The Actor-Topic Model for Extracting Social Networks in Literary Narrative

Asli Celikyilmaz; Dilek Hakkani-Tur
Speech @ Microsoft | Microsoft Research
Mountain View, CA
aslicel@microsoft.com*
dilek@ieee.org

Hua He; Greg Kondrak; Denilson Barbosa
Computer Science Department
University of Alberta
hhe|kondrak|denilson@cs.ualberta.ca

Abstract

We present a generative model for conversational dialogues, namely the actor-topic model (ACTM), that extend the author-topic model (Rosen-Zvi, *et.al*, 2004) to identify actors of given conversation in literary narratives. Thus ACTM assigns each instance of quoted speech to an appropriate character. We model dialogues in a literary text, which take place between two or more actors conversing on different topics, as distributions over topics, which are also mixtures of the term distributions associated with multiple actors. This follows the linguistic intuition that rich contextual information can be useful in understanding dialogues, eventually effecting the social network construction. We propose ACTM to ideally lead our research on social network extraction in literary narratives. Our experiments on nineteenth century English novels indicate that exploiting content structure of dialogues can yield significant improvements over a baseline using language models which is based on local context in constructing social interactions.

1 Introduction

In social network analysis, the patterning of the social connections that link sets of actors is commonly studied. For the most part researchers seek to uncover either or both of the two kinds of patterns: They often look for social groups – collections of actors who are closely linked to one another. Or, they look for social positions – sets of actors who are linked into the total social system in similar ways. Some of the work on social network analysis and construction include information extraction from unstructured text [1, 11], link prediction [6], network construction [5], etc.

Computer-assisted literary analysis is a sub-field of social network research, in which theories center on actors in two specific ways: how many actors novels tend to have, and how these actors interact with each other. Although literary theorists have been developing graphical representations of social connections and other features from literature [8, 9], the analysis of social networks in literature based on *semantic orientations* has been rare due to the complexity of automatic extraction of interactions between actors. In a recent study [2, 3], the text of literary fiction is characterized by extracting the network of social conversations that occur between actors. They identify characters (actors) by assigning a "speaker" (if any) to each instance of quoted speech from among characters and construct a social network by detecting conversations from the set of dialogue acts. Their methods are based on syntactic categorization and supervised multi-class discriminate classification for modeling the quoted speech attribution.

Characterizing the actors in a given conversation setting of a literary text, namely assigning a speaker to each instance of quoted speech, requires large amounts of annotated data. Thus previous work

*This research was conducted when the first author was at University of California, Berkeley and the second author was at the International Computer Science Institute of Berkeley.

obtained gold standard annotations from Amazon's Mechanical Turk's program. Aside from caveats about such data collection, previous work do not include *content-related aspects* of conversations to attribute utterances. In our work we want to uncover patterns from conversations between actors of a corpus that would have otherwise been extremely difficult to discover and to spark off a new approach to social network constructions between actors based on conversation topics.

The research presented in this paper is concerned with *unsupervised* attribution of actors in social networks from literature. To address this problem, we introduce the actor topic model (ACTM) which automatically constructs a network based on dialogue interactions between the characters in a novel without the need for annotated text. A number of methods for building network structure of different domains using latent variable models have been proposed [13, 15] to infer relationship strength. However, to the best of our knowledge, no attempt has been made to identify actors in literary text based on the topics of the conversations or build a network structure based on the topical similarity between actors. Actor attribution to quoted speech based on content-related conversations can improve the performance of machine learning and information retrieval tasks. The proposed method is also useful to automatic generation of network structure in a literary text.

