# POWER LAW DISCOUNTING FOR N-GRAM LANGUAGE MODELS

*Songfang Huang, Steve Renals*

The Centre for Speech Technology Research
University of Edinburgh, Edinburgh EH8 9AB, United Kingdom
{s.f.huang, s.renals}@ed.ac.uk

**ABSTRACT**

We present an approximation to the Bayesian hierarchical Pitman-Yor process language model which maintains the power law distribution over word tokens, while not requiring a computationally expensive approximate inference process. This approximation, which we term power law discounting, has a similar computational complexity to interpolated and modified Kneser-Ney smoothing. We performed experiments on meeting transcription using the NIST RT06s evaluation data and the AMI corpus, with a vocabulary of 50,000 words and a language model training set of up to 211 million words. Our results indicate that power law discounting results in statistically significant reductions in perplexity and word error rate compared to both interpolated and modified Kneser-Ney smoothing, while producing similar results to the hierarchical Pitman-Yor process language model.

***Index Terms***— language model, smoothing, absolute discount, Kneser-Ney, Bayesian, Pitman-Yor, power law

## 1. INTRODUCTION

Smoothing is crucial when estimating a language model (LM), and a large number of methods have been proposed in the literature [1], including interpolated Kneser-Ney [2] and modified Kneser-Ney [1] smoothing which are generally regarded as the best approaches in practice. The Kneser-Ney approaches are based on absolute discounting, with lower order distributions reflecting the marginal constraints. In addition to exploring further constraints and more efficient algorithms [3], there has been a recent body of work in which the Kneser-Ney methods have been shown to approximate a hierarchical Bayesian language model which incorporates a non-parametric prior distribution, the Pitman-Yor process [4]. Our previous work [5] demonstrated the practical application of hierarchical Pitman-Yor process language models (HPYLM) to large vocabulary automatic speech recognition (ASR) of conversational speech in multiparty meetings, indicating that this model can offer consistent and significant reductions in perplexity and word error rate (WER), compared to both an interpolated Kneser-Ney LM (IKNLM) and a modified Kneser-Ney LM. (MKNLM). However estimation of an HPYLM is expensive, requiring sampling, which hinders it from wide application to large vocabulary ASR even with a parallel training algorithm [6].

One of the most remarkable statistical properties of word frequencies in natural language is the fact that they follow a power law distribution, i.e., $P(c_w = x) \propto x^{-d}$ where $c_w$ is the number of occurrences of word $w$ in a corpus and $d$ is a constant. The well-known Zipf's law equivalently states the fact in terms of the frequency ranking of words: a few outcomes have very high probability while most outcomes occur with low probability. It follows that a stochastic process, such as the Pitman-Yor process [7], that has the "rich-get-richer" capacity to generate a power law distribution is able to take advantage of this property for natural language modeling.

In this paper we briefly outline the HPYLM, which offers a hierarchical Bayesian approach to language modeling, with a non-parametric prior distribution. Since the HPYLM is a computationally expensive estimate, we present an approximation to it, that we call *power law discounting* in which the discounting parameters have a direct functional form and do not require approximate inference. Inference in power law discounting has a similar computational complexity to interpolated or modified Kneser-Ney. We evaluate this new approach to language model smoothing in terms of perplexity and WER in the domain of multiparty meetings, reporting results on the NIST RT06s evaluation data and on the AMI corpus. Our results indicate that power law discounting is a good approximation to the more computationally expensive HPYLM, and maintains statistically significant decreases in WER and perplexity compared with interpolated and modified Kneser-Ney smoothing. Finally we present some analysis of the power law discounting scheme, including a demonstration of the power law property, as well as an investigation of the effect of the discounting parameters.

## 2. HIERARCHICAL PITMAN-YOR PROCESS LANGUAGE MODELS

In a Bayesian language model a prior distribution is placed over the predictive probabilities of the LM, and the posterior distribution is inferred from the observed training data. The final predictive probability can then be estimated from the posterior by marginalizing out the latent variables and hyperparameters. In the HPYLM, Pitman-Yor processes are recursively placed as priors over the predictive probabilities in $n$-gram LMs, resulting a suffix tree hierarchy of Pitman-Yor process priors. The Pitman-Yor process $\mathrm{PY}(d, \theta, G_b)$ is a three parameter distribution over distributions, where $d$ is a discount parameter, $\theta$ a strength parameter, and $G_b$ a base distribution that can be understood as a mean of draws from $\mathrm{PY}(d, \theta, G_b)$.

The procedure for generating draws $\mathrm{G} \sim \mathrm{PY}(d, \theta, G_b)$ from a Pitman-Yor process can be described using the "Chinese Restaurant" metaphor. Customers enter a Chinese restaurant containing an infinite number of tables and seat themselves. The first customer sits at the first available table, while each of the subsequent customers sits either at an occupied table with probability proportional to the number of customers already sitting there $c_k - d$, or at a new unoccupied table with probability proportional to $\theta + dt_\bullet$, where $c_k$ is the number of customers sitting at table $k$ and $t_\bullet$ is the current

This work is supported by the European Union Projects FP6-033812 (AMIDA) and FP7-213845 (EMIME), and has made use of the resources provided by the Edinburgh Compute and Data Facility (ECDF, http://www.ecdf.ed.ac.uk/).

number of occupied tables. Goldwater *et al.* [8] demonstrated that a Pitman-Yor process is capable of producing a power law distribution with index $1 + d$ over the number of customers seated at each table . When the Pitman-Yor process is applied to language modeling, a restaurant corresponds to a context, customers to word tokens occurring after the context, and each table to a word type from the vocabulary.

