The Hierarchical Beta-Bernoulli Process as Out-of-Scope Query Detector

The Hierarchical Beta-Bernoulli Process as Out-of-Scope Query Detector

Marco Dalla Pria[a] and Silvia Montagna[a]

[a]Università degli Studi di Torino, C.so Unione Sovietica 218/bis, Torino;
marco.dallapria@unito.it, silvia.montagna@unito.it

Abstract

Task-oriented dialog systems are computer systems that interact with humans in natural language. The system receives a query, converts the sequence of words into a semantic representation to be used by the dialog manager, decides the best response for the user, and manages the task. Occasionally, the system may receive an *out-of-scope* query, namely, a query that falls outside the range of the system's capabilites. In this work, we focus on out-of-scope query prediction, and show how the hierarchical Beta-Bernoulli process outperforms state-of-the-art machine learning classifiers.

Keywords: Hierarchical Beta-Bernoulli processes, nonparametric Bayesian modelling, task-oriented dialog systems, out-of-scope query prediction

1 Introduction

The increasing sophistication of machine learning algorithms in the last years has led to a revolution in task-oriented dialog systems: nowadays people can ask Amazon's Alexa what is their bank account balance while they are cooking, and will get a satisfactory answer from her. Any dialog system is designed to support a fixed number of intents only. For example, a task-driven system designed to support personal finance queries cannot answer the question "What is the weather like tomorrow?". Queries falling outside the range of intents which the dialog system is designed to work upon are defined *out-of-scope* queries (hereafter, OOS). Correctly identifying that a query is OOS is of paramount importance for the system to avoid performing wrong actions. Thus far, however, little attention has been given to evaluating the performance of state-of-the-art, dialog system machine learning classifiers in OOS prediction. An exception is given by [1], who evaluate and compare the performance of a range of benchmark classifier models focusing on OOS prediction relying on a novel dataset. Whilst the tested models work well in predicting known intents, the authors show that all methods struggle with identifying OOS queries.

The hierarchical Beta-Bernoulli process is a well known Bayesian nonparametric process that has shown good performance in document classification tasks [2]. Informally speaking, documents are a collection of words, thus queries themselves can be seen as documents. However, unlike long-text documents, the fact that queries consist of only a few words is an obstacle towards distinguishing an OOS from an in-scope query, which indeed become indistinguishable if a couple of key words were removed from the OOS query; see Table 1. In face of these difficulties, we believe that the flexibility of a nonparametric model could be instrumental in detecting OOS queries. In this work, we fit a Beta-Bernoulli process to the classification data in [1], and show that it outperforms benchmark machine learning classifiers in OOS query prediction.

The remainder of this paper is organised as follows. In Section 2, we introduce the discrete form of the hierarchical Beta-Bernoulli process [2], which we leverage on in this work, and explain how this process can serve as nonparametric Bayesian prior in document classification tasks. Section 3 illustrates the inferential procedure leading to the classification of an unlabelled document. In Section 4, we analyse the dataset in [1] by means of the hierarchical Beta-Bernoulli process, and Section 5 presents conclusions and directions for future work.

2 Methods

The (discrete) Beta process, denoted $BP(c, B_0)$, is a Lévy process over a space Ω whose Lévy intensity is defined by:

$$\nu(d\omega, dp) = \sum \text{Beta}(cq_i, c(1-q_i))(dp)\delta_{\omega_i}(d\omega)$$

where the *base measure* $B_0 = \sum_i q_i \delta_{\omega_i}$ is discrete, the positive real constant c is the *concentration parameter*, and the total mass $\gamma = B_0(\Omega)$ of the base measure is called *mass parameter*; γ is required to be finite. Note that each q_i must lie in $(0, 1)$ in order for the Lévy intensity to be well defined.