Our aim in this paper is two-fold. First we build a probabilistic model on literary text to extract a mapping between actors and the topics they discuss in conversations over an entire novel. Next, using this information we construct a social network structure given a novel. Our aim is not to label the relationships in social networks but rather automatically identify hidden relationships based on actors conversing on latent topics of dialogues without the use of transcribed data. As a pre-processing step, initially the instances of quoted speech, namely *utterances*, are identified to be attributed to an actor (because our ACTM model is unsupervised, we do not use actor labels to construct the model). Then each dialogue boundary is determined to characterize the dialogue model structure. Later, we extract hidden concepts in dialogues and relate them to actors in an analogical way to the author-topic models [10]. Rather than representing utterances and narrations in a given text as additional hierarchy in the topic model, with ACTM we model them as meta-variables by allowing the mixture-weights for different topics to be determined by the utterance and narratives of a given dialogue. Characterizing dialogue-actor level topics, we are not only able to extract topics from dialogues and map on utterances, but also organize, and cluster actors based on these topics.

In the next section, we describe the methods we use to extract dialogues in a given literary text and the actor-topic model to extract hidden concepts of dialogues. We later describe the methods we use to construct the network structure of the conversations between actors. We present results of character attribution and social network construction on two 19^{th} century novels and derive conclusions.

2 Extracting Dialogues and Actors from Literary Text

In this section, we describe the dialogue based information extraction model. Since we want to build a model that can predict actors conversing in a dialogue, the ***dialogues*** are the basis of the probabilistic model we present in this paper. In literary text, conversations take place between people in sequence, forming overlapping dialogues in a window of text (Fig. 1). A dialogue is defined as segmented structured text, a sequence of sentences, which are either an utterance ***u*** (a quoted text that can be attributed to an actor) or a narration ***n***, which have no actor attribution. We want to build a model that can predict actors conversing in a dialogue.

We segment each chapter in a given novel into dialogues, $\mathcal{D} = \{d_i\}_{i=1}^{D}$, where D is the total number of dialogues in a novel and we present each dialogue $d_i = \{t_1, ...t_u\}$ as sequence of text, where $t_j \in \{\mathbf{u}, \mathbf{n}\}$. We extract dialogues separately from each chapter in a literary text.

2.1. Dialogue Identification: We construct dialogues as follows:

1. Start with the first text t_1 at the beginning of each chapter and collect t_i's in sequence. If there are more than n[1], narrations (back to back) interrupting conversations, $\{t_{i-n}^{i} = \mathbf{n}\}$ after the last utterance, then it is an indication of an end-of-dialogue, the dialogue ends.
2. Depending on the conversations, we select maximum of *n* narrations in the sequence preceding the first corresponding utterance to append *n* narrations to the head of the dialogue (overlapping from the previous dialogues permitted (see Fig.1)).
3. Max. of *n* narrations after the last utterance are appended to the foot of dialogue.

[1] *n* is a user defined variable. We iterated $n = \{2, 3\}$ in the experiments.

4. When a chapter ends, there is no more utterance/narration to add to the current dialogue.

2.2. Identifying Actors in a Novel: To identify the list of actors of a given novel, we scan the text for the actor names and mentions. In a given novel, an actor is usually referred to several different ways (*mentions*), e.g., Elizabeth Bennett, a character in Jane Austen's *Pride&Prejudice* (P&P) novel, is referred to *Liz, Lizzy, Miss Lizzy, Miss Bennett*, etc.

2.3. Extracting Expected List of Actors in Dialogues ($\mathbf{A}_d$): We use the dialogue structures and their corresponding list of actors to build the probabilistic ACTM to calculate expected probabilities of each actor based on the topics discussed in conversations. Since we do not use the labeled data (i.e., actor assignments are unknown), we extract a list of expected actors $\mathbf{A}_d$ that could be conversing in a given dialogue. Some utterances are easily attributed such that when the utterance contains *"..., said Mr Darcy."*, the speaker is *Mr. Darcy*. We captured such phrases (*said, whispered, replied, etc.*) to identify utterances with ***explicit*** actors. For the rest of the utterances in a given dialogue, we extract the list of expected actors $\mathbf{A}_d$ based on actor mentions. For a given dialogue, we extract around 5-10 actors. Our evaluations on the labeled P&P novel indicate that, the extracted list of possible actors in dialogues contain the actual actors of that dialogue most of the times.

