to topic t being $\theta_t^{(u)}$. Whenever a new topical segment begins ($c_u = 1$), and so the distribution over topics changes, a new value of $\theta^{(u)}$ is sampled from a symmetric Dirichlet distribution with parameter α, and so

$$P(\theta^{(u)}) = \frac{\Gamma(T\alpha)}{\Gamma(\alpha)^T} \prod_{t=1}^{T} (\theta_t^{(u)})^{\alpha-1} \quad (1)$$

where $\Gamma(\cdot)$ is the generalized factorial function.

A variable **z** records the topic to which each word token in the corpus U is assigned, the topic assignment of the ith word token in the uth utterance being written $z_{u,i}$. Each topic T_j specifies a multinomial distribution over word types which is written $\phi^{(j)}$, and in which the probability of each word type is $\phi_w^{(j)}$. Like with the distributions over topics $\theta^{(u)}$, each distribution over word types in a topic is sampled from a symmetric Dirichlet distribution, this time with parameter β, so

$$P(\phi^{(j)}) = \frac{\Gamma(W\beta)}{\Gamma(\beta)^W} \prod_{w=1}^{W} (\phi_w^{(j)})^{\beta-1} \quad (2)$$

The model under which a whole meeting transcript is generated first requires setting **c** to specify at which utterances new topical segments begin. A distribution over topics ($\theta^{(u)}$) is then sampled for each such utterance, and copied to each other utterance up to the start of the next segment. A distribution over word types $\phi^{(j)}$ is also sampled for each topic j. The word tokens of the transcript are then generated for each utterance by first sampling a topic for each word from the distribution over topics for the utterance, and then sampling a word type from the distribution over word types for that topic. This process generates a complete corpus, a set of topics defined in terms of the probability of each word type occurring in each topic, a distribution over topics for each topical segment, and a segmentation of the corpus into topical segments. The hyperparameters of the model (α, β and γ) were all fixed at 0.01, consistent with the values used in previous models of this kind [3], [5].

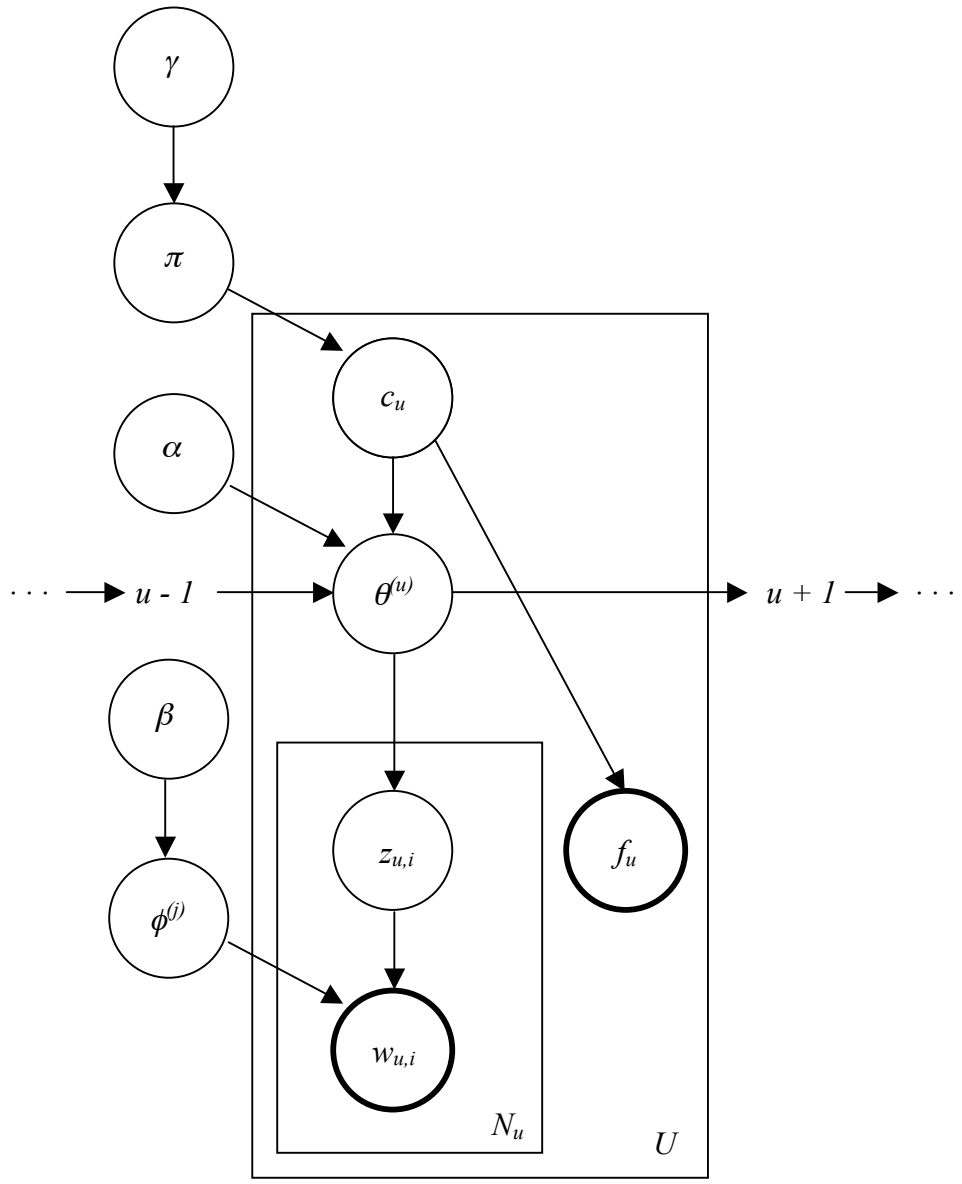

Fig. 1. The Probabilistic Generative Model showing a Single Discourse Feature. Nodes in the graph correspond to random variables, and links indicate dependency structures. Heavy borders on a node indicate observed variables. Boxes around sets of nodes indicate replication of a structure the number of times shown in the corner of the box. Variable names and descriptions appear in the text.

