

# Unsupervised Classification of Dialogue Acts using a Dirichlet Process Mixture Model

Nigel Crook, Ramon Granell, and Stephen Pulman

Oxford University Computing Laboratory

Wolfson Building

Parks Road, OXFORD, UK

nigc@comlab.ox.ac.uk

ramg@comlab.ox.ac.uk

sgp@clg.ox.ac.uk

## Abstract

In recent years Dialogue Acts have become a popular means of modelling the communicative intentions of human and machine utterances in many modern dialogue systems. Many of these systems rely heavily on the availability of dialogue corpora that have been annotated with Dialogue Act labels. The manual annotation of dialogue corpora is both tedious and expensive. Consequently, there is a growing interest in unsupervised systems that are capable of automating the annotation process. This paper investigates the use of a Dirichlet Process Mixture Model as a means of clustering dialogue utterances in an unsupervised manner. These clusters can then be analysed in terms of the possible Dialogue Acts that they might represent. The results presented here are from the application of the Dirichlet Process Mixture Model to the Dihana corpus.

## 1 Introduction

Dialogue Acts (DAs) are an important contribution from discourse theory to the design of dialogue systems. These linguistics abstractions are based on the illocutionary force of speech acts (Austin, 1962) and try to capture and model the communicative intention of human or machine utterances. In recent years, several dialogue systems have made use of DAs for modelling discourse phenomena in either the Dialogue Manager (Keizer et al., 2008), Automatic Speech Recogniser (Stolcke et al., 2000) or the Automatic Speech Synthesiser (Zovato and Romportl, 2008). Additionally, they have been used also in

other tasks such as summarisation, (Murray et al., 2006). Therefore, a correct DA classification of dialogue turns can bring benefits to the performance of these modules and tasks.

Many machine learning approaches have been used to automatically label DAs. They are usually based on Supervised Learning techniques involving combinations of Ngrams and Hidden Markov Models (Stolcke et al., 2000; Martínez-Hinarejos et al., 2008), Neural Networks (Garfield and Wermter, 2006) or Graphical Models (Ji and Bilmes, 2005). Relatively few approaches to DA classification have been based on unsupervised learning methods. Some promising results were reported by Anderach et al (Andernach et al., 1997; Andernach, 1996) who applied Kohonen Self Organising Maps (SOMs) to the problem of DA classification. Although the SOM is *nonparametric* in the sense that it doesn't require that the number of clusters to be found in the data be a parameter of the SOM that is specified before clustering begins, it's capacity to detect clusters is limited to the size of the two-dimensional lattice onto which the clusters are projected, and the size of this lattice *is* determined prior to clustering. This paper investigates the use of an unsupervised, non-parametric Bayesian approach to automatic DA labelling: namely the Dirichlet Process Mixture Model (DPMM). Specifically, the paper reports results from applying the Chinese Restaurant Process (CRP), a popular approach to DPMMs, to the automatic labelling of DAs in the Dihana corpus. The Dihana corpus (J.M.Benedí et al., 2006) has previously been used for the same task but with a supervised learning approach (Martínez-Hinarejos et al., 2008). The results reported here indicate that, treating each utterance as a *bag of words*, the CRP is capable of automatically clus-

tering most utterances according to speaker, level 1 and in some cases level 2 DA annotations (see below).

## 2 The Dihana corpus

The Dihana corpus consists of human-computer spoken dialogues in Spanish about queuing information of train fares and timetables. The acquisition was performed using the Wizard of Oz (WoZ) technique, where a human simulates the system following a prefixed strategy. User and system utterances are different in nature, user utterances are completely spontaneous speech whereas system utterances are based on pre-written patterns that the WoZ selected according to what the user said in the previous turn, the current dialogue state and the WoZ strategy. There is a total of 900 dialogues with a vocabulary of 823 words. However, after applying a process of name entity recognition (cities, times, number, ...) and making the distinction between system and user words there are 964 different words. The same process of name entity recognition was also used by Martínez Hinarejos (Martínez-Hinarejos et al., 2008)