dialogue-1		$t_0=n_0$	-	It was a ...
		$t_1=u_0$	A-1	*"I honor your circumspection.."*
		$t_2=u_1$	A-2	*"Nonsense, nonsense!"*
		$t_3=u_2$	A-1	*"What can be the meaning..."*
		$t_4=n_1$	-	Mary wished to say..
		$t_5=n_2$	-	She was sitting silence...
	dialogue-2	$t_6=u_3$	A-3	*"While Mary is adjusting ..."*
		$t_7=u_4$	A-4	*"I'm sick of Mr. Bingly.."*
		$t_8=u_5$	A-1	*"I'm sorry to hear that..."*
		$t_9=n_3$	-	The astonishment of the ladies...
		$t_{10}=n_4$	-	Though when the first....

Figure 1: Sequence of narratives (n) and utterances (u) from Jane Austen's Pride&Prejudice and identification of dialogues, A: actors. Dialogue-1 comprise of the first 6 text $t_0 - t_5$=$\{n_0, u_0 - u_2, n_1 - n_2\}$, and dialogue-2 comprises of the last 7 text $t_4 - t_{10}$=$\{n_1 - n_2, u_3 - n_5, n_3 - n_4\}$.

3 Actor-Topic Model (ACTM) and Inference

In author-topic model [10], each document with multiple authors is modeled as a distribution over topics, which are associated with the authors. Since the authorship information is given in priori, there is a one-to-one match between each document and on the authors of that document. Different from the author-topic model, we want to attribute each utterance in a given dialogue a possible actor. Although our goals are not so different, the prior information we can use to build the probabilistic model is not the same. Specifically, we do not have a deterministic one-to-one match between the actors and the dialogues because we are extracting information from an unlabeled dataset. Hence, we can only predict the expected actors (from the list of actors of a novel) for a given dialogue, under the assumption that the actual actor of any given utterance of a dialogue will be one of the expected list of actors. We also need to build a more focused model, where there is a one-to-one map between actors and utterances, not actors and dialogues; so we present the actor topic model below:

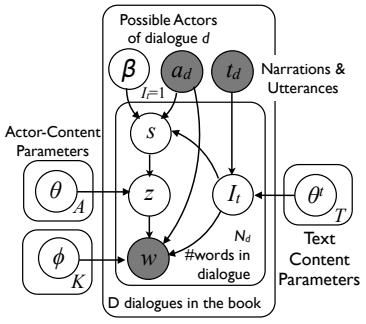

Figure 2: Graphical representation of the actor-topic model. Hyper-parameters are excluded for simplicity.

A dialogue d_i is a vector of N_d words, $\mathbf{w}_d = \{w_{nd}\}_{n=1}^{N_d}$, where each $w_{nd} \in \{1, ..., V\}$, is chosen from a vocabulary of size V, and a vector of A_d actors a_d, chosen from a set of actors of size A. In addition, since we wish to discover templates from dialogues that would attribute for bounded K concepts in text, utterances and narrations in a dialogue, we represent text as observed meta-variable t_d as shown in the graphical model in Fig. 2, a vector of text t_d in d_i. Thus; a collection of D dialogues is defined by $\mathcal{D} = \{(\mathbf{w}_1, \mathbf{a}_1, \mathbf{t}_1), ..., (\mathbf{w}_D, \mathbf{a}_D, \mathbf{t}_D)\}$.