The (discrete) Bernoulli process, denoted $BeP(B)$, is the Lévy process characterised by the Lévy intensity:

$$\mu(d\omega, dp) = \sum \text{Bernoulli}(p_i)(dp)\delta_{\omega_i}(d\omega)$$

where the base measure $B = \sum_i p_i \delta_{\omega_i}$ is discrete. Note that the masses p_i must lie in $(0, 1]$ in order for the Lévy intensity to be well defined. The probability of a particular realisation of a $BeP(B)$ with B discrete is:

$$\mathbb{P}\left(X = \{\omega_{j_1}, \omega_{j_2}, ..., \omega_{j_K}\}\right) = \prod_{k=1}^{K} \mathbb{P}\left(\omega_{j_k} \in X\right) = \prod_{k=1}^{K} \int_{[0,1]} \text{Ber}\left(p_{j_k}\right)(dp) = \prod_{k=1}^{K} p_{j_k}$$

Given a discrete base measure B_0 and a positive real constant c, we can combine the two processes above and obtain the Beta-Bernoulli process (BBp):

$$B \sim BP(c, B_0), \quad X|B \sim BeP(B)$$

and conjugacy holds, that is, given n conditionally *iid* samples $X_1, ..., X_n|B \sim BeP(B)$,

$$B|X_1, ..., X_n \sim BP\left(c + n, \frac{c}{c+n}B_0 + \frac{1}{c+n}\sum_{i=1}^{n} X_i\right).$$

Indeed, by independence over disjoint subsets of Ω and by the discreteness of B_0, inference can be carried out separately for each atom ω_i. The result follows from the well-known Beta-Binomial conjugacy.

We follow [2] and embed the BBp into a hierarchical model, leading to the hierarchical BBp (hBBp):

$$B \sim BP(c_0, B_0), \quad B_1, ..., B_J|B \sim BP(c_j, B), \quad X_{1,j}, ..., X_{n_j,j}|B_j \sim BeP(B_j) \quad j = 1, ..., J$$

An helpful analogy to better understand the model is the following. Consider a corpus X of n documents, each of which is associated with one of J subjects, such that there are n_j documents $X_{1,j}, ..., X_{n_j,j}$ belonging to topic j. Any given word in the English vocabulary is more or less likely to appear in a document depending on the subject: some technical terms will be exclusively used in certain domains, other terms are likely to appear in affine subjects, and some other will be ubiquitous. Let us model a document as a subset of words in some vocabulary, or, equivalently, as a binary vector whose components are indexed by the words in the vocabulary. At a given component, $0/1$ stands for the absence/presence of that word in the document, respectively.

Taking the underlying space $\Omega = \{\omega_1, \omega_2, \omega_3, ...\}$ as the vocabulary, which is potentially infinite (think of every possible misspelled word) but countable, let $B_0 = \sum_i b_0(\omega_i)\delta_{\omega_i}$ be a discrete measure over Ω such that $b_0(\omega_i) \in (0,1)\ \forall i$, and let c_0 be a real constant. Then, B is also discrete, and $B(\omega_i) \sim$ Beta$(c_0 b_0(\omega_i), c_0(1-b_0(\omega_i))\ \forall i$. In this setting each B_j is discrete, supported by Ω and such that $B_j(\omega_i) \sim$ Beta$(c_j B(\omega_i), c_j(1-B(\omega_i))$. Then, $B_j(\omega_i)$ gives the probability that word ω_i appears in a document with topic j, while $B(\cdot)$ encodes the sharing of information between topics. Parameters c_0, c_j encode the semantic richness overall and within each topic, respectively. Specifically, if c_j is small then documents of subject j contain the same few words, whereas if c_j is large documents of topic j are dissimilar. Similarly, a small c_0 induces a lot of shared terms among the different topics, while a large c_0 means that each topic has its own set of specialised terms. Concentration parameters c_0, c_j's will play a key role for the performance of the hBBp.

3 Inference

We rely on the hBBp presented above to assign a topic to an unlabelled document X_{new}. Here we explain the inferential procedure.

Because Lévy processes have independent increments over disjoint sets, it is legitimate to carry out inference separately for the set of observed words (the bag-of-words obtained by the union of all the words in every document) and for its complement (the words in Ω which never appeared in any document of the corpus). Let $\Omega_{\text{obs}} \subset \Omega$ be the collection of unique words that appeared at least once in some document, and let Ω_0 be its complement. The procedures discussed in this Section are summarised in Algorithm 1 and Algorithm 2.

