Unsupervised Event Coreference Resolution with Rich Linguistic Features

Cosmin Adrian Bejan
Institute for Creative Technologies
University of Southern California
Marina del Rey, CA 90292, USA

Sanda Harabagiu
Human Language Technology Institute
University of Texas at Dallas
Richardson, TX 75083, USA

Abstract

This paper examines how a new class of nonparametric Bayesian models can be effectively applied to an open-domain event coreference task. Designed with the purpose of clustering complex linguistic objects, these models consider a potentially infinite number of features and categorical outcomes. The evaluation performed for solving both within- and cross-document event coreference shows significant improvements of the models when compared against two baselines for this task.

1 Introduction

The event coreference task consists of finding clusters of event mentions that refer to the same event. Although it has not been extensively studied in comparison with the related problem of entity coreference resolution, solving event coreference has already proved its usefulness in various applications such as topic detection and tracking (Allan et al., 1998), information extraction (Humphreys et al., 1997), question answering (Narayanan and Harabagiu, 2004), textual entailment (Haghighi et al., 2005), and contradiction detection (de Marneffe et al., 2008).

Previous approaches for solving event coreference relied on supervised learning methods that explore various linguistic properties in order to decide if a pair of event mentions is coreferential or not (Humphreys et al., 1997; Bagga and Baldwin, 1999; Ahn, 2006; Chen and Ji, 2009). In spite of being successful for a particular labeled corpus, these pairwise models are dependent on the domain or language that they are trained on. Moreover, since event coreference resolution is a complex task that involves exploring a rich set of linguistic features, annotating a large corpus with event coreference information for a new language or domain of interest requires a substantial amount of manual effort. Also, since these models are dependent on local pairwise decisions, they are unable to capture a global event distribution at topic or document collection level.

To address these limitations and to provide a more flexible representation for modeling observable data with rich properties, we present two novel, fully generative, nonparametric Bayesian models for unsupervised within- and cross-document event coreference resolution. The first model extends the *hierarchical Dirichlet process* (Teh et al., 2006) to take into account additional properties associated with observable objects (i.e., event mentions). The second model overcomes some of the limitations of the first model. It uses the *infinite factorial hidden Markov model* (Van Gael et al., 2008b) coupled to the *infinite hidden Markov model* (Beal et al., 2002) in order to (1) consider a potentially infinite number of features associated with observable objects, (2) perform an automatic selection of the most salient features, and (3) capture the structural dependencies of observable objects at the discourse level. Furthermore, both models are designed to account for a potentially infinite number of categorical outcomes (i.e., events). These models provide additional details and experimental results to our preliminary work on unsupervised event coreference resolution (Bejan et al., 2009).

2 Event Coreference

The problem of determining if two events are identical was originally studied in philosophy. One relevant theory on event identity was proposed by Davidson (1969) who argued that two events are identical if they have the same causes and effects. Later on, a different theory was proposed by Quine (1985) who considered that each event refers to a physical object (which is well defined in space and time), and therefore, two events are identical

Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pages 1412–1422,
Uppsala, Sweden, 11-16 July 2010.

if they have the same spatiotemporal location. In (Davidson, 1985), Davidson abandoned his suggestion to embrace the Quinean theory on event identity (Malpas, 2009).

2.1 An Example

In accordance with the Quinean theory, we consider that two event mentions are coreferential if they have the same *event properties* and share the same *event participants*. For instance, the sentences from Example 1 encode event mentions that refer to several individuated events. These sentences are extracted from a newly annotated corpus with event coreference information (see Section 4). In this corpus, we organize documents that describe the same seminal event into topics. In particular, the topics shown in this example describe the seminal event of buying ATI by AMD (topic 43) and the seminal event of buying EDS by HP (topic 44).

Although all the event mentions of interest emphasized in boldface in Example 1 evoke the same generic event *buy*, they refer to three individuated events: $e_1 = \{em_1, em_2\}$, $e_2 = \{em_{3-6}, em_8\}$, and $e_3 = \{em_7\}$. For example, em_1(*buy*) and em_3(*buy*) correspond to different individuated events since they have a different AGENT ([BUYER(em_1)=*AMD*] $\neq$ [BUYER(em_3)=*HP*]). This organization of event mentions leads to the idea of creating an event hierarchy which has on the first level, *event mentions*, on the second level, *individuated events*, and on the third level, *generic events*. In particular, the event hierarchy corresponding to the event mentions annotated in our example is illustrated in Figure 1.

Solving the event coreference problem poses many interesting challenges. For instance, in order to solve the coreference chain of event mentions that refer to the event e_2, we need to take into account the following issues: (i) a coreference chain can encode both within- and cross-document coreference information; (ii) two mentions from the same chain can have different word classes (e.g., em_3(*buy*)–verb, em_4(*purchase*)–noun); (iii) not all the mentions from the same chain are synonymous (e.g., em_3(*buy*) and em_8(*acquire*)), although a semantic relation might exist between them (e.g., in WordNet (Fellbaum, 1998), the genus of *buy* is *acquire*); (iv) partial (or all) properties and participants of an event mention can be omitted in text (e.g., em_4(*purchase*)). In Section 5, we discuss additional aspects of the event coreference problem that are not revealed in Example 1.

Topic 43

Document 3

s$_4$: *AMD agreed to* [***buy***]$_{em_1}$ *Markham, Ontario-based ATI for around $5.4 billion in cash and stock, the companies announced Monday.*

s$_5$: *The* [***acquisition***]$_{em_2}$ *would turn AMD into one of the world's largest providers of graphics chips.*

Topic 44

Document 2

s$_1$: *Hewlett-Packard is negotiating to* [***buy***]$_{em_3}$ *technology services provider Electronic Data Systems.*

s$_8$: *With a market value of about $115 billion, HP could easily use its own stock to finance the* [***purchase***]$_{em_4}$.

s$_9$: *If the* [***deal***]$_{em_5}$ *is completed, it would be HP's biggest* [***acquisition***]$_{em_6}$ *since it* [***bought***]$_{em_7}$ *Compaq Computer Corp. for $19 billion in 2002.*

Document 5

s$_2$: *Industry sources have confirmed to eWEEK that Hewlett-Packard will* [***acquire***]$_{em_8}$ *Electronic Data Systems for about $13 billion.*

Example 1: Examples of event mention annotations.