### 2.1 Annotation scheme

Dialogues were manually annotated using a dialogue act annotation scheme based on three levels (see Table 1). The first level corresponds to the general intention of the speaker (speech act), the second level represents the implicit information that is referred to in the first level and the third level is the specific data provided in the utterance. Using these three levels and making the distinction between user and system labels, there are 248 different labels (153 for the user and 95 for the system). Combining only first and second level there are 72 labels (45 for user and 27 for system), and with only first level there are 16 labels (7 for user and 9 for system).

Annotation was done at utterance level. That is, each dialogue turn was divided (segmented) into utterances such that each one corresponds to a unique DA label. An example of the segmentation and annotation of two turns of a dialogue can be seen in Figure 1

## 3 Dirichlet Process Mixture Models

This paper present a Dirichlet Process Mixture Model (DPMM) (Maceachern and Müller, 1998; Escobar and West, 1995; Antoniak, 1974) for the

Level	Labels
First	Opening, Closing, Confirmation, Undefined, Not-understood, Waiting, Consult, Acceptance, Rejection
Second	Departure-hour, Arrival-hour, Fare, Origin, Destination, Day, Train-Type, Service, Class, Trip-time
Third	Departure-hour, Arrival-hour, Fare, Origin, Destination, Day, Train-Type, Service, Class, Trip-time, Order-number, Number-trains, Trip-type

Table 1: Set of dialogue act labels used in the Dihana corpus

automatic, unsupervised clustering of the utterances in the Dihana corpus. This approach treats each utterance as a *bag of words* (i.e. an unordered collection of words) (Sebastiani, 2002). Utterances are clustered according to the relative counts of word occurrences that they contain so that utterances with similar histograms of word counts will, in general, appear in the same cluster.

Bayesian methods for unsupervised data clustering divide into parametric and nonparametric approaches. Parametric approaches to clustering such as Finite Bayesian Mixture Models (McLachlan and Peel, 2000) require prior estimation of the number of clusters that are expected to be found in the data. However, it is not always possible to know this in advance and often it is necessary to repeat a modelling experiment many times over a range of choices of cluster numbers to find an optimal number of clusters. Sub-optimal choices for the number of clusters can lead to a degradation in the generalisation performance of the model. Nonparametric approaches to mixture modelling, on the other hand, do not require prior estimates of the number of clusters in the data; this is discovered automatically as the model clusters the data. Dirichlet Processes offer one approach to developing Bayesian nonparametric mixture models. The remainder of this section briefly introduces DPMMs, beginning with a brief look at finite Bayesian mixture models which will serve as useful background for presenting the Chinese Restaurant Process, the Dirichlet Process paradigm used in this paper.

Speaker	Utterance	Transcription		
		Level 1	Level 2	Level 3
S	S1	Welcome to the railway information system. How may I help you?		
		Opening	Nil	Nil
U	U1	Could you tell me the departure times from Valencia		
		Question	Departure-hour	Origin
	U2	to Madrid .		
		Question	Departure-hour	Destination

Figure 1: An example of some turns from an annotated dialogue of DIHANA corpus.

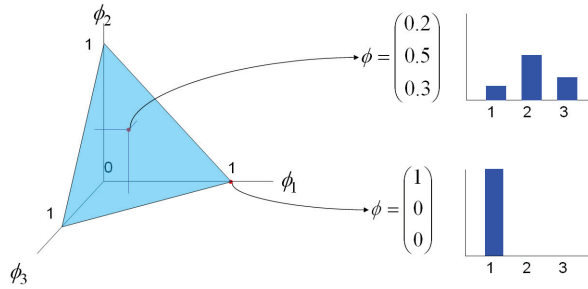


Figure 2: A 3-simplex with two example points and the corresponding distributions