IV Gibbs Sampling of Model Variables

The above has specified a probabilistic Bayesian model for multi-topic meeting transcripts, but does not provide any way to determine which settings of the model's variables have the highest posterior probability with respect to the particular meeting transcripts that we wish to analyze. We would generally expect that those variable settings would be the ones that most closely reflected the process by which the data was generated, and they should therefore give the best estimate of the points in the meetings at which the discourse changed to a different topic. With complex high-dimensional models, like the one used here, one common scheme for finding settings of random variables that account well for the data is MCMC, which allows variables to be sampled from the distribution induced by inverting the generative model [23]. While MCMC does not allow the settings of variables that have the maximum a posteriori probability to be determined directly, if a large number of such samples are produced, they can be averaged to produce an estimate of the posterior distribution over settings of each variable. Most importantly in the case of the present model, if many samples indicate that a topical segment boundary should be placed at a particular location, then it is much more likely that there is a change of topic at that point than if only a few samples indicate that a boundary should be placed there.

The basic idea behind MCMC is to define a Markov chain with a stationary distribution that corresponds to the distribution from which we wish to generate samples. After a large number of iterations, the Markov chain will converge to this stationary distribution, and samples from the Markov chain will behave similarly to samples from the target distribution. The particular flavor of MCMC used here was Gibbs sampling. In Gibbs sampling the underlying Markov chain is defined to be the result of sampling a new value for each variable in the model conditioned on the current settings of all the other variables. This is done in turn for each random variable, and this procedure is repeated over a large number of iterations, until it is believed that enough changes have been made that the current variable settings are independent of their settings before the current series of iterations was begun (i.e. the Markov chain has converged). The settings of the variables at this point are recorded, and kept as one sample, and the

process is then repeated. In some cases it is possible to show how many samples need to be taken in order for MCMC to produce a reliable estimate of the posterior distribution over values of the random variables, but in the case of complex models, such as the one used here, this is not possible. However, our empirical investigations did not find a significant improvement when the number of samples collected was increased beyond 200 (obtained from a single run at intervals of 1000 iterations), and so all results reported here were obtained in this condition.

In order to apply Gibbs sampling to the present model, we need to sample over the discrete variables, namely each individual value of **c** and **z**, each time taking into consideration the settings of all the other values of these variables, and the words in the meeting transcripts **w**. (The continuous random variables π, θ and ϕ can be integrated out, and so there is no need to sample over alternative values for these variables.) We therefore need to obtain an expression that allows us to calculate values for $P(z_{u,i} \mid \mathbf{z}_{-(u,i)}, \mathbf{c}, \mathbf{w})$ and $P(c_u \mid \mathbf{c}_{-u}, \mathbf{z}, \mathbf{w})$, so that we can sample new settings for $z_{u,i}$ and c_u. ($\mathbf{z}_{-(u,i)}$ means the settings of all the values of **z** except the one for the *i*th word in the *u*th utterance, and $\mathbf{c}_{-u}$ means the settings of all the values of **c** except the one for the *u*th utterance.) Reference [5] showed that

$$P(z_{u,i} \mid \mathbf{z}_{-(u,i)}, \mathbf{c}, \mathbf{w}) \propto \frac{n_{w_{u,i}}^{(t)} + \beta}{n_{\cdot}^{(t)} + W\beta} \frac{n_{z_{u,i}}^{(\mathbf{S}_u)} + \alpha}{n_{\cdot}^{(\mathbf{S}_u)} + T\alpha} \tag{3}$$

where $\boldsymbol{S}_u$ is the set of utterances sharing the same topic distribution as u, t is the topic to which $z_{u,i}$ makes an assignment, and the n terms all represent counts. $n_{w_{u,i}}^{(t)}$ is the number of times word type $w_{u,i}$ is assigned to topic t in **z**, $n_{\cdot}^{(t)}$ is the total number of words assigned to topic t in **z**, $n_{z_{u,i}}^{(S_u)}$ is the number of times the topic that is assigned by $z_{u,i}$ is used in $\boldsymbol{S}_{u.}$ and $n_{\cdot}^{(S_u)}$ is the total number of topic assignments in $\boldsymbol{S}_u$ (which is equal to the number of words). These counts exclude the topic assignment under consideration, $z_{u,i}$.

Purver et al also showed that when $c_u = 0$

$$P(c_u \mid \mathbf{c}_{-u}, \mathbf{z}, \mathbf{w}) \propto \frac{\prod_{t=1}^{T} \Gamma(n_t^{(\mathbf{S}_u^0)} + \alpha)}{\Gamma(n_{\cdot}^{(\mathbf{S}_u^0)} + T\alpha)} \frac{n_0 + \gamma}{N + 2\gamma} \tag{4}$$

and when $c_u = 1$

$$P(c_u \mid \mathbf{c}_{-u}, \mathbf{z}, \mathbf{w}) \propto \frac{\Gamma(T\alpha)}{\Gamma(\alpha)^T} \frac{\prod_{t=1}^{T} \Gamma(n_t^{(\mathbf{S}_{u-1}^1)} + \alpha)}{\Gamma(n_{\cdot}^{(\mathbf{S}_{u-1}^1)}) + T\alpha} \frac{\prod_{t=1}^{T} \Gamma(n_t^{(\mathbf{S}_u^1)} + \alpha)}{\Gamma(n_{\cdot}^{(\mathbf{S}_u^1)}) + T\alpha} \frac{n_1 + \gamma}{N + 2\gamma} \tag{5}$$

where n_0 is the number of utterances for which $c_u = 0$ and n_1 is the number of utterances for which $c_u = 1$. N is the total number of word tokens in the corpus. The length of the segment containing c_u will vary, depending on whether c_u is 0 or 1, as c_u defines whether or not a new segment begins at utterance u. If $c_u = 1$, then a new segment $\mathbf{S}_u^1$ will start at the present utterance, and the previous segment will be $\mathbf{S}_{u-1}^1$. If $c_u = 0$, then the utterances in $\mathbf{S}_{u-1}^1$ and $\mathbf{S}_u^1$ will both merge into a single larger segment $\mathbf{S}_u^0$. The above counts exclude c_u itself, and the utterances in either $\mathbf{S}_u^0$ or both $\mathbf{S}_{u-1}^1$ and $\mathbf{S}_u^1$ as appropriate. $n_t^{(\mathbf{S})}$ is the number of times topic t is used in segment S, and $n_{\cdot}^{(\mathbf{S})}$ is the total number of topic assignments in S, where S can be any of $\mathbf{S}_u^0$, $\mathbf{S}_{u-1}^1$ or $\mathbf{S}_u^1$. This completes the specification of the basic model, and of how samples were obtained. The rest of this paper concerns itself with extensions to this model, and evaluations of the effectiveness of each extension in comparison to this baseline.