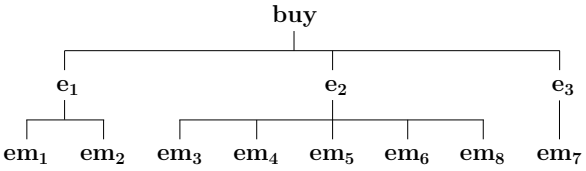

Figure 1: Fragment from the event hierarchy.

2.2 Linguistic Features

The events representing coreference clusters of event mentions are characterized by a large set of linguistic features. To compute an accurate event distribution for event coreference resolution, we associate the following categories of linguistic features with each annotated event mention.

Lexical Features (LF) We capture the lexical context of an event mention by extracting the following features: the head word (HW), the lemmatized head word (HL), the lemmatized left and right words surrounding the mention (LHL,RHL), and the HL features corresponding to the left and right mentions (LHE,RHE). For instance, the lexical features extracted for the event mention em_7(*bought*) from our example are HW:*bought*, HL:*buy*, LHL:*it*, RHL:*Compaq*, LHE:*acquisition*, and RHE:*acquire*.

Class Features (CF) These features aim to group mentions into several types of classes: the part-of-speech of the HW feature (POS), the word class of the HW feature (HWC), and the event class of the mention (EC). The HWC feature can take one of the following values: VERB, NOUN, ADJEC-

TIVE, and OTHER. As values for the EC feature, we consider the seven event classes defined in the TimeML specification language (Pustejovsky et al., 2003a): OCCURRENCE, PERCEPTION, REPORTING, ASPECTUAL, STATE, I_ACTION, and I_STATE. In order to extract the event classes corresponding to the event mentions from a given dataset, we employed the event extractor described in (Bejan, 2007). This extractor is trained on the TimeBank corpus (Pustejovsky et al., 2003b), which is a TimeML resource encoding temporal elements such as events, time expressions, and temporal relations.

WordNet Features (WF) In our efforts to create clusters of event mention attributes as close as possible to the true attribute clusters of the individuated events, we build two sets of word clusters using the entire lexical information from the WordNet database. After creating these sets of clusters, we then associate each event mention with only one cluster from each set. The first set uses the transitive closure of the WordNet SYNONYMOUS relation to form clusters with all the words from WordNet (WNS). For instance, the verbs *buy* and *purchase* correspond to the same cluster ID because there exist a chain of SYNONYMOUS relations between them in WordNet. The second set considers as grouping criteria the categorization of words from the WordNet lexicographer's files (WNL). In addition, for each word that is not covered in WordNet, we create a new cluster ID in each set of clusters.

Semantic Features (SF) To extract features that characterize participants and properties of event mentions, we use the semantic parser described in (Bejan and Hathaway, 2007). One category of semantic features that we identify for event mentions is the *predicate argument structures* encoded in PropBank annotations (Palmer et al., 2005). In PropBank, the predicate argument structures are represented by events expressed as verbs in text and by the semantic roles, or *predicate arguments*, associated with these events. For example, ARG0 annotates a specific type of semantic role which represents the AGENT, DOER, or ACTOR of a specific event. Another argument is ARG1, which plays the role of the PATIENT, THEME, or EXPERIENCER of an event. In particular, the predicate arguments associated to the event mention em_8(*bought*) from Example 1 are ARG0:[*it*], ARG1:[*Compaq Computer Corp.*], ARG3:[*for $19 billion*], and ARG-TMP:[*in 2002*].

Event mentions are not only expressed as verbs in text, but also as nouns and adjectives. Therefore, for a better coverage of semantic features, we also employ the semantic annotations encoded in the FrameNet corpus (Baker et al., 1998). FrameNet annotates word expressions capable of evoking conceptual structures, or *semantic frames*, which describe specific situations, objects, or events (Fillmore, 1982). The semantic roles associated with a word in FrameNet, or *frame elements*, are locally defined for the semantic frame evoked by the word. In general, the words annotated in FrameNet are expressed as verbs, nouns, and adjectives.

To preserve the consistency of semantic role features, we align frame elements to predicate arguments by running the PropBank semantic parser on the manual annotations from FrameNet; conversely, we also run the FrameNet parser on the manual annotations from PropBank. Moreover, to obtain a better alignment of semantic roles, we run both parsers on a large amount of unlabeled text. The result of this process is a map with all frame elements statistically aligned to all predicate arguments. For instance, in 99.7% of the cases the frame element BUYER of the semantic frame COMMERCE BUY is mapped to ARG0, and in the remaining 0.3% of the cases to ARG1. Additionally, we use this map to create a more general semantic feature which assigns to each predicate argument a frame element label. In particular, the features for em_8(*acquire*) are FEA0:BUYER, FEA1:GOODS, FEA3:MONEY, and FEATMP:TIME.

Two additional semantic features used in our experiments are: (1) the semantic frame (FR) evoked by every mention;[1] and (2) the WNS feature applied to the head word of every semantic role (e.g., WSA0, WSA1).

Feature Combinations (FC) We also explore various combinations of the features presented above. Examples include HW+HWC, HL+FR, FR+ARG1, LHL+RHL, etc.

It is worth noting that there exist event mentions for which not all the features can be extracted. For example, the LHE and RHE features are missing for the first and last event mentions in a document, respectively. Also, many semantic roles can be absent for an event mention in a given context.

[1] The reason for extracting this feature is given by the fact that, in general, frames are able to capture properties of generic events (Lowe et al., 1997).

3 Nonparametric Bayesian Models

As input for our models, we consider a collection of I documents, where each document i has J_i event mentions. For features, we make the distinction between *feature types* and *feature values* (e.g., POS is a feature type and has values such as NN and VB). Each event mention is characterized by L feature types, FT, and each feature type is represented by a finite vocabulary of feature values, fv. Thus, we can represent the observable properties of an event mention as a vector of L feature type – feature value pairs $\langle(\text{FT}_1 : fv_{1i}), \ldots, (\text{FT}_L : fv_{Li})\rangle$, where each feature value index i ranges in the feature value space associated with a feature type.

3.1 A Finite Feature Model

We present an extension of the *hierarchical Dirichlet process* (HDP) model which is able to represent each observable object (i.e., event mention) by a finite number of feature types L. Our HDP extension is also inspired from the Bayesian model proposed by Haghighi and Klein (2007). However, their model is strictly customized for entity coreference resolution, and therefore, extending it to include additional features for each observable object is a challenging task (Ng, 2008; Poon and Domingos, 2008).