### 3.1 Finite Bayesian Mixture Models

A Dirichlet distribution is defined as a *measure on measures*. Specifically, a Dirichlet distribution defines a probability measure over the  $k$ -simplex. The  $k$ -simplex is a convex hull constructed so that each point on the surface of the simplex describes a probability distribution over  $k$  outcomes:

$$Q_k = \{(x_1, \dots, x_k) : x_i \geq 0 \\ \forall i \in \{1 \dots k\}, \sum_{i=1}^k x_i = 1\}$$

Figure 2 shows a 3-simplex with two example points and the corresponding distributions. The Dirichlet distribution places a probability measure over the  $k$ -simplex so that certain subsets of points on the simplex (i.e. certain distributions) have higher probabilities than others (Figure 3). The probability measure in the Dirichlet is parameterised by a set of positive, non-zero concentration constants  $\alpha = \{\alpha_1, \dots, \alpha_k : \alpha_i > 0\}$ , written  $Dirichlet_k(\alpha_1, \dots, \alpha_k)$ . The effects of different values of  $\alpha$  for the 3-simplex are shown in Figure 3.

The probability density function of the Dirichlet

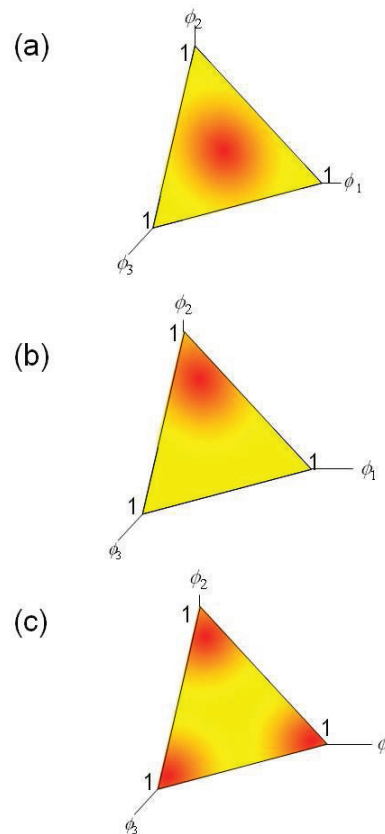


Figure 3: Three example Dirichlet Distributions over the 3-simplex with darker regions showing areas of high probability: (a) Dirichlet(5,5,5), (b) Dirichlet(0.2, 5, 0.2), (c) Dirichlet(0.5,0.5,0.5).

distribution is given by:

$$\begin{aligned} \text{Dirichlet}_k(\alpha_1, \dots, \alpha_k) &= f(x_1, \dots, x_k; \alpha_1, \dots, \alpha_k) \\ &= \frac{\Gamma(\sum_{i=1}^k \alpha_i)}{\prod_{i=1}^k \Gamma(\alpha_i)} \prod_{i=1}^k x_i^{\alpha_i-1} \end{aligned}$$

where  $\Gamma(x)$  ( $= \int_0^\infty t^{(x-1)} e^{-t} dt$ ) extends the factorial function to the real numbers. Since a draw from a Dirichlet distribution (written  $\beta \sim \text{Dirichlet}_k(\alpha)$ ) gives a distribution, a Dirichlet can be used as the prior for a Bayesian finite mixture model:

$$\beta \sim \text{Dirichlet}_k(\alpha_1, \dots, \alpha_k)$$

$\beta$  is a distribution over the  $k$  components  $\phi$  of the finite mixture model. Each component  $\phi_{z_i}$  is drawn from a base measure  $G_0$  ( $\phi_{z_i} \sim G_0$ ). The choice of distribution  $G_0$  depends on the nature of the data to be clustered; with data that is represented using the *bag of words* model,  $G_0$  must generate distributions over the word vocabulary. Hence the Dirichlet distribution is an appropriate choice in this case:

$$\phi_{z_i} \sim \text{Dirichlet}_v(\alpha_1, \dots, \alpha_v)$$

where  $v$  is the size of the vocabulary.