Let $w$ and $w'$ be words, $\mathbf{u}$ be a context of length $n-1$, and $\pi(\mathbf{u}) = \mathbf{u}'$ be the context that is one word shorter than $\mathbf{u}$ with length $n-2$, such that $w'\mathbf{u}' = \mathbf{u}$. In the case of the HPYLM, we obtain the following expression for the predictive probability:

$$P(w|\mathbf{u},\mathcal{S},\Theta) = \frac{c_{\mathbf{u}w} - d_{|\mathbf{u}|}t_{\mathbf{u}w}}{\theta_{|\mathbf{u}|} + c_{\mathbf{u}\bullet}} + \frac{\theta_{|\mathbf{u}|} + d_{|\mathbf{u}|}t_{\mathbf{u}\bullet}}{\theta_{|\mathbf{u}|} + c_{\mathbf{u}\bullet}} P(w|\pi(\mathbf{u}),\mathcal{S},\Theta). \quad (1)$$

where $\mathcal{S}$ denotes latent variables and $\Theta = \{d_m, \theta_m : 0 \le m < n\}$ represents hyperparameters. If we set the discounting parameters $d_{|\mathbf{u}|} = 0$ for all $\mathbf{u}$, we obtain the hierarchical Dirichlet language model (HDLM) [9].

The overall predictive probability can be approximately obtained by collecting $I$ samples from the posterior over $\mathcal{S}$ and $\Theta$, and then averaging (1) to approximate the integral with samples:

$$P(w|\mathbf{u}) \approx \sum_{i=1}^{I} P(w|\mathbf{u},\mathcal{S}^{(i)},\Theta^{(i)})/I\,. \quad (2)$$

If we assume that the strength parameters $\theta_{|\mathbf{u}|} = 0$ for all $\mathbf{u}$, and restrict $t_{\mathbf{u}w}$ to be at most 1 (i.e., all customers representing the same word token should only sit on the same table together), then the predictive probability in (1) directly reduces to the predictive probability given by the IKNLM. We can thus interpret IKN as an approximate inference scheme for the hierarchical Pitman-Yor process language model [4].

The HPYLM has two principal advantages over interpolated and modified Kneser-Ney LMs. First the HPYLM utilizes the power law characteristic of natural language via Pitman-Yor processes for language modeling. Second the HPYLM is able to do more flexible absolute discounting, i.e., specified to each different context $\mathbf{u}$, via the discount parameter $d_{|\mathbf{u}|}t_{\mathbf{u}w}$ of the HPYLM and thus also improves over LMs based on a Dirichlet distribution or Dirichlet process prior. However, it is much more resource intensive to infer an HPYLM than the IKNLM or MKNLM for large vocabulary ASR, even with a parallel training algorithm [6].

## 3. POWER LAW DISCOUNTING LANGUAGE MODEL

In this section we introduce an efficient approximation to the HPYLM, in which $t_{\mathbf{u}w}$ the number of tables occupied by word $w$ in the restaurant corresponding to context $\mathbf{u}$ is approximated as a function of $c_{\mathbf{u}w}$, the count of occurrences of word $w$ following context $\mathbf{u}$. This is in contrast to the HPYLM in which this parameter is obtained using an approximate inference scheme based on Markov Chain Monte Carlo sampling.

We simplify the notations for $d_{|\mathbf{u}|}$ and $\theta_{|\mathbf{u}|}$ by ignoring the subscript $|\mathbf{u}|$. In the HPYLM, the $c_{\mathbf{u}w}$ occurrences of a word $w$ following context $\mathbf{u}$ is discounted by $dt_{\mathbf{u}w}$, i.e., $\hat{c}_{\mathbf{u}w} = c_{\mathbf{u}w} - dt_{\mathbf{u}w}$. The approximate inference of $t_{\mathbf{u}w}$ is time and memory intensive. However, the expected number of tables $\mathbb{E}(t_{\mathbf{u}\bullet})$ in a Pitman-Yor process used in the HPYLM follows a power law growth with $c_{\mathbf{u}\bullet}$ where $\bullet$ denotes the marginal operation [10]. Based on this observation, we therefore propose a power law discounting LM (PLDLM) which smoothes $n$-grams as follows:

$$d = \frac{n_1}{n_1 + 2n_2} \quad (3)$$

$$t_{\mathbf{u}w} = f(c_{\mathbf{u}w}) = c_{\mathbf{u}w}^{d} \quad (4)$$

$$t_{\mathbf{u}\bullet} = \sum_w t_{\mathbf{u}w} = \sum_w c_{\mathbf{u}w}^{d} \quad (5)$$

$$P^{\text{PLD}}(w|\mathbf{u}) = \frac{\max(c_{\mathbf{u}w} - dt_{\mathbf{u}w}, 0)}{\theta + c_{\mathbf{u}\bullet}} + \frac{\theta + dt_{\mathbf{u}\bullet}}{\theta + c_{\mathbf{u}\bullet}} P^{\text{PLD}}(w|\pi(\mathbf{u})) \quad (6)$$