In the HDP model, a *Dirichlet process* (DP) (Ferguson, 1973) is associated with each document, and each mixture component (i.e., event) is shared across documents. To describe its extension, we consider $\mathbf{Z}$ the set of indicator random variables for indices of events, ϕ_z the set of parameters associated with an event z, $\boldsymbol{\phi}$ a notation for all model parameters, and $\mathbf{X}$ a notation for all random variables that represent observable features.[2] Given a document collection annotated with event mentions, the goal is to find the best assignment of event indices $\mathbf{Z}^*$, which maximize the posterior probability $P(\mathbf{Z}|\mathbf{X})$. In a Bayesian approach, this probability is computed by integrating out all model parameters:

$$P(\mathbf{Z}|\mathbf{X}) = \int P(\mathbf{Z}, \phi|\mathbf{X})d\phi = \int P(\mathbf{Z}|\mathbf{X}, \phi)P(\phi|\mathbf{X})d\phi$$

Our HDP extension is depicted graphically in Figure 2(a). Similar to the HDP model, the distribution over events associated with each document, $\boldsymbol{\beta}$, is generated by a Dirichlet process with a concentration parameter $\alpha > 0$. Since this setting enables a clustering of event mentions at the document level, it is desirable that events be shared across documents and the number of events K be inferred from data. To ensure this flexibility, a global nonparametric DP prior with a hyperparameter γ and a global base measure H can be considered for $\boldsymbol{\beta}$ (Teh et al., 2006). The global distribution drawn from this DP prior, denoted as $\boldsymbol{\beta}_0$ in Figure 2(a), encodes the event mixing weights. Thus, same global events are used for each document, but each event has a document specific distribution β_i that is drawn from a DP prior centered on the global weights $\boldsymbol{\beta}_0$.

To infer the true posterior probability of $P(\mathbf{Z}|\mathbf{X})$, we follow (Teh et al., 2006) and use the *Gibbs sampling algorithm* (Geman and Geman, 1984) based on the direct assignment sampling scheme. In this sampling scheme, the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\phi}$ are integrated out analytically. Moreover, to reduce the complexity of computing $P(\mathbf{Z}|\mathbf{X})$, we make the naïve Bayes assumption that the feature variables $\mathbf{X}$ are conditionally independent given $\mathbf{Z}$. This allows us to factorize the joint distribution of feature variables $\mathbf{X}$ conditioned on $\mathbf{Z}$ into product of marginals. Thus, by Bayes rule, the formula for sampling an event index for mention j from document i, $Z_{i,j}$, is:[3]

$$P(Z_{i,j} \mid \mathbf{Z}^{-i,j}, \mathbf{X}) \propto P(Z_{i,j} \mid \mathbf{Z}^{-i,j}) \prod_{X \in \mathbf{X}} P(X_{i,j} | \mathbf{Z}, \mathbf{X}^{-i,j})$$

where $X_{i,j}$ represents the feature value of a feature type corresponding to the event mention j from the document i.

In the process of generating an event mention, an event index z is first sampled by using a mechanism that facilitates sampling from a prior for infinite mixture models called the *Chinese restaurant franchise* (CRF) representation, as reported in (Teh et al., 2006):

$$P(Z_{i,j} = z \mid \mathbf{Z}^{-i,j}, \beta_0) \propto \begin{cases} \alpha\beta_0^u, & \text{if } z = z_{new} \\ n_z + \alpha\beta_0^z, & \text{otherwise} \end{cases}$$

Here, n_z is the number of event mentions with event index z, z_{new} is a new event index not used already in $\mathbf{Z}^{-i,j}$, β_0^z are the global mixing proportions associated with the K events, and β_0^u is the weight for the unknown mixture component.

Next, to generate a feature value x (with the feature type X) of the event mention, the event z is

[2] In this subsection, the feature term is used in context of a feature type.

[3] $\mathbf{Z}^{-i,j}$ represents a notation for $\mathbf{Z} - \{Z_{i,j}\}$.

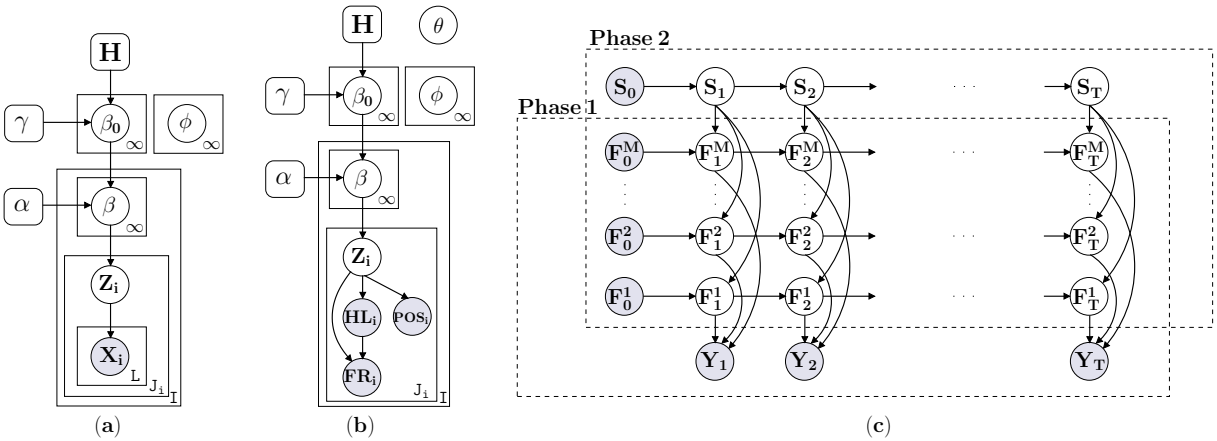

Figure 2: Graphical representation of our models: nodes correspond to random variables; shaded nodes denote observable variables; a rectangle captures the replication of the structure it contains, where the number of replications is indicated in the bottom-right corner. The model in (a) illustrates a flat representation of a limited number of features in a generalized framework (henceforth, HDP$_{flat}$). The model in (b) captures a simple example of structured network topology of three feature variables (henceforth, HDP$_{struct}$). The dependencies involving parameters ϕ and θ in these models are omitted for clarity. The model from (c) shows the representation of the iFHMM-iHMM model as well as the main phases of its generative process.

associated with a multinomial emission distribution over the feature values of X having the parameters $\boldsymbol{\phi} = \langle \phi^x_Z \rangle$. We assume that this emission distribution is drawn from a symmetric Dirichlet distribution with concentration λ_X:

$$P(X_{i,j} = x \mid \mathbf{Z}, \mathbf{X}^{-i,j}) \propto n_{x,z} + \lambda_X$$

where $X_{i,j}$ is the feature type of the mention j from the document i, and $n_{x,z}$ is the number of times the feature value x has been associated with the event index z in $(\mathbf{Z}, \mathbf{X}^{-i,j})$. We also apply the Lidstone's smoothing method to this distribution. In cases when only a feature type is considered (e.g., $\mathbf{X} = \langle HL \rangle$), the HDP$_{flat}$ model is identical with the original HDP model. We denote this one feature model by HDP$_{1f}$.