For each data point (utterance)  $x_i$  a component  $\phi_{z_i}$  is selected by a draw  $z_i$  from the multinomial distribution  $\beta$ :

$$z_i \sim \text{Multinomial}_k(\beta)$$

A suitable distribution  $F(\phi_{z_i})$  is then used to draw the data point (utterance). In the bag of words model, the multinomial distribution is used to draw the words for each data point  $x_i$ :

$$x_i \sim \text{Multinomial}_v(\phi_{z_i})$$

A small example will illustrate this generative process. Imagine that there are just two types of utterances with a vocabulary consisting simply of the words A, B and C. A finite Bayesian mixture model in this case would first draw  $\beta$  from a suitable Dirichlet distribution (e.g.  $\beta \sim \text{Dirichlet}_2(0.5, 1)$ ) as, for example, is shown in Figure 4(a). Next the two components  $\phi_{z_1}$  and  $\phi_{z_2}$  would be drawn from a suitable base distribution  $G_0$  (e.g.  $\phi_{z_1} \sim \text{Dirichlet}_3(1, 0.5, 0.5)$  and  $\phi_{z_2} \sim \text{Dirichlet}_3(0.5, 0.5, 1)$ , see Figure 4(b) and 4(c)). In this case,  $\phi_{z_1}$  will tend to generate

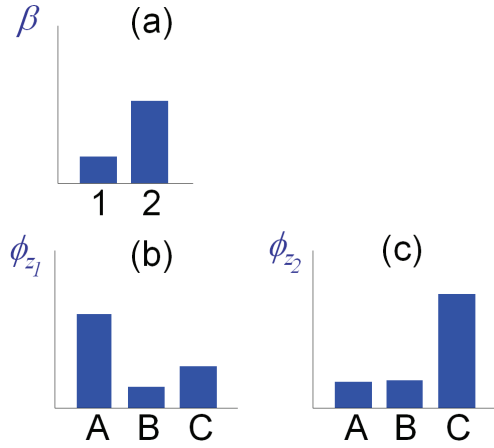


Figure 4: An example finite Bayesian mixture model. (a) The prior distribution over components  $\phi_{z_1}$  (b) and  $\phi_{z_2}$  (c)

utterances containing more occurrences of word A than B or C, whilst  $\phi_{z_2}$  will tend to generate utterances with more C's than A's or B's. A component  $z_i$  is then selected for each utterance ( $z_i \sim \text{Multinomial}_k(\beta)$ ). Note that in this example, the distribution  $\beta$  would lead to more utterances generated by  $\phi_{z_2}$  than by  $\phi_{z_1}$ . Suppose that five utterances are to be generated by this model and that the components for each utterance are  $z_1 = 1, z_2 = 2, z_3 = 2, z_4 = 1$  and  $z_5 = 2$ . The words in each utterance are then generated by repeated draws from the corresponding component (e.g.  $x_1 = ACAAB, x_2 = ACCBCC, x_3 = CCC, x_4 = CABAAC$  and  $x_5 = ACC$ ).

### 3.2 Dirichlet Processes

A Dirichlet Process can be thought of as an extension of a Dirichlet distribution where the dimensions of the distribution are infinite. The problem with the infinite dimension Dirichlet distribution, though, is that its probability mass would be distributed across the whole of the distribution. However, in most practical applications of mixture modelling there will be a finite number of clusters. The solution is to have a process which will tend to place most of the probability mass at the beginning of the infinite distribution, thereby making it possible to assign probabilities to clusters without restricting the number of clusters available. The GEM *stick breaking* construction (the name comes from the first letters of Griffiths, Engen and McCloskey (Pitman, 2002)) achieves precisely this (Pitman and Yor, 1997). Starting with

a stick of unit length, random portions  $\beta'_k$  are repeatedly broken off the stick, with each part that is broken off representing the proportion of probability assigned to a component:

$$\beta'_k \sim \text{Beta}(1, \alpha) \quad \beta_k = \prod_{i=1}^{k-1} (1 - \beta'_i) \cdot \beta'_k$$