The s indicates the actor responsible for a given word, chosen from the $\mathbf{a}_d$. An actor is sampled from the marginal prior distribution $GEM(\alpha)$ defines a distribution of partitions of unit interval into a countable number of parts. defined by the finite *stick-breaking* construction [12] where $A_d < \infty$ with α concentration parameter. $GEM(.)$ distribution named after Grifitths, Engen and McCloskey is a probability law for the sequence X_n arising as a *residual allocation model* RAM, $X_1 = U_1$, $X_n = (1 - U_1)(1 - U_2)...(1 - U_{n-1})U_n$, $n = 2, 3, ...$, where the residual fractions $U_1, U_2...$ are i.i.d. with beta $(1, \theta)$, $\theta(1 - x)^{\theta-1}$, $0 < x < 1$, for some $0 < \theta < \infty$.

At training time, we have the extracted actors a_d, only some of which can be explicitly identified (§ 2.2), hence we enforce each explicit actor in a dialogue to be assigned larger prior probabilities compared to the rest of the actors in a_d, because explicit actors are more likely to continue speaking if they have already spoken once.

Each actor is associated with a distribution over topics, $\theta_k^{(a)}$, chosen from a symmetric Dirichlet (α^a) prior. In addition, to discover which expected actors use which terms more frequently, we keep track of the words sampled for each actor (shown as a dependency between a_d and w in the graphical representation). Similarly, I_d indicates the text in a dialogue, either utterance or narration (Fig. 1) responsible for a given word, chosen from the list t_d that contains the word. Each text t_d is associated with a distribution over topics, $\theta^t \sim Dirichlet(\alpha^t)$. A text t_{di} in a dialogue is chosen uniformly at random from utterance/narrations containing the word (deterministic if only one text contains it). If sampled t_{di} is a narration, no author is sampled. The sampler only updates θ^t and ϕ, otherwise an actor is sampled from $\beta \sim GEM(\alpha)$ and θ is updated. If the sampled actor is *explicit* for the sampled text t_{di}, then only random variables for the explicit actor are updated. The proposed ACTM assumes the following generative process for a set of extracted dialogues given a novel:

- Draw the text level topic proportions $\theta^t \sim Dirichlet(\alpha^t)$; topic-word proportions $\phi \sim Dirichlet(\alpha^{(k)})$, $k = 1...K$; dialogue-level actor proportions $\beta^{(d)} \sim GEM(\alpha)$; actor-level topic proportions $\theta^a \sim Dirichlet(\alpha^a)$.
- For each word w_{id}, $i = 1, ..., N_d$ in a dialogue $d = 1, ...D$, given the vector of text $t_d \in \{u_d, n_d\}$; Conditioned on t_d, choose a text $I_t \sim Uniform(t_d)$:
  - if I_t=0 text is a narration, draw a topic $z_i \sim \theta^t$, and sample a word $w_{di} \sim Discrete(\phi_{z_i})$
  - if I_t=1 text is an utterance, conditioned on a_d, choose an actor $s_{t_d} \sim \beta^{(d)}$, sample a topic $z_{i,s} \sim \theta_{a_d}$, sample a word $w_{di} \sim Discrete(\phi_{z_{i,s}})$

The ACTM has unknown parameters, θ^a, β, θ^t, ϕ and latent variables corresponding to the assignments of individual words to topics, z, actors a, and text t. We construct Markov chain via Gibbs sampling which converges to the posterior distribution using update equations extracted from Bayesian inference, where when I_t=0, $p(z_i = k, I_t = 0|w_i, t_j, \mathbf{w}_{-i}, \mathbf{z}_{-i}) =$

$$[(n_{-i,k}^{(w_i)} + \alpha^k)/(n_{-i,k}^{(\cdot)} + V\alpha^k)] * [(n_{-k,j}^{(t_j)} + \alpha^t)/(n_j^{(t_j)} + K\alpha^t)] \quad (1)$$

and when I_t=1 using the probability assignment in (1);
$p(a_{t_j}^{(d)} = a, z_i = k, I_t = 1|w_i, t_j, \mathbf{w}_{-i}, \mathbf{z}_{-i}, \mathbf{a}_{d,-i}) =$