When dependencies between feature variables exist (e.g., in our case, frame elements are dependent on the semantic frames that define them, and frames are dependent on the words that evoke them), various global distributions are involved for computing $P(\mathbf{Z}|\mathbf{X})$. For the model depicted in Figure 2(b), for instance, the posterior probability is given by:

$$P(Z_{i,j}) P(FR_{i,j} \,|\, HL_{i,j}, \boldsymbol{\theta}) \prod_{X \in \mathbf{X}} P(X_{i,j} \,|\, \mathbf{Z})$$

In this formula, $P(FR_{i,j}|HL_{i,j}, \boldsymbol{\theta})$ is a global distribution parameterized by $\boldsymbol{\theta}$, and X is a feature variable from the set $\mathbf{X} = \langle HL, POS, FR \rangle$. For the sake of clarity, we omit the conditioning components of $\mathbf{Z}$, $\mathbf{HL}$, $\mathbf{FR}$, and $\mathbf{POS}$.

3.2 An Infinite Feature Model

To relax some of the restrictions of the first model, we devise an approach that combines the *infinite factorial hidden Markov model* (iFHMM) with the *infinite hidden Markov model* (iHMM) to form the iFHMM-iHMM model.

The iFHMM framework uses the *Markov Indian buffet process* (mIBP) (Van Gael et al., 2008b) in order to represent each object as a sparse subset of a potentially unbounded set of latent features (Griffiths and Ghahramani, 2006; Ghahramani et al., 2007; Van Gael et al., 2008a).[4] Specifically, the mIBP defines a distribution over an unbounded set of binary Markov chains, where each chain can be associated with a binary latent feature that evolves over time according to Markov dynamics. Therefore, if we denote by M the total number of feature chains and by T the number of observable components, the mIBP defines a probability distribution over a binary matrix $\mathbf{F}$ with T rows, which correspond to observations, and an unbounded number of columns M, which correspond to features. An observation y_t contains a subset from the unbounded set of features $\{f^1, f^2, \ldots, f^M\}$ that is represented in the matrix by a binary vector $\mathbf{F_t} = \langle F_t^1, F_t^2, \ldots, F_t^M \rangle$, where $F_t^i = 1$ indicates that f^i is associated with y_t. In other words, $\mathbf{F}$ decomposes the observations and represents them as feature factors, which can then be associated with hidden variables in an iFHMM model as depicted in Figure 2(c).

[4] In this subsection, a feature will be represented by a (feature type:feature value) pair.

Although the iFHMM allows a more flexible representation of the latent structure by letting the number of parallel Markov chains M be learned from data, it cannot be used as a framework where the number of clustering components K is infinite. On the other hand, the iHMM represents a nonparametric extension of the *hidden Markov model* (HMM) (Rabiner, 1989) that allows performing inference on an infinite number of states K. To further increase the representational power for modeling discrete time series data, we propose a nonparametric extension that combines the best of the two models, and lets the parameters M and K be learned from data.

As shown in Figure 2(c), each step in the new iHMM-iFHMM generative process is performed in two phases: (i) the latent feature variables from the iFHMM framework are sampled using the mIBP mechanism; and (ii) the features sampled so far, which become observable during this second phase, are used in an adapted version of the *beam sampling algorithm* (Van Gael et al., 2008a) to infer the clustering components (i.e., latent events).

In the first phase, the stochastic process for sampling features in $\mathbf{F}$ is defined as follows. The first component samples a number of $\mathsf{Poisson}(\alpha')$ features. In general, depending on the value that was sampled in the previous step $(t-1)$, a feature f^m is sampled for the t^{th} component according to the $P(F_t^m = 1 \mid F_{t-1}^m = 1)$ and $P(F_t^m = 1 \mid F_{t-1}^m = 0)$ probabilities.[5] After all features are sampled for the t^{th} component, a number of $\mathsf{Poisson}(\alpha'/t)$ new features are assigned for this component, and M gets incremented accordingly.

To describe the adapted beam sampler, which is employed in the second phase of the generative process, we introduce additional notations. We denote by $(s_1, \ldots, s_T)$ the sequence of hidden states corresponding to the sequence of event mentions $(y_1, \ldots, y_T)$, where each state s_t belongs to one of the K events, $s_t \in \{1, \ldots, K\}$, and each mention y_t is represented by a sequence of latent features $\langle F_t^1, F_t^2, \ldots, F_t^M \rangle$. One element of the transition probability $\boldsymbol{\pi}$ is defined as $\pi_{ij} = P(s_t = j \mid s_{t-1} = i)$, and a mention y_t is generated according to a likelihood model $\mathcal{F}$ that is parameterized by a state-dependent parameter ϕ_{s_t} $(y_t \mid s_t \sim \mathcal{F}(\phi_{s_t}))$. The observation parameters $\boldsymbol{\phi}$ are drawn independently from an identical prior base distribution H.

The beam sampling algorithm combines the ideas of slice sampling and dynamic programming for an efficient sampling of state trajectories. Since in time series models the transition probabilities have independent priors (Beal et al., 2002), Van Gael and colleagues (2008a) also used the HDP mechanism to allow couplings across transitions. For sampling the whole hidden state trajectory $\mathbf{s}$, this algorithm employs a forward filtering-backward sampling technique.

In the forward step of our adapted beam sampler, for each mention y_t, we sample features using the mIBP mechanism and the auxiliary variable $u_t \sim \mathsf{Uniform}(0, \pi_{s_{t-1}s_t})$. As explained in (Van Gael et al., 2008a), the auxiliary variables $\mathbf{u}$ are used to filter only those trajectories $\mathbf{s}$ for which $\pi_{s_{t-1}s_t} \geq u_t$ for all t. Also, in this step, we compute the probabilities $P(s_t \mid y_{1:t}, u_{1:t})$ for all t:

$$P(s_t|y_{1:t},u_{1:t}) \propto P(y_t|s_t) \sum_{s_{t-1}:u_t<\pi_{s_{t-1}s_t}} P(s_{t-1}|y_{1:t-1},u_{1:t-1})$$

Here, the dependencies involving parameters $\boldsymbol{\pi}$ and $\boldsymbol{\phi}$ are omitted for clarity.