V Modeling Discourse Features

As noted above, properties of the discourse other than the transcript itself might give clues about where there was a change of topic. Discourse features were incorporated into the Bayesian generative model by introducing a new variable f_u indicating the features of utterance u, which depends only on the segment marker c_u. Features will be useful in segmentation only if their values at utterances where new segments begin have a different distribution to their values at other utterances. We can therefore model the set of all values of a feature **f** with two distributions, one of which describes all those feature values for which c_u =

0 and another all those feature values for which $c_u = 1$. Generatively speaking, we generate a feature value for each utterance using whichever of these distributions is appropriate given the value of c_u for that utterance. Each feature is treated independently of all other features, so it is unproblematic to add any number of features, even though the following discussion only makes reference to a single feature.

The approach used here does not introduce any further discrete latent variables into the generative model, so when sampling we still only need to sample over individual values of **z** and **c**, but we must now take **f** into account as well as **w**. However, as word topic assignments **z** are not dependent on **f** given **c** (which can be seen from the structure of the graphical model in Fig. 1), $P(z_{u,i} \mid \mathbf{z}_{-(u,i)}, \mathbf{c}, \mathbf{w}, \mathbf{f})$ is equal to $P(z_{u,i} \mid \mathbf{z}_{-(u,i)}, \mathbf{c}, \mathbf{w})$, and so we need make no change to the equation obtained by [5] for sampling over **z**.

We do, however, need to obtain new equations that we can use to determine $P(c_u \mid \mathbf{c}_{-u}, \mathbf{z}, \mathbf{w}, \mathbf{f})$, so allowing us to sample values for **c** taking into account the dependence of the features on **c**. We can begin by noting that, as (**c**, **z**, **w**, **f**) is simply ($\mathbf{c}_{-u}$, **z**, **w**, **f**) and c_u together.

$$P(c_u \mid \mathbf{c}_{-u}, \mathbf{z}, \mathbf{w}, \mathbf{f}) = \frac{P(\mathbf{c}, \mathbf{z}, \mathbf{w}, \mathbf{f})}{\mathrm{P}(\mathbf{c}_{-u}, \mathbf{z}, \mathbf{w}, \mathbf{f})} \tag{6}$$

Using the chain rule, and noting that the word topic assignments are independent of **c** for a given **z**, and that the probability of **f** depends only on **c**, we can obtain

$$P(c_u \mid \mathbf{c}_{-u}, \mathbf{z}, \mathbf{w}, \mathbf{f}) = \frac{P(\mathbf{c})P(\mathbf{z} \mid \mathbf{c})}{\mathrm{P}(\mathbf{c}_{-u})P(\mathbf{z} \mid \mathbf{c}_{-u})} \frac{P(\mathbf{f} \mid \mathbf{c})}{P(\mathbf{f} \mid \mathbf{c}_{-u})} \tag{7}$$

The first two terms on the right hand side of equation (7) are unchanged from the previous version of the system (see [5]). Hence we can incorporate utterance features into the existing sampling procedure simply by multiplying each of the values obtained by the original equations, (4) and (5), by the third term, reflecting the probability of the features given this segmentation. (Where there are multiple discourse features, we need to multiply in this term for each one in turn.) However, as $P(\mathbf{f} \mid \mathbf{c}_{-u})$ is constant over

alternative values of c_u and relative rather than absolute probabilities are sufficient for Gibbs sampling, when sampling a value for c_u we need only compare the value of $P(\mathbf{f} \mid \mathbf{c})$ when $c_u = 0$ and when $c_u = 1$.

A Real Valued Features

Real valued features were modeled under the assumption that both those values occurring at segment boundaries and those values occurring elsewhere are normally distributed, but the mean and variance of the distribution in each case is unknown. Following [24], we model the distribution over the variance with an inverse chi-square distribution, and the distribution over values of the means with a normal distribution. Reference [24] gives a probability density function for feature values, which in the case of discourse features f_u for which $c_u = 0$ is

$$P(f_u \mid c_u, \boldsymbol{f}_{-u}, \boldsymbol{c}_{-u}) \sim t_{a0}\,(\mu_0,\ \sigma_0^2(1 + 1/\lambda_0)) \tag{8}$$

where

$$\lambda_0 = \lambda + n_0 \tag{9}$$

$$a_0 = a + n_0 \tag{10}$$

$$\mu_0 = \frac{\lambda\mu + n_0\bar{f}_0}{\lambda + n_0} \tag{11}$$

$$\sigma_0^{\ 2} = \frac{a\sigma^2 + (n_0 - 1)s_0 + \dfrac{\lambda n_0}{\lambda + n_0}(\mu - \bar{f}_0)^2}{a + n_0} \tag{12}$$

and n_0 is the number of features for which $c = 0$, s_0 is the variance of the sample for those features for which $c = 0$, $\bar{f}_0$ is the mean value of features for which $c = 0$ and σ^2, a, λ and μ are parameters of the prior distribution. More specifically these parameters are the mean prior variance σ^2, the confidence in that prior variance a, the prior mean μ, and the confidence in the prior mean λ. It was not expected that

the exact values of these parameters would have a significant effect on the results so σ^2, a, and λ were simply set equal to 1, and μ to 0.

The function t in equation (8) is Student's t, and when this is expanded we obtain

$$P(f_u \mid c_u, \mathbf{f}_{-u}, \mathbf{c}_{-u}) \sim \frac{\Gamma\left(\frac{(a_0+1)}{2}\right)}{\Gamma\left(\frac{a_0}{2}\right)\sqrt{a_0\pi}\,\sigma_0\left(1+\frac{1}{\lambda_0}\right)}\left(1+\frac{1}{a_0}\left(\frac{f-\mu_0}{\sigma_0\left(1+\frac{1}{\lambda_0}\right)}\right)^2\right)^{-\frac{a_0+1}{2}} \tag{13}$$

In this equation π is the mathematical constant, and is hence distinct from the variable π referenced elsewhere in this paper. The corresponding equation for features for which $c_u = 1$ will be exactly the same, except that those symbols with subscript 0 will be replaced by ones with subscript 1. These equations can be used to derive a probability for the whole set of features as follows

$$P(\mathbf{f} \mid \mathbf{c}) = P(f_u \mid c_u, \mathbf{f}_{-u}, \mathbf{c}_{-u}) P(\mathbf{f}_{-u} \mid \mathbf{c}_{-u}) \tag{14}$$

However, we in fact need only to calculate the relative probability of $P(\mathbf{f} \mid \mathbf{c})$ for the two alternative settings of c_u. As $P(\mathbf{f}_{-u} \mid \mathbf{c}_{-u})$ is not affected by the value of c_u it can be ignored during sampling and we need only calculate probabilities for $P(f_u \mid c_u, \mathbf{f}_{-u}, \mathbf{c}_{-u})$ for $c_u = 0$ and $c_u = 1$.