$$p(z_i = k, I_t = 0|...) * [(n_{-k,a}^{(a_{t_j})} + \alpha^a)/(n_a^{(a_{t_j})} + K\alpha^a)] * [(n_a^{(d)} + (\alpha/\#a_d))/(i - 1 + \alpha)] \quad (2)$$

where $n_{-i}^{(\cdot)}$ is a count that does not include the current assignment of z_i, t_j is the *j*th sentence in dialogue. Note from (2) (the last square-bracketed part) that since we use finite dimensional mixture model, the probability of seeing an actor $a_{t_j} = a$ is proportional to the number of actors already assigned to that dialogue. The hyper-parameters α, α^t, and α^a can be estimated by the fixed-point iteration method described in [7].

In the experiments, we wanted to test the effect of using the stick-breaking construction on the selection of the actors (actor assignments), which puts more weight on the explicit actors compared to the rest of the actors. For this reason we prepared another variation of the ACTM model where an actor is randomly chosen among a_d for a given utterance t_d from a uniform distribution $s_{t_d} \sim Uniform(a_d)$. Thus we refer to the latter model as ACTM, and the previous one where the explicit actors are given more preference via $GEM(\alpha)$ distribution the ACTM* model.

4 Experiments

4.1 Automatic Characterization of Actors from Actor Topic Model

In order to automatically compute a map between actors and dialogues in a given novel, we use the expected posterior actor, topic and word probabilities from the ACTM. We compiled two datasets of literary text: (i) Jane Austen's P&P from which we extracted 79 dialogues and list of 52 possible actors for each dialogue (§ 2); (ii) Emma, by the same author, from which 33 dialogues between set of 15 possible actors are extracted. To evaluate the performance of the model in predicting the

actors given any utterance in a dialogue $p(a_j^{(d)}|u_m^{(d)})$, we rank the actors per utterance based on the the following probability measures:
(1) ***Actor-Topic-Term Probabilities:*** Given an utterance u_m, we use the expected probability of its terms $\{w_{i=1}^{n_m}\} \in u_m$ that are sampled for a possible actor a_j averaged over different latent topic variable assignments $k = 1, ..., K$ obtained from ACTM output:

$$p_1(a_j|u_m) = \sum_{k=1}^{K} \left[\prod_{i=1}^{n_m} p(z_{ki}|w_i) * p(a_j|z_{ki}) \right] * p(z_k|u_m) \tag{3}$$

(2) ***Actor-Topic Probabilities:*** Using transformed Kullback-Leibler divergence as a measure of similarity between actor and an utterance based on the conversational topics being mentioned respectively, we introduce a new measure:

$$p_2(a_j|u_m) = \frac{1}{Z} e^{-\left[KL(p||\frac{p+q}{2}) + KL(q||\frac{p+q}{2})\right]} \tag{4}$$

where $p = \hat{\theta}(t_m|a_j)$ is the author-topic posterior probability distribution, $q = \hat{\theta}^t(t_m|u_m)$ is the sentence-topic posterior probability distribution from ACTM and Z is a normalization constant of the sum of the KL divergence between all the utterances u_d. We combine the p_1 and p_2 probabilities by interpolation and also include some heuristics in ranking actors. For instance in a conversational setting speaker take turns. Hence, if the previous actor of an utterance is explicit then it is an indication that the current utterance cannot be attributed to previous explicit actor. In such cases the list of possible actors of a corresponding utterance are updated and re-ranked.

Baseline: In the experiments we use the explicit actors of dialogues as the only information to build a baseline model and measure the performance.

Language Models for Benchmarking: There are many ways to construct a baseline classifier model to attribute utterances, e.g., CRF, SVM, etc., however most of these models would require labeled training data. We needed a baseline for a fair benchmark analysis since our ACTM model is an unsupervised model. Language models are particularly good at learning classifiers when there is limited or no labeled data. Treating each utterance as a sequence of words, we build a statistical language model (LM) for each actor separately using only the utterances with explicit actors attributed to the corresponding actor.

$$a_j^* = argmax_{1 \le j \le a_d} p(a_j|u_m) = argmax_{1 \le j \le a_d} P(w_1, ..., w_{n_m}|a_j) * P(a_j) \tag{5}$$

To assign an actor to each unlabeled utterance, we calculate the probability of its word sequences for each LM and rank based on the maximum likelihood probability.