In the backward step, we first sample the event for the last state s_T directly from $P(s_T \mid y_{1:T}, u_{1:T})$ and then, for all $t : T-1 \ldots 1$, we sample each state s_t given s_{t+1} by using the formula $P(s_t \mid s_{t+1}, y_{1:T}, u_{1:T}) \propto P(s_t \mid y_{1:t}, u_{1:t}) P(s_{t+1} \mid s_t, u_{t+1})$. To sample the emission distribution $\boldsymbol{\phi}$ efficiently, and to ensure that each mention is characterized by a finite set of representative features, we set the base distribution H to be conjugate with the data distribution $\mathcal{F}$ in a Dirichlet-multinomial model with the multinomial parameters $(o_1, \ldots, o_K)$ defined as:

$$o_k = \sum_{t=1}^{T} \sum_{f^m \in B_t} n_{mk}$$

In this formula, n_{mk} counts how many times the feature f^m was sampled for the event k, and B_t stores a finite set of features for y_t.

The mechanism for building a finite set of representative features for the mention y_t is based on *slice sampling* (Neal, 2003). Letting q_m be the number of times the feature f^m was sampled in the mIBP, and v_t an auxiliary variable for y_t such that $v_t \sim \mathsf{Uniform}(1, \max\{q_m : F_t^m = 1\})$, we define the finite feature set B_t for the observation y_t as $B_t = \{f^m : F_t^m = 1 \wedge q_m \geq v_t\}$. The finiteness of this feature set is based on the observation that, in the generative process of the mIBP, only a finite set

[5] Technical details for computing these probabilities are described in (Van Gael et al., 2008b).

of features are sampled for a component. We denote this model as iFHMM-iHMM$_{uniform}$. Also, it is worth mentioning that, by using this type of sampling, only the most representative features of y_t get selected in B_t.

Furthermore, we explore the mechanism for selecting a finite set of features associated with an observation by: (1) considering all the observation's features whose corresponding feature counter $q_m \geq 1$ ($unfiltered$); (2) selecting only the higher half of the feature distribution consisting of the observation's features that were sampled at least once in the mIBP model ($median$); and (3) sampling v_t from a discrete distribution of the observation's features that were sampled at least once in the mIBP ($discrete$).

4 Experiments

Datasets One dataset we employed is the automatic content extraction (ACE) (ACE-Event, 2005). However, the utilization of the ACE corpus for the task of solving event coreference is limited because this resource provides only within-document event coreference annotations using a restricted set of event types such as LIFE, BUSINESS, CONFLICT, and JUSTICE. Therefore, as a second dataset, we created the **EventCorefBank** (ECB) corpus[6] to increase the diversity of event types and to be able to evaluate our models for both within- and cross-document event coreference resolution. One important step in the creation process of this corpus consists in finding sets of related documents that describe the same seminal event such that the annotation of coreferential event mentions across documents is possible. For this purpose, we selected from the GoogleNews archive[7] various topics whose description contains keywords such as *commercial transaction*, *attack*, *death*, *sports*, *terrorist act*, *election*, *arrest*, *natural disaster*, etc. The entire annotation process for creating the ECB resource is described in (Bejan and Harabagiu, 2008). Table 1 lists several basic statistics extracted from these two corpora.

Evaluation For a more realistic approach, we not only trained the models on the manually annotated event mentions (i.e., true mentions), but also on all the possible mentions encoded in the two datasets. To extract all event mentions, we ran the event identifier described in (Bejan, 2007). The mentions extracted by this system (i.e., system mentions) were able to cover all the true mentions from both datasets. As shown in Table 1, we extracted from ACE and ECB corpora 45289 and 21175 system mentions, respectively.

	ACE	ECB
Number of topics	–	43
Number of documents	745	482
Number of within-topic events	–	339
Number of cross-document events	–	208
Number of within-document events	4946	1302
Number of true mentions	6553	1744
Number of system mentions	45289	21175
Number of distinct feature values	391798	237197

Table 1: Statistics of the ACE and ECB corpora.

We report results in terms of recall (R), precision (P), and F-score (F) by employing the *mention*-based B^3 metric (Bagga and Baldwin, 1998), the *entity*-based CEAF metric (Luo, 2005), and the pairwise F1 (PW) metric. All the results are averaged over 5 runs of the generative models. In the evaluation process, we considered only the true mentions of the ACE test dataset, and the event mentions of the test sets derived from a 5-fold cross validation scheme on the ECB dataset. For evaluating the cross-document coreference annotations, we adopted the same approach as described in (Bagga and Baldwin, 1999) by merging all the documents from the same topic into a meta-document and then scoring this document as performed for within-document evaluation. For both corpora, we considered a set of 132 feature types, where each feature type consists on average of 3900 distinct feature values.

Baselines We consider two baselines for event coreference resolution (rows 1&2 in Tables 2&3). One baseline groups each event mention by its event class (BL$_{eclass}$). Therefore, for this baseline, we cluster mentions according to their corresponding EC feature value. Similarly, the second baseline uses as grouping criteria for event mentions their corresponding WNS feature value (BL$_{syn}$).

HDP Extensions Due to memory limitations, we evaluated the HDP models on a restricted set of manually selected feature types. In general, the HDP$_{1f}$ model with the feature type HL, which plays the role of a baseline for the HDP$_{flat}$ and HDP$_{struct}$ models, outperforms both baselines on the ACE and ECB datasets. For the HDP$_{flat}$ models (rows 4–7 in Tables 2&3), we classified the experiments according to the set of feature types described in Section 2. Our experiments reveal that the best configuration of features for this model

[6] ECB is available at http://www.hlt.utdallas.edu/∼ady.
[7] http://news.google.com/