B Count Features

A somewhat different approach was needed to model the *cue phrases* discourse feature, as this feature is a count, rather than a real value. However, as before, if this feature is to be useful in segmenting documents, it must have a different distribution at the start of segments to elsewhere. Therefore count features were also modeled using two probability distributions, one for those feature values for which $c = 0$ and another separate distribution for those feature values for which $c = 1$. The distributions used were Poisson distributions, and a Gamma distribution with its shape parameter q fixed at 2 and its inverse scale

parameter r fixed at 1 was used as a prior over the rate parameter of each Poisson distribution (see [25] for details). The probability of feature f_u when $c_u = 0$, given all the other features and **c**, is therefore given by

$$P(f_u \mid \mathbf{f}_{-u}, \mathbf{c}) = \frac{r^q}{\Gamma(q) f_u!} \frac{\Gamma(t_0 + n_0(q-1)+1)((n_0-1)(r+1))^{t_0 - f_u + (n_0-1)(q-1)+1}}{\Gamma(t_0 - f_u + (n_0-1)(q-1)+1)(n_0(r+1))^{t_0 + n_0(q-1)+1}} \tag{15}$$

where the sum of all features for which $c_u = 0$ is t_0. A parallel equation can be obtained for the case in which $c_u = 1$, by replacing t_0 with t_1 and n_0 with n_1. When sampling, we need only calculate $P(f_u \mid \mathbf{f}_{-u})$ when $c_u = 0$ and when $c_u = 1$, and multiply these values in to the probabilities obtained for c_u based on the other features and the words. There is, however, a special case when there is only one feature for which $c = 0$ (or for which $c = 1$). In this case we cannot obtain $P(f_u \mid \mathbf{f}_{-u}, \mathbf{c})$ by dividing $P(\mathbf{f} \mid \mathbf{c})$ by $P(\mathbf{f}_{-u} \mid \mathbf{c})$, because there are no features other than u, and so the set of features $\mathbf{f}_{-u}$ is empty. In this case f_u is the whole set of features **f**, and so

$$P(f_u \mid \mathbf{c}) = P(\mathbf{f} \mid \mathbf{c}) = \frac{r^q}{\Gamma(q) f_u!} \frac{\Gamma(t_0 + q)}{(r+1)^{t_0+q}} \tag{16}$$

VI Alternative Models of Segmentation

In the original generative model, the probability of a new segment starting at any point, $P(c_u = 1)$, is determined by a variable π giving this probability directly. (Although there is a Beta distribution over possible values of this variable, so the variable itself is integrated out. This approach is henceforth referred to as VariPi.) However, this approach does not seem to model segment lengths very well, as it forces the prior distribution over segment lengths to be geometric, which, as can be seen in the results below, produces oversegmentation when features are used. One alternative approach would be to make the probability of a topic break at u conditional on the value of c_{u-1}, but as the inferred segmentation would normally contain occasional topic breaks separated by many utterances for which $c_u = 0$, we would expect to gain little if any advantage from this approach. However, we considered a variety of other

approaches to modeling **c** by replacing π with a range of different functions. If these new priors improve performance it could either be because they more accurately reflect the behavior of participants in meetings, or it could simply be that they direct the model towards the latent structure in the meetings that corresponds to the gold standard, and away from any other types of latent structure that may also be present. Here we do not attempt to distinguish between these two possibilities, but simply consider any prior that produces a closer match to the gold standard to be better model of **c** for our present purpose of inferring the topical structure of the meetings.

A *Fixed* π

The simplest solution to controlling the number of segments is simply to fix the value of π. As π is the prior probability of a segment starting at each utterance, this effectively defines the expected number of segments, and so π was fixed at 0.0755 to reflect the number of segments marked in the gold standard (179 of 23,703 utterances for which a segmentation was provided were the start of new segments). However, if the transcript were best described by a generative model containing a different number of segments, we would expect the final number of segments to be a compromise between that number of segments and the expected number as defined by π. Under this solution, the prior distribution over segment lengths remained geometric.

In the equations which are used during the sampling procedure, the prior probability for alternative segmentations shows up only as the rightmost term in equations (4) and (5). Therefore, if we change the model of where new segments begin, we need only replace these terms. It is straightforward to show that if we fix the value of π then we need only replace the term in equation (4) with $(1-\pi)$ and the one in equation (5) with π. This reflects the fact that equation (5) describes a situation in which there is one more segment than the situation described by equation (4), which instead has one more utterance which is not the start of a new segment, and hence we multiply in either the *a priori* probability of a new segment starting or not starting at utterance u.

B A Poisson Distribution over Segment Lengths

An alternative approach is to model the length of each segment (how many utterances it contains) with a Poisson distribution. This allowed a distribution over the length of segments to be used, making it easier to eliminate very short segments. A gamma distribution with the same hyperparameters (q and r) as those used for the count features was used as a prior over the rate parameter of this Poisson distribution.

When sampling over a particular value of c_u, we need only compare the probability of the two segments that would result if the value of c_u was 1 (those being the segment ending just before u, and the segment beginning at u) with the probability of the single segment that would result if c_u were 0, and the two segments merged into one big one. **c** implicitly defines the lengths of all the segments, and it is easier to discuss this prior in terms of segment lengths than in terms of segment boundary locations, so we will use **s** to denote the set of lengths of all the segments, s_1 to denote the length (in utterances) of the segment that would end just before the u if c_u were 1, and s_2 to denote the length of the segment that would begin at the current utterance were c_u 1. The length of the segment that would take the place of these two shorter segments if c_u were 0 would be equal to $s_1 + s_2$. We should note that any setting of **s** can alternatively be expressed in terms of a setting of **c**.