The Dirichlet Process mixture model can now be specified as:

$$\beta \sim \text{GEM}(\alpha) \quad \phi_{z_i} \sim G_0 \quad z_i \in (1 \dots \infty) \\ z_i \sim \text{Multinomial}(\beta) \quad x_i \sim F(\phi_{z_i})$$

### 3.3 Chinese Restaurant Process

The Chinese Restaurant Process (CRP) is a popular Dirichlet Process paradigm that has been successfully applied to many clustering problems. In the CRP, one is asked to imagine a Chinese restaurant with an infinite number of tables. The customers enter the restaurant and select, according to a given distribution, a table at which to sit. All the customers on the same table share the same dish. In this paradigm, the tables represent data clusters, the customers represent data points ( $x_i$ ) and the dishes represent components ( $\phi_z$ ). As each customer (data point) enters the restaurant the choice of which table (cluster) and therefore which dish (component) is determined by a draw from the following distribution:

$$\phi_i | \phi_1, \dots, \phi_{i-1} \sim \frac{1}{(\alpha + i - 1)} \left( \sum_{j=1}^{i-1} \delta_{\phi_j} + \alpha G_0 \right)$$

where  $\alpha$  is the concentration parameter for the CRP. The summation over the  $\delta_{\phi_j}$ 's counts the number of customers sat at each of the occupied tables. The probability of sitting at an already occupied table, therefore, is proportional to the number of customers already sat at the table, whilst the probability of starting a new table is proportional to  $\alpha G_0$ . Figure 5 illustrates four iterations of this initial clustering process.

Once all the customers (data points) have been placed at tables (clusters), the inference process begins. The posterior  $p(\beta, \phi, z | \mathbf{x})$  cannot be calculated exactly, but Gibbs sampling can be used. Gibbs sampling for the CRP involves iteratively removing a randomly selected customer from their table, calculating the posterior probability distribution across all the occupied tables together with a potential new table (with a randomly drawn dish,

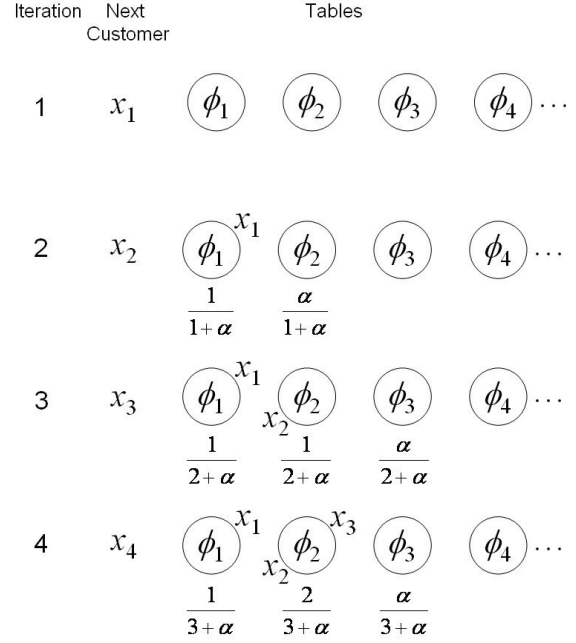


Figure 5: The first four steps of the initial clustering process of the CRP. The probability distribution over the tables is also shown in each case.