	Model configuration	B^3			CEAF			PW			B^3			CEAF			PW		
		R	P	F	R	P	F	R	P	F	R	P	F	R	P	F	R	P	F
		ECB | WD									ECB | CD								
1	BL$_{eclass}$	97.7	55.8	71.0	44.5	80.1	57.2	93.7	25.4	39.8	93.8	49.6	64.9	36.6	72.7	48.7	90.7	28.6	43.3
2	BL$_{syn}$	91.5	57.4	70.5	45.7	75.9	57.0	65.3	21.9	32.6	84.6	48.1	61.3	32.8	63.6	43.3	66.2	26.0	37.3
3	HDP$_{1f}$ (HL)	84.3	89.0	86.5	83.4	79.6	81.4	36.6	53.4	42.6	67.0	86.2	75.3	76.2	57.1	65.2	34.9	58.9	43.5
4	HDP$_{flat}$ (LF)	81.4	98.2	89.0	92.7	77.2	84.2	24.7	82.8	37.7	63.8	97.3	77.0	84.9	54.3	66.1	27.2	88.5	41.5
5	(LF+CF)	81.5	98.0	89.0	92.8	77.9	84.7	24.6	80.7	37.4	64.6	97.3	77.6	85.3	55.6	67.2	27.6	88.7	42.0
6	(LF+CF+WF)	82.0	98.9	89.6	93.7	78.4	85.3	26.8	89.9	41.0	65.8	98.0	78.7	86.7	57.1	68.8	29.6	93.0	44.8
7	(LF+CF+WF+SF)	82.1	99.2	89.8	93.9	78.2	85.3	27.0	92.4	41.3	65.0	98.7	78.3	86.9	56.0	68.0	29.2	95.1	44.4
8	HDP$_{struct}$ (HL→FR→FEA)	84.3	97.1	**90.2**	92.7	81.1	**86.5**	34.4	83.0	**48.6**	69.3	95.8	**80.4**	86.2	60.1	**70.8**	37.5	85.6	**52.1**
9	iFHMM-iHMM$_{unfiltered}$	82.6	97.7	89.5	92.7	78.5	85.0	28.5	82.4	41.8	67.2	96.4	79.1	85.6	58.0	69.1	32.5	87.7	47.2
10	iFHMM-iHMM$_{discrete}$	82.6	98.1	89.7	93.2	79.0	85.5	29.7	85.4	44.0	66.2	96.2	78.4	84.8	57.2	68.3	32.2	88.1	47.1
11	iFHMM-iHMM$_{median}$	82.6	97.8	89.5	92.9	78.8	85.3	29.3	83.7	43.0	67.0	96.5	79.0	86.1	58.3	69.5	33.1	88.1	47.9
12	iFHMM-iHMM$_{uniform}$	82.5	98.1	89.6	93.1	78.8	85.3	29.4	86.6	43.7	67.0	96.4	79.0	85.5	58.0	69.1	33.3	88.3	48.2

Table 2: Results for within-document (WD) and cross-document (WD) coreference resolution on the ECB dataset.

	B^3			CEAF			PW		
	R	P	F	R	P	F	R	P	F
	ACE | WD								
1	97.9	25.0	39.9	14.7	64.4	24.0	93.5	8.2	15.2
2	89.3	36.7	52.1	25.1	64.8	36.2	63.8	10.5	18.1
3	86.0	70.6	77.5	62.3	76.4	68.6	50.5	27.7	35.8
4	82.9	82.6	82.7	74.9	75.8	75.3	42.4	41.9	42.1
5	82.0	84.9	83.4	77.8	75.3	76.6	37.9	45.1	41.2
6	83.3	83.6	83.4	76.3	76.2	76.3	42.2	43.9	43.0
7	83.4	84.2	**83.8**	76.9	76.5	**76.7**	43.3	47.1	**45.1**
8	86.2	76.9	81.3	69.0	77.5	73.0	53.2	38.1	44.4
9	82.8	83.6	83.2	75.8	75.0	75.4	41.4	42.6	42.0
10	83.1	81.5	82.3	73.7	75.1	74.4	41.9	40.1	41.0
11	83.0	81.3	82.1	73.2	75.2	74.2	40.7	39.0	39.8
12	81.9	82.2	82.1	74.6	74.5	74.5	37.2	39.0	38.1

Table 3: Results for WD coreference resolution on ACE.

consists of a combination of feature types from all the categories of features (row 7). For the HDP$_{struct}$ experiments, we considered the set of features of the best HDP$_{flat}$ experiment as well as the dependencies between HL, FR, and FEA. Overall, we can assert that HDP$_{flat}$ achieved the best performance results on the ACE test dataset (Table 3), whereas HDP$_{struct}$ proved to be more effective on the ECB dataset (Table 2). Moreover, the results of the HDP$_{flat}$ and HDP$_{struct}$ models show an F-score increase by 4-10% over HDP$_{1f}$, and therefore, the results prove that the HDP extension provides a more flexible representation for clustering objects with rich properties.

We also plot the evolution of our generative processes. For instance, Figure 3(a) shows that the HDP$_{flat}$ model corresponding to row 7 in Table 3 converges in 350 iteration steps to a posterior distribution over event mentions from ACE with around 2000 latent events. Additionally, our experiments with different values of the λ parameter for the Lidstone's smoothing method indicate that this smoothing method is useful for improving the performance of the HDP models. However, we could not find a λ value in our experiments that brings a major improvement over the non-smoothed HDP models. Figure3(b) shows the performances of HDP$_{struct}$ on ECB with various λ values.[8] The HDP results from Tables 2&3 correspond to a λ value of 10^{-4} and 10^{-2} for HDP$_{flat}$ and HDP$_{struct}$, respectively.

iFHMM-iHMM In spite of the fact that the iFHMM-iHMM model employs automatic feature selection, its results remain competitive against the results of the HDP models, where the feature types were manually tuned. When comparing the strategies for filtering feature values in this framework, we could not find a distinct separation between the results obtained by the $unfiltered$, $discrete$, $median$, and $uniform$ models. As observed from Tables 2&3, most of the iFHMM-iHMM results fall in between the HDP$_{flat}$ and HDP$_{struct}$ results. The results were obtained by automatically selecting only up to 1.5% of distinct feature values. Figure 3(c) shows the percents of features employed by this model for various values of the parameter α' that controls the number of sampled features. The best results (also listed in Tables 2&3) were obtained for $\alpha' = 10$ (0.05%) on ACE and $\alpha' = 150$ (0.91%) on ECB.

To show the usefulness of the sampling schemes considered for this model, we also compare in Table 4 the results obtained by an iFHMM-iHMM model that considers all the feature values associated with an observable object (iFHMM-iHMM$_{all}$) against the iFHMM-iHMM models that employ the mIBP sampling scheme together with the $unfiltered$, $discrete$, $median$, and $uniform$ filtering schemes. Because of the memory limitation constraints, we performed the experiments listed in Table 4 by selecting only a subset from

[8] A configuration $\lambda = 0$ in the Lidstone's smoothing method is equivalent with a non-smoothed version of the model on which it is applied.