The total length of all segments (in utterances) will be equal to the number of utterances in the corpus U. The number of segments in the whole corpus if c_u is 0 will be written $d_{c=0}$ and the number of segments if c_u is 1 will be written $d_{c=1}$. We can note that

$$d_{c=0} = d_{c=1} - 1 \tag{17}$$

By the chain rule. the probability of the single large segment, given all the other segments is

$$P(c_u = 0 \mid \mathbf{c}_{-u}) = P((s_1 + s_2) \mid \mathbf{s}_{-(s_1+s_2)}) = \frac{P((s_1 + s_2), \mathbf{s}_{-(s_1+s_2)})}{P(\mathbf{s}_{-(s_1+s_2)})} \tag{18}$$

where (s_1+s_2) denotes the segment in question, and $\mathbf{s}_{-(s_1+s_2)}$ denotes the set of all segments except (s_1+s_2). Therefore

$$P(c_u = 0 \mid \mathbf{c}_{-u}) = P((s_1 + s_2) \mid \mathbf{s}_{-(s_1+s_2)}) = \frac{r^q}{\Gamma(q)} \frac{\Gamma(U + d_{c=0}(q-1)+1)((d_{c=0}-1)(r+1))^{U+(d_{c=0}-1)(q-1)+1}}{\Gamma(U - s_1 - s_2 + (d_{c=0}-1)(q-1)+1)(d_{c=0}(r+1))^{U+d_{c=0}(q-1)+1}(s_1+s_2)!} \tag{19}$$

The probability of the two smaller segments given all the other segments is

$$P(c_u = 1 \mid \mathbf{c}_{-u}) = P(s_1, s_2 \mid \mathbf{s}_{-s_1,s_2}) = \frac{P(s_1, s_2, \mathbf{s}_{-s_1,s_2})}{P(\mathbf{s}_{-s_1,s_2})} \tag{20}$$

$$= \left(\frac{r^q}{\Gamma(q)}\right)^2 \frac{\Gamma(U + d_{c=1}(q-1)+1)((d_{c=1}-2)(r+1))^{U+(d_{c=1}-2)(q-1)+1}}{\Gamma(U - s_1 - s_2 + (d_{c=1}-2)(q-1)+1)(d_{c=1}(r+1))^{U+d_{c=1}(q-1)+1} s_1! s_2!} \tag{21}$$

Like with the fixed π model, equation (19) can be used to replace the rightmost term in (4) and equation (21) the rightmost term in (5).

C Sampling a New Value of π *for each Segment*

An alternative approach would be to keep π, but to sample a new value of this variable independently for each new segment. The actual value of π in each case would be integrated out. The key effect of this change would be that the length of one segment would no longer be so directly dependent on the length of the other segments, as lower values of π would come to dominate the posterior for longer segments, and higher ones for shorter segments. The distribution over segment lengths would therefore no longer be geometric.

Under this approach, the probability of a complete segmentation of the corpus **c** would depend not only on the number of utterances for which c = 0 and for which c = 1, but also on the order in which the utterances for which c = 0 and c = 1 occur. We must therefore treat each segment separately, and then

obtain the joint probability of **c** for the whole corpus by multiplying together the probability of the segmentation of each individual segment. As a segment always starts with an utterance for which $c = 1$, and all other utterances it contains have $c = 0$, we can write

$$P(\mathbf{c}) = \prod_{s_i \in \mathbf{s}} \int_0^1 \pi(1-\pi)^{s_i - 1} P(\pi) d\pi \tag{22}$$

where **s** is the set of all segments, and s_i is the length of each particular segment in **s**. If we again use a symmetric beta distribution over π then we obtain

$$P(\mathbf{c}) = \prod_{s_i \in \mathbf{s}} \int_0^1 \pi(1-\pi)^{s_i - 1} \frac{\Gamma(2\gamma)}{\Gamma(\gamma)^2} \pi^{\gamma-1}(1-\pi)^{\gamma-1} d\pi \tag{23}$$

$$= \left(\frac{\Gamma(2\gamma)\Gamma(\gamma+1)}{\Gamma(\gamma)^2} \right)^{n_1} \prod_{s_i \in \mathbf{s}} \frac{\Gamma(s_i + \gamma - 1)}{\Gamma(s_i + 2\gamma)} \tag{24}$$

where n_1 is the number of utterances for which $c = 1$.

We can obtain the conditional probability $P(c_u \mid \mathbf{c}_{-u})$ by dividing $P(\mathbf{c})$ by $P(\mathbf{c}_{-u})$. (Note, however, that we cannot obtain $P(\mathbf{c}_{-u})$ simply by applying the equation for the probability of **c** but omitting c_u, as was the case with the other approaches to modeling segment length. This is because the probability of **c** in this case depends not just on the number of utterances for which $c = 1$, but also on their order.) When $c_u = 1$, we create two short segments of lengths s_1 and s_2 in place of the long segment of length $(s_1 + s_2)$ that would result were $c_u = 0$. $P(\mathbf{c}_{-u})$ will be constant over the two alternative values of c_u. For different values of c_u however, the value of n_1 and **s** is different. n_1 will be one greater when $c_u = 1$ than when $c_u = 0$. Changing c_u from 0 to 1 will split one larger segment into two smaller ones, so **s** will differ as to whether it contains the one large or two small segments. However, all other members of **s** will be identical for $c_u = 0$ and $c_u = 1$, and so can be replaced by a constant h, as can any other terms that do not depend on the value of c_u. We therefore obtain one equation for when $c_u = 1$

$$P(c_u = 1 \mid \mathbf{c}_{-u}) = h \frac{\Gamma(2\gamma)\Gamma(\gamma+1)}{\Gamma(\gamma)^2} \frac{\Gamma(s_1+\gamma-1)}{\Gamma(s_1+2\gamma)} \frac{\Gamma(s_2+\gamma-1)}{\Gamma(s_2+2\gamma)} \tag{25}$$

and another for when $c_u = 0$

$$P(c_u = 0 \mid \mathbf{c}_{-u}) = h \frac{\Gamma(s_1+s_2+\gamma-1)}{\Gamma(s_1+s_2+2\gamma)} \tag{26}$$

D A Distribution over the Number of Segments

The results below show that with most of the above approaches, there was a tendency for too many segments to be produced. In order to allow the number of segments to be controlled more closely, a new model of segmentation that specified a distribution over the number of segments in the meetings was introduced. This distribution (henceforth referred to as DNS) had a sharp peak when the number of segments was close to the number in the gold standard, and much lower probabilities for values considerably different to this number. This clearly makes the inference problem easier, as the system now no longer has to infer an appropriate number of segments, but it allowed us to determine whether discrepancies with the gold standard were mainly caused by the incorrect placement of segment boundaries, or by a tendency to segment at different level of granularity. In an application of the system the exact number of segments would be unknown in advance, but this parameter could be used to determine how fine-grained a segmentation of the corpus should be obtained. It was however still necessary to use a distribution over the number of segments, as if the number of segments had been completely fixed then no segments could have been added or removed during sampling. It might appear that the same effect could have been achieved with the other models of segmentation by setting the hyperparameters to appropriate values, but there is no easy way to determine at what value to set the hyperparameters in order to produce the desired number of segments, and any one setting might result in quite different numbers of segments in different samples.