Inference over Ω_{obs} Fix a $\omega \in \Omega_{\text{obs}}$ and define $x_{i,j} = X_{i,j}(\omega)$, $x = \{X_{i,j}(\omega) | j \leq J; i \leq n_j\}$, $b_j = \{B_j(\omega) | j \leq J\}$, $b = B(\omega)$, and $m_j = \sum_{i=1}^{n_j} x_{i,j}$. It is possible to show that:

$$\mathbb{P}\left(x_{n_j+1,j} = 1 \middle| x\right) = \mathbb{E}\left[\mathbb{E}\left[b_j | b, x\right] | x\right] = \mathbb{E}\left[\frac{c_j b + m_j}{c_j b + m_j + c_j(1-b) + (n_j - m_j)} \middle| x\right] = \frac{c_j \mathbb{E}\left[b|x\right] + m_j}{c_j + n_j}.$$

The posterior expectation $\mathbb{E}[b|x]$ is not available in closed form analytically, thus we will rely on its Monte Carlo approximation. In particular, it is possible to show that the density of $b|x$ is bounded above by the unnormalised density of a Gamma(α, β), where

$$\alpha = c_0 b_0 + \sum_{j=1}^{J} 1(m_j > 0),\ \beta = \frac{c_0(1-b_0)-1}{1-b^*} - \sum_{j=1}^{J} \sum_{i=1}^{m_j - 1} \frac{c_j}{c_j b^* + i} + \sum_{j=1}^{J} \sum_{i=0}^{n_j - m_j - 1} \frac{c_j}{c_j(1-b^*) + i}$$

where b^* is the mode of the density of $b|x$, which can be easily obtained by any appropriate numerical optimisation method, being such a density concave in $(0,1)$. Relying on the Gamma(α, β) as proposal distribution, we generate T samples $b_1, b_2, \ldots, b_T$ via Metropolis-Hastings and then approximate $\mathbb{E}[b|x]$ via the empirical mean. After some testing, we realised that the Gamma approximation is very precise: $20 \leq T \leq 30$ samples yield a satisfactory approximation in the application discussed hereafter.

Inference over Ω_0 Fix some $\omega \in \Omega_0$. Adapt the notation used above to this new ω. As before, we would like to compute the probability $\mathbb{P}\left(x_{n_j+1,j} = 1 \middle| x = 0\right)$, but this turns out to be more challenging. Given W words $\{\omega_1, ..., \omega_W\} \subset \Omega_0$, and defining $\lambda_j := \sum_{k=1}^{K} \frac{c_0 B_0(\Omega_0)}{c_0 + k - 1} p_{k,j}$,

$$\mathbb{P}\left(\{\omega_1, ..., \omega_W\} \subset X_{n_j+1,j} \middle| x = 0\right) \approx \text{Pois}\left(\lambda_j\right)(W) \prod_{i=1}^{W} b_0(\omega_i)$$

where $p_{k,j} = \mathbb{P}_k(x_{n_j+1,j} = 1, x = 0)$ and $\mathbb{P}_k$ is the probability over the slice of the hBBp corresponding to ω, and is approximated via simulation. The larger K, the better the approximation, which is in fact exact for $K \to \infty$. See Algorithm 2 and [2] for further details.

Combining the above, given a new document $X_{\text{new}} = \{\omega_1, ..., \omega_W\}$ whose subject has to be inferred, we compute

$$\mathbb{P}(X_{n_j+1,j} = X_{\text{new}}|X) \approx \prod_{\omega \in X_{\text{new}} \cap \Omega_{\text{obs}}} \frac{c_j \mathbb{E}\left[B(\omega)|X\right] + m_j}{c_j + n_j} \times \text{Pois}\left(\lambda_j\right)(W) \prod_{\omega \in X_{\text{new}} \cap \Omega_0} b_0(\omega).$$

We compute this probability for all topics $j = 1, \ldots, J$, and then assign X_{new} to the topic j^* that maximises the probability above.

Algorithm 1: HBBp training

```
Data: corpus $X$ of $n$ documents, $n_j$ documents for each topic $1 \leq j \leq J$ ; $\Omega$ ; $c_1, \ldots, c_J$ ; $B_0$
$\gamma \leftarrow$ mean number of unique words in a query ;
$B_0 \leftarrow \frac{B_0}{B_0(\Omega)}\gamma$ ;
$\Omega_{\text{obs}} \leftarrow$ unique words in the corpus ;
$F \leftarrow |\Omega_{\text{obs}}|$ ;
$c_0 \leftarrow$ solution of $c_0 = \frac{F-\gamma}{\gamma \log\left(\frac{c_0+n}{c_0+1}\right)}$ ;
for $\omega \in \Omega_{\text{obs}}$ do
    $M_{\omega,j} \leftarrow$ number of documents of topic $j$ having $\omega$, for $1 \leq j \leq J$
    $b^*_\omega \leftarrow$ mode of the posterior density of $B(\omega)|X(\omega)$
    $\alpha_\omega \leftarrow c_0 b_0(\omega) + \sum_{j=1}^{J} 1(M_{\omega,j} > 0)$, with $b_0(\omega) := 0$ if $\omega \notin \Omega$
    $\beta_\omega \leftarrow \frac{c_0(1-b_0(\omega))-1}{1-b^*_\omega} - \sum_{j=1}^{J} \sum_{i=1}^{M_{\omega,j}-1} \frac{c_j}{c_j b^*_\omega + i} + \sum_{j=1}^{J} \sum_{i=0}^{n_j - M_{\omega,j}-1} \frac{c_j}{c_j(1-b^*_\omega)+i}$
end
```