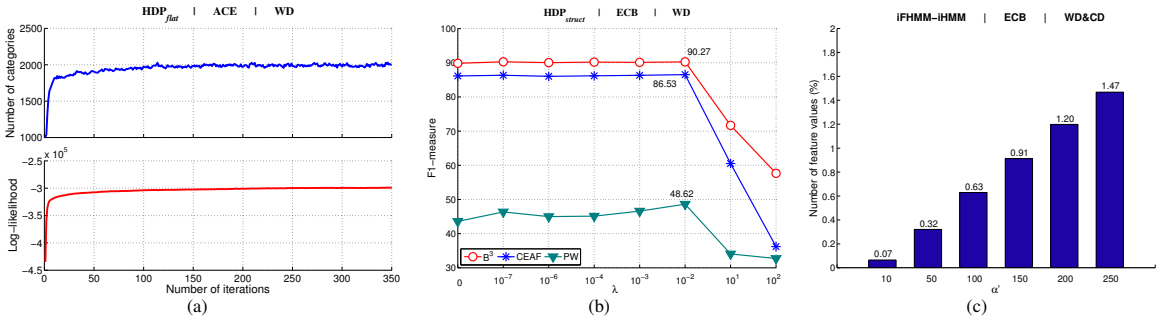

Figure 3: (a) Evolution of K and log-likelihood in the HDP$_{flat}$ model. (b) Evaluation of the Lidstone's smoothing method in the HDP$_{struct}$ model. (c) Counts of features employed by the iFHMM-iHMM model for various α' values.

Model	B^3			CEAF			PW		
	R	P	F	R	P	F	R	P	F
ACE | WD									
all	89.3	39.8	55.0	30.2	68.8	42.0	62.7	9.1	15.9
unfiltered	83.3	77.7	80.4	70.6	75.9	73.2	42.1	34.6	38.0
discrete	83.8	80.7	82.2	73.0	75.8	74.4	43.9	39.1	41.4
median	83.5	80.2	81.8	72.2	75.3	73.7	42.7	38.2	40.3
uniform	82.8	80.7	81.7	72.8	75.2	73.9	41.4	39.3	40.3
ECB | WD									
all	89.5	62.5	73.6	53.3	76.5	62.8	60.7	22.9	33.2
unfiltered	82.6	96.6	89.0	92.0	79.1	85.1	28.4	75.6	41.0
discrete	83.1	96.7	89.4	91.6	79.2	84.9	30.5	79.0	43.9
median	82.5	97.3	89.3	92.8	78.9	85.3	29.2	78.8	42.0
uniform	82.7	96.0	88.9	91.1	79.0	84.6	29.3	74.9	41.6
ECB | CD									
all	79.3	54.4	64.5	43.3	61.3	50.7	59.6	26.2	36.4
unfiltered	67.2	94.5	78.5	84.7	59.2	69.6	32.8	82.5	46.8
discrete	67.6	94.8	78.9	83.8	58.3	68.8	34.3	85.3	48.9
median	66.7	95.2	78.4	84.5	57.7	68.5	32.2	83.7	46.3
uniform	67.7	93.6	78.4	83.6	59.2	69.2	33.6	79.5	46.9

Table 4: Feature non-sampling vs. feature sampling in the iFHMM-iHMM model.

the feature types which proved to be salient in the HDP experiments. As listed in Table 4, all the iFHMM-iHMM models that used a feature sampling scheme significantly outperform the iFHMM-iHMM$_{all}$ model; this proves that all the sampling schemes considered in the iFHMM-iHMM framework are able to successfully filter out noisy and redundant feature values.

The closest comparison to prior work is the supervised approach described in (Chen and Ji, 2009) that achieved a 92.2% B^3 F-measure on the ACE corpus. However, for this result, ground truth event mentions as well as a manually tuned coreference threshold were employed.

5 Error Analysis

One frequent error occurs when a more complex form of semantic inference is needed to find a correspondence between two event mentions of the same individuated event. For instance, since all properties and participants of em_3(*deal*) are omitted in our example and no common features exist between em_3(*buy*) and em_1(*buy*) to indicate a similarity between these mentions, they will most probably be assigned to different clusters. This example also suggests the need for a better modeling of the discourse salience for event mentions.

Another common error is made when matching the semantic roles corresponding to coreferential event mentions. Although we simulated entity coreference by using various semantic features, the task of matching participants of coreferential event mentions is not completely solved. This is because, in many coreferential cases, partonomic relations between semantic roles need to be inferred.[9] Examples of such relations extracted from ECB are *Israeli forces* $\xrightarrow{\text{PART OF}}$ *Israel*, *an Indian warship* $\xrightarrow{\text{PART OF}}$ *the Indian navy*, *his cell* $\xrightarrow{\text{PART OF}}$ *Sicilian jail*. Similarly for event properties, many coreferential examples do not specify a clear location and time interval (e.g., *Jabaliya refugee camp* $\xrightarrow{\text{PART OF}}$ *Gaza*, *Tuesday* $\xrightarrow{\text{PART OF}}$ *this week*). In future work, we plan to build relevant clusters using partonomies and taxonomies such as the WordNet hierarchies built from MERONYMY/HOLONYMY and HYPERNYMY/HYPONYMY relations, respectively.[10]

6 Conclusion

We have presented two novel, nonparametric Bayesian models that are designed to solve complex problems that require clustering objects characterized by a rich set of properties. Our experiments for event coreference resolution proved that these models are able to solve real data applications in which the feature and cluster numbers are treated as free parameters, and the selection of feature values is performed automatically.

[9] This observation was also reported in (Hasler and Orasan, 2009). [10] This task is not trivial since, if applying the transitive closure on these relations, all words will end up being part from the same cluster with *entity* for instance.

References

ACE-Event. 2005. ACE (Automatic Content Extraction) English Annotation Guidelines for Events, version 5.4.3 2005.07.01.

David Ahn. 2006. The stages of event extraction. In *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*, pages 1–8.

James Allan, Jaime Carbonell, George Doddington, Jonathan Yamron, and Yiming Yang. 1998. Topic Detection and Tracking Pilot Study: Final Report. In *Proceedings of the Broadcast News Understanding and Transcription Workshop*, pages 194–218.