i.e. component), and making a draw from that distribution to determine the new table for that customer. The posterior distribution across the tables is calculated as follows:

$$\phi_i | \phi_1, \dots, \phi_{i-1}, \mathbf{x} \\ \sim \frac{1}{B} \left( \sum_{j=1}^{i-1} \delta_{\phi_j} p(x_i | \phi_j) + \alpha G_0 p(x_i | \phi_i) \right)$$

where  $B = \alpha p(x_k) + \sum_{j=1}^{i-1} p(x_i | \phi_j)$  is the normalising constant. After a predetermined number of samples, the dish (component) of each occupied table is updated to further resemble the customers (data points) sitting around it. In the *bag of words* approach used here, this involves converting the histogram of word counts in each customer (utterance) sitting at the table into an empirical distribution  $\mathcal{H}(x_i)$ , taking the average of these empirical distributions and modifying the dish (component) to further resemble this distribution:

$$\phi_i = \phi_i + \frac{\mu}{m_i} \sum_{j=1}^{m_i} \mathcal{H}(x_j)$$

where  $\mu$  ( $0 \leq \mu < 1$ ) is the learning constant and  $m_i$  is the number of customers around

table  $i$ . The inference process continues to iterate between Gibbs sampling and updating the table dishes (components) until the process converges. Convergence can be estimated by observing  $n$  consecutive samples in which the customer was returned to the same table they were taken from.

#### 4 Results

The CRP with Gibbs sampling was used to cluster both user and system utterances from the 900 dialogues in the Dihana corpus. Each utterance is treated as an independent bag of words where all information about the dialogue that it came from and the context in which it was uttered is ignored during training. Intra-cluster and inter-cluster similarity measures were used to evaluate the resulting clusters. Intra-cluster similarity  $S'_i$  is calculated by averaging the Euclidean distance between every pair of data points in the cluster  $i$ :

$$S'_i = \frac{1}{2m_i} \sum_{i=1; j=1}^{m_i} |x_i - x_j|$$

Inter-cluster similarity  $S''$  is calculated by summing the Euclidean distance between the centroids of all pairs of clusters:

$$S'' = \sum_{i=1; j=1}^n |C_i - C_j|$$

where  $C_i$  is the centroid of cluster  $i$  and  $n$  is the number of clusters.

Two classification error measures were also used, one from the cluster (table) perspective  $E'$ , and the other from the perspective of the Dialogue Act (DA) annotations (first level) of the Dihana corpus  $E''$ . The cluster classification error of table  $i$  is calculated by summing up the occurrences of each DA on the table, finding the DA with the largest total and allocating that DA as the correct classification for that table  $D_i$ . The number of false positives  $f_i^p$  for that table is the count of all customers (utterances) with DA annotations not in  $D_i$ . The number of false negatives  $f_i^n$  is the count of utterances with label  $D_i$  that occur on other tables. The cluster classification error for table  $i$  is therefore:

$$E'_i = \frac{1}{n} (f_i^p + f_i^n)$$

The DA classification error  $E''_i$  measures how well DA  $i$  has been clustered, using the size of the

Cluster No.	Ans	Ask	Clo	Not	Rej	Und
1	1					5
4	2	91				2
9		2	1			9
12	7	161	1			1
13	273	26				8
14	<b>382</b>	12	1			5
15	6	1	<b>909</b>	1	<b>327</b>	22
17	47	39			1	1
18	73				1	3
19		1				4
20	131	115	1		3	1
22	270	29	3			3
23	135	8			2	2
25	83	31	1			4
28	247	16	1			4
29	349	6	1			12
33	13	3	5	1	4	<b>25</b>
41	202	45	1		2	3
46		4				1
49	6	251	1	<b>2</b>		4
51	124	<b>896</b>	1			12
53	45	477				10

Table 2: Clusters of user utterances, with the counts for each level 1 speech act. The largest cluster for each speech act is in bold. The abbreviations are: Und = Undefined, Ans = Answering, Ask = Asking, Clo = Closing, Rej = Rejection, Not = Not-understood.