Performance Measure: Once we calculate the probability of all possible actors $\mathbf{a}_d$ to be attributed to a given utterance and repeat for each utterance $\mathbf{u}_d$ using **ACTM**, **ACTM***, and **LMs**, we measure our performance against the gold-standard using a statistical measure, mean reciprocal rank (**MRR**), borrowed from information retrieval tasks. The reciprocal rank (RR) of a query response is the multiplicative inverse of the rank of the first correct answer and MRR is the average of the reciprocal ranks of results of a sample of queries. In an analogical way, the queries are our utterances, the correct answers are the possible actors and we rank the actors based on the performance measures for ACTM, ACTM* and the expected probabilities from LMs discussed above and measure MRR for U number of utterances in a novel as follows:

$$MRR = \frac{1}{U} \sum_{m=1}^{U} 1/rank_m \tag{6}$$

For example suppose we have "*Lizzy*" as the actual actor for an utterance. Our system predicts $n = 3$ most likely correct actors in sequence ranked based on the predicted probabilities: $\{$*Mrs. Bennet, Elizabeth, Darcy*$\}$. The RR for this utterance would be 1/2, since we predict the correct actor in rank 2 (after actor name normalization). We then calculate the MRR based on the basis of all utterances. Table 1 demonstrates the benchmark results based on MRR performance (top-1, top-3 and top5 MRR) for two 19th century novels: P&P and Emma. Note that there are around 33% utterances in P&P dialogues and similarly around 30% utterances in Emma dialogues that can be explicitly identified (Baseline Top-1 MRR results). The results of ACTM and ACTM* indicate that with no supervision our generative probabilistic models are able to attribute utterances much better than Language Models. In addition, using a special prior that assigns larger probabilities to explicit actors in dialogues via stick-breaking construction has a significant effect on the performance of the ACTM* in comparison to the random actor assignment in ACTM.

Table 1: MRR experiment results for top1, top3 and top5 evaluations on two novels: Pride&Prejudice and Emma by Jane Austin. LM:Language Model, ACTM uses random actor sampling, ACTM* uses stick-breaking construction for actor sampling.

	Pride&Prejudice			**Emma**		
MRR	Top-1	Top-3	Top-5	Top-1	Top-3	Top-5
Baseline	33.9	-	-	29.6	-	-
LM	37.8	42.6	46.4	38.1	46.9	51.1
ACTM	45.6	53.3	56.7	44.5	56.2	59.7
ACTM*	55.4	59.8	60.5	53.7	64.9	65.2

4.2 Social Network Construction via Topical Similarities between Actors

Using the results from the actor topic model with no labeled data, we would like to derive a conversational network structure. Previous work on spoken language processing for broadcast conversations and multi-party meetings proposed forming social network graphs by including a node for each conversation participant. Such approaches use the fact that one speaker speakes immediately after the other as an evidence for a relationship [4, 14]. The degree of the relationship is determined by the number of times these speakers speak after each other. Following a similar approach, we use the fact that speakers are involved in a conversation as an evidence of their relationship, however in this work we do not attribute relationships. In our network, the nodes (vertices) represent the actors and the links are the edge weights, the similarities measured between each actor. Our aim is to find hidden relationships between actors of a novel using the unsupervised ACTM* model. We construct a similarity measure between probability distributions of topics z_k given actors a_j as follows:

$$w_1(a_{j_1}, a_{j_2}) = \frac{1}{K} \sum_{k=1}^{K} e^{-\left[KL(p||\frac{p+q}{2}) + KL(q||\frac{p+q}{2})\right]} \tag{7}$$

where j_1 and j_2 are two separate actors and $p(z_k|a_{j_i})$ and $q(z_k|a_{j_i})$ are the expected (discrete) probabilities from ACTM*. We weigh the similarity measure in (7) based on the most frequent terms that actors are using in conversations:

$$w_2(a_{j_1}, a_{j_2}) = w_1 * \frac{1}{V} \prod_{v=1}^{V} p(w_v|a_i) * p(w_v|a_j) \tag{8}$$