The distribution over the number of segments, d is given in (27), and has an integer parameter Ω, which corresponds to the peak of the distribution, and which was set at 603 to reflect the number of segments in the gold standard. Two other parameters Δ and Ξ control the shape of the distribution, and were set at 1.1 and 3 respectively, so creating a strong preference for settings of **c** for which the number of segments was close to that in the gold standard.

$$P(d) = \frac{c}{\Delta^{\left(|d-\Omega|^{\Xi}\right)}} \tag{27}$$

c is a normalizing constant that ensures that the total for all possible values of d is one, and so that this is a valid probability distribution. (There can only be a finite number of values of d, as it must always be between one and the total number of utterances.)

In terms of the generative model, under this approach first a number of segments is sampled, and then utterances in the transcript are sampled uniformly (without replacement), and the segment boundaries are placed at those utterances. The existing parts of the generative model then fill in the topic and word assignments, and generate the features. Implementation of this approach is very straightforward, as we simply need to calculate a the probability of d when $c_u = 0$ and when $c_u = 1$, and multiply these values into the probabilities for $P(c_u \mid \mathbf{c}_{-u}, \mathbf{z}, \mathbf{w})$ when sampling a new value of c_u.

E Combining Multiple Segmentation Strategies

While the DNS approach allows the number of segments to be controlled very closely, it returns to using a geometric distribution over the lengths of segments, losing the benefits that were gained by the introduction of a Poisson distribution over segment lengths and the NewPi approach. However, it is quite straightforward to combine these approaches with the DNS model. We can simply modify the generative model so that we first sample a number of segments, and then generate a transcript as before, but this time we stop when the sampled number of segments has been generated. Implementation of these approaches simply required multiplying in the prior for the number of segments when $c_u = 0$ and when $c_u = 1$ to the

existing equations for calculating $P(c_u \mid \mathbf{c}_{-u}, \mathbf{z}, \mathbf{w})$ when using a Poisson distribution, or when sampling a new value of π for each segment.

VII Evaluating Segmentation

We evaluated the performance of each segmentation model presented above both with and without features, and the results are shown in Table I. The DNS model performed best overall, although when features were used both the Poisson and the combined NewPi-DNS model were equally good. Boundaries were marked on 0.8% of utterances in the gold standard, so FixedPi greatly undersegmented without features, while NewPi greatly oversegmented. VariPi and Poisson produced approximately the correct number of segments. (The percentages are averages across all samples.) All these strategies oversegmented when features were added, but in all conditions DNS was successful in producing almost exactly the correct number of segments, both when used by itself and when combined with another strategy.

Table I. Segmentation Performance

Segmentation Strategy	No features			All features		
	% boundaries	P_k	WD	% boundaries	P_k	WD
VariPi	1.1	0.34	0.41	2.6	0.32	0.47
FixedPi	0.07	0.33	0.37	2.5	0.33	0.48
Poisson	0.6	0.40	0.48	2.4	0.28	0.36
NewPi	2.7	0.35	0.40	2.6	0.31	0.45
DNS	0.8	0.32	0.39	0.8	0.28	0.36
Poisson-DNS	0.8	0.40	0.48	0.8	0.34	0.41
NewPi-DNS	0.8	0.34	0.36	0.8	0.28	0.36

We evaluated how closely the inferred segmentations matched the gold standard using the P_k [19] and WD [26] metrics. Both metrics use a window that is moved over the text, and is of length equal to half the mean length of the segments marked in the gold standard. We measured segment length, and hence window size, in terms of number of utterances, unlike [5] who used actual times. P_k records the proportion of times that the gold standard and the learned segmentation disagree about whether there is a topical boundary between the utterance at the start of the window and the utterance at the end of the window. WD records the proportion of times that the number of segment boundaries placed between these two utterances is different for the learned segmentation and for the gold standard. For both measures lower values indicate better performance. In order to produce a single segmentation, we averaged the samples for **c** and thresholded these at whatever value produced the number of segments closest to that in the gold standard. Unlike [5] we did not perform smoothing, as we did not find that this produced any improvement in performance.

The DNS model clearly outperformed all other models on the P_k measure without features. FixedPi and NewPi-DNS resulted in better WD scores without features, but worse P_k ones, but the advantage for FixedPi can be attributed to undersegmentation. (It is a weakness of both P_k and WD that when the boundaries in the gold standard are unevenly spaced, good scores can be obtained by placing very few boundaries. This is because the segmentation will then be correct for those parts of the gold standard that contain no boundaries.) The Poisson model only produced good results with features and without DNS; without features and when combined with DNS its performance was quite poor. The other models did not perform well with features, in some cases performing worse (probably due to oversegmentation) with the exception of the combined NewPi-DNS model. However, this model did not result in a tangible improvement over the DNS model alone, so the DNS model was judged to be preferable on grounds of simplicity.

In the gold standard, there were usually short segments at the beginning and end of each meeting, but the other boundaries tended to be spaced quite unevenly, with there being an average of 7.2 segments per

meeting. Fig. 2 shows the gold standard and several inferred segmentations for a portion of the corpus, covering several meetings. (It is plotted from a single sample for each condition.) We can see that the boundaries that were placed using FixedPi were mainly quite accurate, but many valid boundaries were omitted. The addition of features resulted in more boundaries being placed, but these new boundaries were mainly clustered around those found without features, rather than corresponding to boundaries missed when not using features. VariPi and NewPi (not shown here) were qualitatively similar to FixedPi, but resulted in many more boundaries when no features were used. The Poisson distribution avoided the problem of clustering of boundaries, but it achieved this by producing a fairly even spacing of boundaries, which did not reflect the quite uneven spacing seen in the gold standard. DNS ensured that the number of boundaries was always close to that in the gold standard, but otherwise did not appear to greatly alter the properties of the segmentation (both for the simple DNS shown here, and for Poisson-DNS-and NewPi-DNS).