Algorithm 2: Document classification via hBBp

```
Input: New document $X_{\text{new}} = \{\omega_1, \ldots, \omega_W\}$ ; trained hBBp ; $T_1, T_2, K \in \mathbb{N}$
Output: Most likely topic $j$ s.t. $X_{\text{new}} = X_{n_j+1,j}$
for unique $\omega \in X_{\text{new}} \cap \Omega_{\text{obs}}$ do
    $b_1, \ldots, b_{T_1} \leftarrow$ Metropolis-Hastings with Gamma($\alpha_\omega$, $\beta_\omega$) proposal and target $B(\omega)|X(\omega)$
    $\bar{b} \leftarrow \frac{1}{T_1} \sum_{i=1}^{T_1} b_i$
    $P_{1,j} \leftarrow P_{1,j} \frac{c_j \bar{b} + M_{\omega,j}}{c_j + n_j}, \ \forall\, 1 \leq j \leq J$
end
for $1 \leq k \leq K$ do
    $b_1, \ldots, b_{T_2} \leftarrow$ sample from Beta(1, $c_0 + k - 1$)
    for $1 \leq j \leq J$ do
        $r_{i,j} \leftarrow \frac{c_j b_i}{c_j + n_j} \prod_{j'=1}^{J} \frac{\Gamma(c_{j'})\Gamma(c_{j'}(1-b_i)+n_{j'})}{\Gamma(c_{j'}(1-b_i))\Gamma(c_{j'}+n_{j'})}, \forall\, 1 \leq i \leq T_2$
        $p_{k,j} \leftarrow \frac{1}{T_2} \sum_{i=1}^{T_2} r_{i,j}$
    end
end
$\lambda_j \leftarrow \sum_{k=1}^{K} \frac{c_0 B_0(\Omega_0)}{c_0+k-1} p_{k,j}, \ \forall\, 1 \leq j \leq J$ ;
$P_{2,j} \leftarrow \text{Pois}(\lambda_j)(W) \times \prod_{\text{unique}\ \omega \in X_{\text{new}} \cap \Omega_0} b_0(\omega), \ \forall\, 1 \leq j \leq J$ ;
return $j^* \leftarrow 1 \leq j \leq J$ maximising $P_{1,j} \times P_{2,j}$
```

4 Data Analysis and Results

We fit the hBBp to the CLINC150[1] data. The dataset contains a training set made of 15000 in-scope queries, 100 for each of 150 intents, 100 OOS training queries, and a test set made of 4500 in-scope queries and 1000 OOS queries. A snapshot of the CLINC150 data is provided in Table 1.

Table 1: A snapshot of in-scope and OOS queries from the CLINC150 dataset.

Query	Intent
what is the temperature in costa mesa	weather
does france have their own version of a visa	international_visa
where can i pick up my w2 to do my taxes	w2
pay my gas bill from my saving account	pay_bill
what do i have on my calendar for march 2	calendar
who are some notable alumni of ucsd	OOS
when was nintendo created	OOS
when was the theory of evolution first considered	OOS
why do males want to be alpha	OOS
what are van gogh's best pieces	OOS

We consider as our space Ω the set of the most common words in Wikipedia[2], which contains more that 280000 terms. Despite its size, such a vocabulary does not contain many frequent terms appearing in the dataset. Indeed CLINC150 queries are full of misspelled words, symbols, numbers and proper names. However this is not an issue for the hBBp in that its nonparametric nature allows Ω to grow as the data is observed. Here the underlying assumption is that if ω is observed in a training query but it is not in Ω, then we treat it as if ω belongs to Ω with $b_0(\omega) = 0$. The prior distribution over such ω is improper, but becomes proper after the Bayesian update. Therefore, misspelled words have been retained in the dataset. Further, we did not remove stop words, which indeed appear to be informative predictors especially when appearing in clusters (e.g., "how would I...in...?", often appears under the intent "translate"), and no stemming has been applied.