DA class  $N_i^c$ , the size of the largest cluster of utterances from that DA class  $M_i^c$ , and the total number of utterances  $n$  in the corpus:

$$E''_i = \frac{1}{n} (N_i^c - M_i^c)$$

Table 6 summarises the results from three separate runs of the CRP, each increasing in number of epochs. It should be noted here that the Dihana corpus has 72 DA categories, so the ideal number of clusters discovered by the CRP would be 72. It should also be noted that given an initial random clustering, a good clustering algorithm will reduce intra-cluster similarity ( $\bar{S}$ ), increase inter-cluster similarity ( $S''$ ) and reduce the classification errors ( $\bar{E}'$  and  $\bar{E}''$ ).

Epochs (K)	No. Clusters	$\bar{S}'$	$S''$	$\bar{E}'$	$\bar{E}''$
0	70	99703.6	243.74	0.05303	0.00979
1000	44	14975.4	217.56	0.01711	0.00385
1500	54	10093.7	336.15	0.01751	0.00435

Figure 6: The results from three separate runs of the CRP on utterances from the Dihana corpus. Cluster similarity measures and classification error values are shown after 0 (i.e. random clustering), 1000K, and 1500K epochs.  $\bar{S}'$ ,  $\bar{E}'$  and  $\bar{E}''$  are averaged values.

Level 1	Level 2	Cluster No.
Answering	Day	14
	Destination	22
	Fare	29
	Departure-hour	28, 41
Asking	Departure-hour,Fare	4
	Train-type	12
	Fare	49
	Departure-hour	51, 53

Table 3: Clusters that have specialised on level 1 and level 2 annotations.

## 5 Discussion

The first row of the table in Figure 6 shows the cluster similarity measures and classification errors after 0 epochs of the inference procedure (i.e. for a random clustering of utterances). This gives a baseline for the measures and error values used in subsequent runs. The second row of values shows the results after a run of 1000K epochs of the inference procedure. This run finds only 44 clusters but has a much lower value for  $\bar{S}'$  than was found in the random clustering, showing a significant increase in the similarity between utterances within each cluster. Surprisingly, the value for  $S''$  is also reduced, showing that the differentiation between the clusters formed at this stage is even lower than there was with the random clustering.  $\bar{E}'$  and  $\bar{E}''$  show suitable reductions indicating that the classification errors are being reduced by the inference process. The third row of values show that after 1500K epochs 54 clusters have been found, intra-cluster similarity is increased beyond that for the random clustering, but the classification errors remain essentially the same as for the 1500K run.

Although the 1500K epoch run found only 54 clusters, it was able to clearly distinguish between system and user utterances: with 30 clusters containing system utterances only, 22 clusters con-

taining user utterances only and 2 clusters containing instances of both. Given that the system utterances in the Dihana corpus are generated from a restricted set of sentences, it is not surprising that these were easy to cluster and differentiate from user utterances. However, the CRP was also able to cluster user utterances well, which is more of a challenge. Table 2 shows the clusters that have specialised on user utterances, with the counts of the level 1 annotations in each case. The largest cluster for each level 1 annotation is shown in bold typeface. From here it can be seen that cluster 15 has specialised on both *Closing* and *Rejection*. It is not surprising that these fall within the same cluster since the words used in each are often the same (e.g. “No thank you” can act as either a closing statement or a rejection statement). Clusters 14, 22, 29, 28 and 41 have specialised to the Answering annotation, whilst clusters 4, 12, 49, 51 and 53 have specialised to Asking. Table 3 shows how each of these clusters have specialised to level 2 annotations. Cluster 14, for example, specialises on the Answering:Day pair, whilst 22 specialises on Answering:Destination pair.

These initial results show that, at least for the Dihana corpus, the DPMM can successfully cluster utterances into Speaker, Level 1, and Level2 classes. Whilst this looks promising, it must be acknowledged that the Dihana corpus is restricted to train service inquiries and it remains unclear whether this approach will generalise to other dialogue corpora with a broader range of topics and wider vocabularies. Future work will include investigating the use of ngrams of words, syntactic features, the DAs of previous utterances and experimentation with other corpora such as Switchboard (Godfrey et al., 1992).