Amit Bagga and Breck Baldwin. 1998. Algorithms for Scoring Coreference Chains. In *Proceedings of the 1st International Conference on Language Resources and Evaluation (LREC-1998).*

Amit Bagga and Breck Baldwin. 1999. Cross-Document Event Coreference: Annotations, Experiments, and Observations. In *Proceedings of the ACL Workshop on Coreference and its Applications*, pages 1–8.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics (COLING-ACL).*

Matthew J. Beal, Zoubin Ghahramani, and Carl Edward Rasmussen. 2002. The Infinite Hidden Markov Model. In *Advances in Neural Information Processing Systems 14 (NIPS).*

Cosmin Adrian Bejan and Sanda Harabagiu. 2008. A Linguistic Resource for Discovering Event Structures and Resolving Event Coreference. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC).*

Cosmin Adrian Bejan and Chris Hathaway. 2007. UTD-SRL: A Pipeline Architecture for Extracting Frame Semantic Structures. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval)*, pages 460–463.

Cosmin Adrian Bejan, Matthew Titsworth, Andrew Hickl, and Sanda Harabagiu. 2009. Nonparametric Bayesian Models for Unsupervised Event Coreference Resolution. In *Advances in Neural Information Processing Systems 23 (NIPS).*

Cosmin Adrian Bejan. 2007. Deriving Chronological Information from Texts through a Graph-based Algorithm. In *Proceedings of the 20th Florida Artificial Intelligence Research Society International Conference (FLAIRS), Applied Natural Language Processing track.*

Zheng Chen and Heng Ji. 2009. Graph-based Event Coreference Resolution. In *Proceedings of the 2009 Workshop on Graph-based Methods for Natural Language Processing (TextGraphs-4)*, pages 54–57.

Donald Davidson, 1969. *The Individuation of Events*. In N. Rescher et al., eds., *Essays in Honor of Carl G. Hempel*, Dordrecht: Reidel. Reprinted in D. Davidson, ed., *Essays on Actions and Events*, 2001, Oxford: Clarendon Press.

Donald Davidson, 1985. *Reply to Quine on Events*, pages 172–176. In E. LePore and B. McLaughlin, eds., *Actions and Events: Perspectives on the Philosophy of Donald Davidson*, Oxford: Blackwell.

Marie-Catherine de Marneffe, Anna N. Rafferty, and Christopher D. Manning. 2008. Finding Contradictions in Text. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT)*, pages 1039–1047.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Thomas S. Ferguson. 1973. A Bayesian Analysis of Some Nonparametric Problems. *The Annals of Statistics*, 1(2):209–230.

Charles J. Fillmore. 1982. Frame Semantics. In *Linguistics in the Morning Calm*.

Stuart Geman and Donald Geman. 1984. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741.

Zoubin Ghahramani, T. L. Griffiths, and Peter Sollich, 2007. *Bayesian Statistics 8*, chapter Bayesian nonparametric latent feature models, pages 201–225. Oxford University Press.

Tom Griffiths and Zoubin Ghahramani. 2006. Infinite Latent Feature Models and the Indian Buffet Process. In *Advances in Neural Information Processing Systems 18 (NIPS)*, pages 475–482.

Aria Haghighi and Dan Klein. 2007. Unsupervised Coreference Resolution in a Nonparametric Bayesian Model. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL)*, pages 848–855.

Aria Haghighi, Andrew Ng, and Christopher Manning. 2005. Robust Textual Inference via Graph Matching. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT-EMNLP)*, pages 387–394.

Laura Hasler and Constantin Orasan. 2009. Do coreferential arguments make event mentions coreferential? In *Proceedings of the 7th Discourse Anaphora and Anaphor Resolution Colloquium (DAARC 2009).*

Kevin Humphreys, Robert Gaizauskas, and Saliha Azzam. 1997. Event coreference for information extraction. In *Proceedings of the Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts, 35th Meeting of ACL*, pages 75–81.

John B. Lowe, Collin F. Baker, and Charles J. Fillmore. 1997. A frame-semantic approach to semantic annotation. In *Proceedings of the SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How?*, pages 18–24.

Xiaoqiang Luo. 2005. On coreference resolution performance metrics. In *Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (EMNLP-2005)*, pages 25–32.

Jeff Malpas. 2009. Donald Davidson. In *The Stanford Encyclopedia of Philosophy (Fall 2009 Edition), Edward N. Zalta (ed.), http://plato.stanford.edu/archives/fall2009/entries/davidson/*.

Srini Narayanan and Sanda Harabagiu. 2004. Question Answering Based on Semantic Structures. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING)*, pages 693–701.

Radford M. Neal. 2003. Slice Sampling. *The Annals of Statistics*, 31:705–741.

Vincent Ng. 2008. Unsupervised Models for Coreference Resolution. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 640–649.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–105.

Hoifung Poon and Pedro Domingos. 2008. Joint Unsupervised Coreference Resolution with Markov Logic. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 650–659.

James Pustejovsky, Jose Castano, Bob Ingria, Roser Sauri, Rob Gaizauskas, Andrea Setzer, and Graham Katz. 2003a. TimeML: Robust Specification of Event and Temporal Expressions in Text. In *Proceedings of the Fifth International Workshop on Computational Semantics (IWCS)*.

James Pustejovsky, Patrick Hanks, Roser Sauri, Andrew See, Robert Gaizauskas, Andrea Setzer, Dragomir Radev, Beth Sundheim, David Day, Lisa Ferro, and Marcia Lazo. 2003b. The TimeBank Corpus. In *Corpus Linguistics*, pages 647–656.

W. V. O. Quine, 1985. *Events and Reification*, pages 162–171. In E. LePore and B. P. McLaughlin, eds., *Actions and Events: Perspectives on the philosophy of Donald Davidson*, Oxford: Blackwell. Reprinted in R. Casati and A. C. Varzi, eds., *Events*, 1996, pages 107–116, Aldershot: Dartmouth.

Lawrence R. Rabiner. 1989. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. In *Proceedings of the IEEE*, pages 257–286.

Yee Whye Teh, Michael Jordan, Matthew Beal, and David Blei. 2006. Hierarchical Dirichlet Processes. *Journal of the American Statistical Association*, 101(476):1566–1581.

Jurgen Van Gael, Y. Saatci, Yee Whye Teh, and Zoubin Ghahramani. 2008a. Beam Sampling for the Infinite Hidden Markov Model. In *Proceedings of the 25th Annual International Conference on Machine Learning (ICML)*, pages 1088–1095.

Jurgen Van Gael, Yee Whye Teh, and Zoubin Ghahramani. 2008b. The Infinite Factorial Hidden Markov Model. In *Advances in Neural Information Processing Systems 21 (NIPS)*.