Note that a test query might include a word that has never been observed in the training set and is also not present in Ω. Indeed, this is quite common, especially if the test query is OOS. Assuming $b_0(\omega) = 0$ is not a good choice in such case since this would translate into a zero probability of observing such query under every intent, and the classification would not be possible. To overcome this issue, we add a special out-of-training (OOT) feature to the vocabulary of Wikipedia's most common terms. Specifically, if a test query contains a word that has never been observed in the training set and is not present in Wikipedia's vocabulary, then such a word is mapped to OOT and is interpreted as a feature of the query at hand.

To choose an appropriate B_0, we rely on a power-law determined by the ranking in the list of Wikipedia's most frequent terms. Having shifted down in the ranking each word by one position (the most frequent word becomes the second most, the second most frequent becomes the third most, and so on) and having put OOS on top of the ranking in the first position, the chosen power-law is rank$^{-0.1}$, the exponent close to zero to avoid the tail from becoming too thin. The total mass of B_0 has also to be coherent with the data. One can show that $\gamma = B_0(\Omega)$ is the mean number of unique words per document, which amounts to 8.31 in CLINC150.

Choosing the concentration parameters is challenging. Besides c_0, which is computed as the fixed point solution of a real valued function (see Algorithm 1 and [2]), the crucial point is the choice of the concentration parameters associated to the 151 topics. Unfortunately, an exhaustive grid search for the optimal combination for these hyperparameters is unfeasible given our computational resources. After some trial and error, a tuning "by hand" led to choices giving a good trade-off between in-scope accuracy and OOS recall, and is the chosen setting leading to the results presented hereafter.

[1] https://github.com/clinc/oos-eval

[2] https://en.lexipedia.org/

Table 2: In-scope and out-of-scope performance comparison between the hierarchical Beta-Bernoulli process and benchmark machine learning methods in [1]: FastText, CNN, MLP, BERT neural networks; SVM, a linear support-vector classifier; Google's DialogFlow and Rasa's NLU conversational AIs.

Classifier	In-Scope Accuracy	Out-of-Scope Recall
FastText	89.0	9.7
SVM	91.0	14.5
CNN	91.2	18.9
DialogFlow	91.7	14.0
Rasa	91.5	45.3
MLP	93.5	47.4
BERT	**96.9**	40.3
Hierarchical Beta-Bernoulli process	86.5	**79.5**

A comparison with benchmark machine learning methods is displayed in Table 2, where results referring to models from FastText to BERT are taken from [1] (see [1] for details on these models). These results are promising: although slightly underperforming in terms of in-scope accuracy, the hBBp outperforms in terms of OOS recall, which is the goal of our application. Further, these results should be treated as preliminary results for our work and we expect both in-scope and OOS perfomance to improve with more accurate tuning of the model hyperparameters, as done for the machine learning competitors instead.

5 Conclusions

In this paper, we proposed a hierarchical Beta-Bernoulli process for OOS query prediction. The methodology outperforms state-of-the-art machine learning techniques used by task-based dialog systems, and its in-scope performance is in line with that of existing techniques. Moreover, it can handle misspelled words in a straightforward and appealing manner.

Possible future work could be the estimation of the number of different topics within the OOS class via Kingman's *Coalescent*, which plays the role of a nonparametric prior over the dendrogram governing the clustering structure of OOS queries.

References

[1] Larson, S., Mahendran, A., Peper, J. J., Clarke, C., Lee, A., Hill, P., Kummerfeld, J. K., Leach, K., Laurenzano, M. A., Tang, L., and Mars, J.: An Evaluation Dataset for Intent Classification and Out-of-Scope Prediction. In: *Proc. of the 2019 Conf. on Empir. Methods in Nat. Lang. Process. and the 9th Int. Jt. Conf. on Nat. Lang. Process. (EMNLP-IJCNLP).* Assoc. for Comput. Linguistics, 2019, pp. 1311–1316.

[2] Thibaux, R. and Jordan, M. I.: Hierarchical Beta Processes and the Indian Buffet Process. In: *Proc. of the Eleventh Int. Conf. on Artif. Intell. and Stat.* Vol. 2. Proc. of Mach. Learn. Res. PMLR, 2007, pp. 564–571.