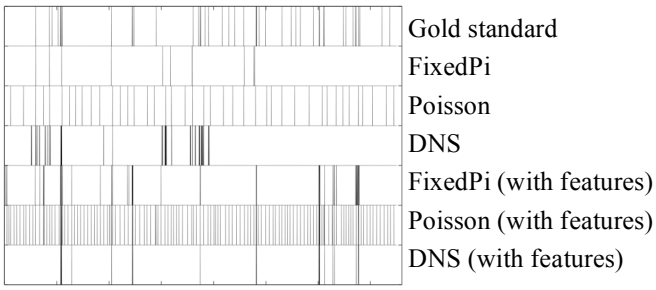

Fig. 2. Segmentations of one part of the corpus

We measured the average distance from each inter-meeting boundary and each within meeting topical boundary in the gold standard to the closest inferred boundary (using the DNS model), in order to determine whether the system was more successful at detecting one type of boundary or the other. The mean distance to inter-meeting boundaries was 0.54 utterances (s.d. 0.78) and to topic boundaries 143.2

utterances (s.d. 153.7), showing that inter-meeting boundaries were detected with very high accuracy, but that there was a great deal of variability in how accurately within meeting topic boundaries were detected.

Fig. 3 shows how the segmentations were obtained at a more fine-grained level of detail. The graph shows the proportion of words assigned to each topic in a series of 40 utterances from a single sample (obtained using DNS, without features). The height of the area assigned to each topic corresponds to the proportion of words in the utterance at that point which were assigned to the topic. (The six topics that were least common in this part of the transcript have been grouped together for clarity, and are shown in white.) The white vertical line marks an inferred topic boundary. The primary cue to segmentation appears to be the increased frequency of the striped topic after the boundary.

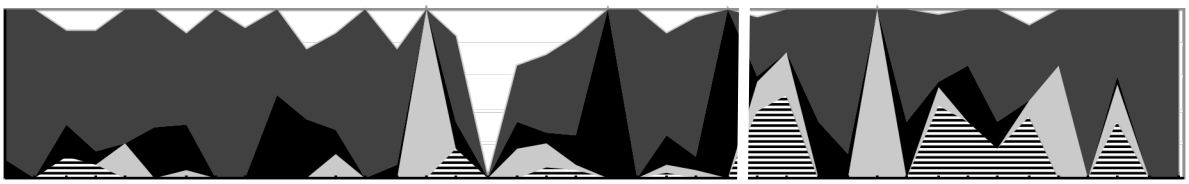

Fig. 3. Change in topic frequencies and an inferred boundary

VIII Evaluating Features

We used the DNS segmentation model to determine the relative contribution of each feature towards overall performance. We compared the results of the model with the full feature set to the results yielded by removing one feature at a time (Fig. 4). We found that three of the seven features not only do not contribute to good performance, but affect performance negatively. These were *speaker overlap*, *average segment length* and *median segment length*. It therefore seems likely that these features indicate structure in the transcript that does not correspond to that marked in the gold standard. By removing all three of the offending features we were able to substantially improve the overall score of the model (P_k = 0.26, WD =

0.33 versus P_k = 0.28, WD = 0.36 with all features). We also found that the most useful feature was *LCSEG cohesion*. Removing this feature dramatically lowered performance (P_k = 0.37, WD = 0.41).

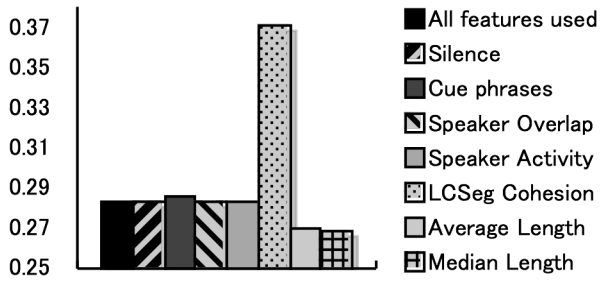

Fig. 4. P_k Values when single features were removed

Next, we investigated the relative contribution of the features on one hand, and the text of the transcript on the other. We compared three sets of results: one set was obtained without any features (P_k = 0.32, WD = 0.39); another set was obtained using all useful features, but not the generative model of the text itself (P_k = 0.27, WD = 0.34); a third set was obtained using both components of the model (P_k = 0.26, WD = 0.33). We can see that the generative model of the text and the features both contribute to the performance of the system.

In order to compare our system's performance to that of [8], we mapped our gold standard and inferred segmentation to be based on words not utterances and then recalculated P_k, as that mirrors the procedure used by [8]. Like us, [8] also concatenated the meetings together and aimed to infer both inter-meeting and within meeting boundaries. We obtained a P_k score of 0.26 using this procedure, against the score of 0.21 for the best version of the system reported in [8]. (Reference [8] did not report *WD* scores.) While we cannot claim to have the best overall performance on this task, we should note that our system is unsupervised, and so has the advantage of being much easier to adapt to other text types for which training data may not be available, for example transcripts of meetings in languages other than English.

IX Evaluating Hyperparameters

We investigated how sensitive this best obtained version of the system was to changes in the hyperparameters of the model. α and β control the sparsity of the distribution over topics within each segment and the distribution over words within each topic respectively, with smaller values favoring sparser distributions (i.e. fewer topics receiving high probability in each segment, and fewer words receiving high probability in each topic). γ controls the strength of the prior on π, with smaller values favoring more extreme values of π (i.e. those close to 0 and 1). q and r together control what frequency of cue phrases is assigned the highest prior probability, and how tightly the probability mass is concentrated around that value. The meaning of the hyper parameters for real valued features was described in Section V A. We performed empirical investigations to determine the effect of each hyperparameter on the overall performance of the system. 20 samples were collected at intervals of 100 iterations in a series of short runs, and in all but the first case we changed the value of one of the hyperparameters to a new value. The results of these runs, shown in Table II, show that the performance of the system is almost constant over a considerable range of values for each hyperparameter, suggesting that the data largely overwhelms any the prior bias due to these parameters.