We linked each actor by assigning undirected edges between nodes that represent adjacency in conversations based on Eq (8). We predicted top-10 most frequently conversing actor pairs from ACTM* model. In order to provide a reference to the social network results obtained via ACTM*, we constructed social network structures of the two novels using the labeled actors and the frequencies of conversations between these actors. We extracted top-10 pairs as the most frequently conversing actor pairs from the actual network structure as well and compared to the ACTM* output. The results show that ACTM* can correctly identify $\sim$60% of the top-10 most frequently conversing actors in Emma and $\sim$50% of top-10 most frequently conversing actors in P&P. Thus, ACTM* is a promising model also for constructing social network in literary text with there is no transcription.

5 Conclusion

Our probabilistic actor-topic model can both characterize dialogues in literary text, i.e., novels, and attribute each utterance in dialogues with a possible actor. Our model enables characterizing not only the actors of a given dialogue, but also extracts hidden topics that each individual actors of a given dialogue are conversing on. In particular, we described a probabilistic approach for detecting conversations between actors in a novel. Our findings indicate that proposed approach can be used to construct implicit topical similarities between actors in conversational settings and to automatically extract hidden relations between actors so as to assist construction of social networks for literary text. Our results thus far suggest further review of our methods for more insights into the social networks found in this and other genres of fiction. We would like to extend our work on todays literary text as well as short novels.

References

[1] N. Chambers and D. Jurafsky. Unsupervised learning of narrative event chains. In *ACL*, 2008.

[2] D.K Elson, N. Dames, and K.R. McKowen. Extracting social networks from literary fiction. In *ACL*, 2010.

[3] D.K. Elson and K.R. McKeown. Automatic attribution of quoted speech in literary narrative. In *AAAI*, 2010.

[4] N. P. Garg, S. Favre, H. Salamin, D. Hakkani Tür, and A. Vinciarelli. Role recognition for meeting participants: an approach based on lexical information and social network analysis. In *16th ACM international conference on Multimedia (MM)*, 2000.

[5] A. McCallum, X. Wang, and A. Corrada-Emmanual. Topic and role discovery in social networks and experiments in enron and academic e-mail. In *JAIR*, 2007.

[6] K. Miller, T. Griffiths, and M. Jordan. Nonparametric latent feature models for link prediction. In *NIPS*, 2009.

[7] T. Minka. Estimating a dirichlet distribution. In *Tech. Report, MIT*, 2000.

[8] F. Moretti. Atlas of the european novel 1800-1900. In *Verso, London*, 1999.

[9] F. Moretti. Graphs, maps, trees: Abstract models for a literary history. In *Verso, London*, 2005.

[10] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The author-topic model for authors and documents. In *NIPS*, 2004.

[11] P. Sarkar and A. W. Moore. Dynamic social network analysis using latent space models. In *NIPS*, 2005.

[12] J. Sethuraman. A constructive definition of dirichlet priors. In *Statistica Sinica*, volume 4, pages 639–650, 1994.

[13] B. Taskar, M. F. Wong, P. Abbeel, and D. Koller. Link prediction in relational data. In *NIPS*, 2003.

[14] A. Vinciarelli. Speakers role recognition in multiparty audio recordings using social network analysis and duration distribution modeling. In *IEEE Trans. on Multimedia*, volume 9(6), pages 1215–26, 2007.

[15] R. Xiang, J. Neville, and M. Rogati. Modeling relationship strength in online social networks. In *WWW 2010*, 2010.