## Acknowledgments

This work was funded by the Companions project ([www.companions-project.org](http://www.companions-project.org)) sponsored by the

European Commission as part of the Information Society Technologies (IST) programme under EC grant number IST-FP6-034434. We thank Jeff Bilmes (University of Washington) for many very helpful discussions about Dirichlet processes and their application.

## References

- Toine Andernach, Mannes Poel, and Etto Salomons. 1997. Finding classes of dialogue utterances with kohonen networks. In *In Daelemans*, pages 85–94.
- J.A. Andernach. 1996. A machine learning approach to the classification and prediction of dialogue utterances. In *Proceedings of the 2nd International Conference on New Methods in Language Processing*, pages 98–109.
- Charles E. Antoniak. 1974. Mixtures of dirichlet processes with applications to bayesian nonparametric problems. *The Annals of Statistics*, 2(6):1152–1174.
- J.L. Austin. 1962. *How to do things with words*. Oxford: Clarendon Press.
- Michael D. Escobar and Mike West. 1995. Bayesian density estimation and inference using mixtures. *Journal of the American Statistical Association*, 90(430):577–588.
- Sheila Garfield and Stefan Wermter. 2006. Call classification using recurrent neural networks, support vector machines and finite state automata. *Knowl. Inf. Syst.*, 9(2):131–156.
- J. J. Godfrey, E. C. Holliman, and J. Mcdaniel. 1992. SWITCHBOARD: telephone speech corpus for research and development. In *Proc. ICASSP*, volume 1, pages 517–520 vol.1.
- Gang Ji and J. Bilmes. 2005. Dialog act tagging using graphical models. In *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on*, volume 1, pages 33–36.
- J.M.Benedí, E.Lleida, A. Varona, M.J.Castro, I.Galiano, R.Justo, I. López, and A. Miguel. 2006. Design and acquisition of a telephone spontaneous speech dialogue corpus in spanish: Dihana. In *Fifth International Conference on Language Resources and Evaluation (LREC)*, pages 1636–1639, Genova, Italy, May.
- S. Keizer, M. Gasic, F. Mairesse, B. Thomson, K. Yu, and S. Young. 2008. Modelling user behaviour in the his-pomdp dialogue manager. In *IEEE SLT*, pages 121–124, Dec.
- Steven N. Maceachern and Peter Müller. 1998. Estimating mixture of dirichlet process models. *Journal of Computational and Graphical Statistics*, 7(2):223–238.
- C. D. Martínez-Hinarejos, J. M. Benedí, and R. Granell. 2008. Statistical framework for a spanish spoken dialogue corpus. *Speech Communication*, 50:992–1008.
- Geoffrey Mclachlan and David Peel. 2000. *Finite Mixture Models*. Wiley Series in Probability and Statistics. Wiley-Interscience, October.
- Gabriel Murray, Steve Renals, Jean Carletta, and Johanna Moore. 2006. Incorporating speaker and discourse features into speech summarization. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 367–374, Morristown, NJ, USA. Association for Computational Linguistics.
- J. Pitman and M. Yor. 1997. The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. *Annals of Probability*, 25(2):855–900.
- J. Pitman. 2002. Combinatorial stochastic processes.
- Fabrizio Sebastiani. 2002. Machine learning in automated text categorization. *ACM Comput. Surv.*, 34(1):1–47, March.
- Andreas Stolcke, Noah Coccaro, Rebecca Bates, Paul Taylor, Carol Van Ess-Dykema, Klaus Ries, Elizabeth Shriberg, Daniel Jurafsky, Rachel Martin, and Marie Meteer. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Comput. Linguist.*, 26(3):339–373.
- E. Zovato and J. Romportl. 2008. Speech synthesis and emotions: a compromise between flexibility and believability. In *Proceedings of Fourth International Workshop on Human-Computer Conversation*, Bellagio, Italy.