Table II. Sensitivity to Hyperparameters

Change to hyperparameters	P_k	WD
none	0.26	0.33
$\alpha = 0.1$	0.26	0.34
$\alpha = 0.001$	0.27	0.34
$\beta = 0.1$	0.27	0.34
$\beta = 0.001$	0.26	0.33
$\gamma = 0.1$	0.26	0.33
$\gamma = 0.001$	0.26	0.33
$\sigma^2 = 10$	0.27	0.34
$\sigma^2 = 0.1$	0.26	0.33
$a = 10$	0.27	0.33
$a = 0.1$	0.26	0.33
$\lambda = 10$	0.27	0.34
$\lambda = 0.1$	0.27	0.34
$\mu = 10$	0.27	0.34
$\mu = -10$	0.27	0.34
$q = 20$	0.26	0.33
$q = 1.2$	0.26	0.33
$r = 10$	0.26	0.33
$r = 0.1$	0.26	0.33

X Conclusion

We have shown how probabilistic models can be used to detect topical structure in large text corpora. Empirical testing revealed that integration of discourse features into the model improved performance, as did modeling the number of topical segments in a way that was strongly biased towards the number of segments in the gold standard. More generally, this work demonstrates how probabilistic models allow for the highly principled integration of multiple cues to text structure, and the specification of complex generative processes in which prior biases are made explicit and so can easily be manipulated. The use of MCMC to estimate non-numeric variables is an essential component of this approach, as it enables the technique to be applied even with complex models and large datasets.

Acknowledgement

This work was supported by the CALO project (DARPA grant NBCH-D-03-0010) and Mike Dowman was supported by a JSPS postdoctoral fellowship.

References

[1] S. Goldwater and T. L. Griffiths and M. Johnson, "Contextual dependencies in unsupervised word segmentation", in *Proc. of ACL/COLING,* 2006.

[2] D. Blei, A. Ng, and M. Jordan, "Latent Dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[3] T. L. Griffiths and M. Steyvers, "Finding scientific topics," *Proceedings of the National Academy of Sciences*, vol. 101, pp. 5228-5235, 2004.

[4] T. L. Griffiths, M. Steyvers and J. B. Tenenbaum, "Topics in semantic representation," *Psychological Review*, vol 114, pp. 211-244, 2007.

[5] M. Purver, K. P. Kording, T. L. Griffiths and J. B. Tenenbaum, "Unsupervised topic modeling for multi-party spoken discourse," in Proc. of COLING/ACL, 2006.

[6] A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters, "The ICSI meeting corpus," in *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing,* 2003, pp. 364–367.

[7] M. Galley, K. McKeown, E. Fosler-Lussier, and H. Jing, "Discourse segmentation of multi-party conversation," in *Proc. of ACL*, 2003, pp 562–569.

[8] M. Georgescul, A. Clark and S. Armstrong, “Exploiting structural meeting-specific features for topic segmentation”, in Actes de la 14ème Conférence sur le Traitement Automatique des Langues Naturelles, 2007, pp 15-24.

[9] M. A. Hearst, “Multi-paragraph segmentation of expository text,” in *Proc. ACL*, Los Cruces, NM, 1994.

[10] F. Y. Y. Choi, “Advances in domain independent linear text segmentation,” in *Proc. of NAACL*, Seattle, 2000, pp. 26-33.

[11] M. Kan, J. L. Klavans and K. R. McKeown, “Linear segmentation and segment significance,” in *Proc. of the 6th International Workshop on Very Large Corpora*, Montreal, 1998, pp. 197-205.

[12] M. Dowman, V. Tablan, H. Cunningham and B. Popov, “Web-assisted annotation, semantic indexing and search of television and radio news,” in Proc. of World Wide Web, Chiba, Japan, 2005, pp. 225-234.

[13] I. Malioutov and R. Barzilay, “Minimum cut model for spoken lecture segmentation,” In Proc. of COLING/ACL, 2006.

[14] M. Franz, B. Ramabhadran, T. Ward and M. Picheny, “Automated transcription and topic segmentation of large spoken archives,” in *Proc. of Eurospeech*, Geneva, Switzerland, 2003, pp. 953-956.

[15] P. V. Mulbregt, I. Carp, L. Gillick, S. Lowe and J. Yamron, “Text segmentation and topic tracking on broadcast news via a hidden Markov model approach,” in *Proc. 5th international conference on spoken language processing*, Sydney, Australia, 1998.

[16] A. Kehagias, A. Nicolaou, V. Petridis and P. Fragkou, "Text segmentation by product partition models and dynamic programming," *Mathematical and Computer Modelling*, vol. 39(2-3), pp. 209-217, Jan. 2004.

[17] S. R. Maskey and J. Hirschberg, "Automatic summarization of broadcast news using structural features," in *Eurospeech*, Geneva, Switzerland, *2003*.

[18] L. Chaisorn, T. Chua, C. Koh, Y. Zhao, H. Xu, H. Feng, and Q. Tian, "A two-level multi-modal approach for story segmentation of large news video corpus," *In Proc. of TRECVID Conference*, Washington D.C, 2003.

[19] D. Beeferman, A. Berger, and J. D. Lafferty, "Statistical models for text segmentation," *Machine Learning*," vol. 34(1-3), pp. 177–210, 1999.

[20] T. Imai, R. Schwartz, F. Kubala and L. Nguyen, "Improved topic discrimination of broadcast news using a model of multiple simultaneous topics," in *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1997, pp. 727–730.

[21] R. Barzilay and L. Lee, "Catching the drift: Probabilistic content models, with applications to generation and summarization," in *HLT-NAACL 2004: Proc. of the Main Conference*, 2004, pp. 113–120.

[22] M. Georgescul, A. Clark, and S. Armstrong. "Word distributions for thematic segmentation in a support vector machine approach". In *Proc. of CoNLL*, 2006, pp 101-108.

[23] D. J. C. MacKay, "Introduction to Monte Carlo methods," in *Learning in Graphical Models*, M. I. Jordan, Ed. Cambridge, MA: MIT Press,1999.

[24] J. R. Anderson, "The adaptive nature of human categorization," *Psychological Review*, vol. 98(3), pp. 409-429, 1991

[25] A. Gelman, J. B. Carlin, H. S. Stern and D. B. Rubin, *Bayesian Data Analysis*, Boca Raton, FL: Chapman and Hall, 2004.

[26] L. Pevzner and M. Hearst, "A critique and improvement of an evaluation metric for text segmentation," *Computational Linguistics*, vol. 28(1), pp 19-36, 2002.