The parameter $d$ corresponds to the traditional discount parameter in the IKNLM for $(|\mathbf{u}|+1)$-grams, $n_1$ and $n_2$ are the total number of $n$-grams with exactly one and two counts and $\theta$ is the strength parameter. The estimate of $t_{\mathbf{u}w}$ in (4) significantly simplifies the model while maintaining the most important part of the HPYLM: the power law characteristic. Another key issue for the PLDLM is the modified counts $c_{\mathbf{u}'w}$ for lower order $(n-1)$-grams of context $\mathbf{u}'$, as shown in (8).

$$\begin{cases} t_{\mathbf{u}w} = 0 & \text{if } c_{\mathbf{u}w} = 0; \\ 1 \le t_{\mathbf{u}w} < c_{\mathbf{u}w} & \text{if } c_{\mathbf{u}w} > 0; \end{cases} \quad (7)$$

$$c_{\mathbf{u}'w} = \sum_{\mathbf{u}:\pi(\mathbf{u})=\mathbf{u}'} t_{\mathbf{u}w} = \sum_{w'} t_{w'\mathbf{u}'w} \quad (8)$$

The strength parameter $\theta$ can be estimated using the same technique as that for the HPYLM, i.e., a sampling method based on auxiliary variables [10]. Since the model is insensitive to $\theta$, as demonstrated in Section 6, we can alternatively set the values empirically, or simply ignore $\theta$ in (6). In our experiments in section 5 we set $\theta = 0$.

We can decide on the number of discount parameters, denoted as $p$, to be used in the PLDLM, i.e., using one for each for $c_{\mathbf{u}w} = 1, 2, \ldots, p-1$ and another for all $c_{\mathbf{u}w} \ge p$. The PLDLM exactly reduces to the IKNLM when $p = 1$. The PLDLM with $p = 3$, $\text{PLD}_3$, is directly comparable to the MKNLM since both have three free discount parameters, except that the PLDLM takes a more straightforward form ($dc_{\mathbf{u}w}^{d}$) than the MKNLM [1]. The amount of discount in the PLDLM is a function of counts $c_{\mathbf{u}w}$, which grows slowly as the count increases.

## 4. MARGINAL CONSTRAINTS

Kneser and Ney [2] demonstrated the importance of preserving the marginal constraints in language modeling. Following [1, 10], we show that the PLDLM satisfies the marginal constraints when the strength parameter $\theta = 0$, and we use the predictive probability in (6) and modified counts in (8):

$$\frac{c_{\mathbf{u}'w}}{c_{\mathbf{u}'\bullet}} = \sum_{w'} \frac{c_{w'\mathbf{u}'}}{c_{\bullet\mathbf{u}'}} P^{\text{PLD}}_{w'\mathbf{u}'}(w) \quad (9)$$

$$\begin{aligned} c_{\mathbf{u}'w} &= \sum_{w'} c_{w'\mathbf{u}'} \left[ \frac{c_{w'\mathbf{u}'w} - dt_{w'\mathbf{u}'w}}{c_{w'\mathbf{u}'}} + \frac{dt_{w'\mathbf{u}'} P^{\text{PLD}}_{\mathbf{u}'}(w)}{c_{w'\mathbf{u}'}} \right] \\ &= \sum_{w'} \left( c_{w'\mathbf{u}'w} - dt_{w'\mathbf{u}'w} + dt_{w'\mathbf{u}'} P^{\text{PLD}}_{\mathbf{u}'}(w) \right) \\ &= c_{\mathbf{u}'w} - dt_{\mathbf{u}'w} + dt_{\mathbf{u}'} P^{\text{PLD}}_{\mathbf{u}'}(w) \end{aligned} \quad (10)$$

If we solve this and apply (8) we obtain:

$$P^{\text{PLD}}_{\mathbf{u}'}(w) = \frac{t_{\mathbf{u}'w}}{t_{\mathbf{u}'}} = \frac{\sum_{w'} t_{w'\mathbf{u}'w}}{\sum_w \sum_{w'} t_{w'\mathbf{u}'w}} = \frac{c_{\mathbf{u}'w}}{c_{\mathbf{u}'}} \quad (11)$$

## 5. EXPERIMENTS AND RESULTS

We have evaluated the PLDLM using two meeting transcription tasks: the NIST Rich Transcription 2006 spring meeting evaluation (RT06s), and the scenario portion of the AMI meeting corpus. In each case we compared the PLDLM to the IKNLM, the MKNLM and the HPYLM. We trained the all following trigram LMs using cutoff values for counts of 1, by using the SRILM toolkit [11] and the PLDLM program [1]. We did not use the strength parameter $\theta$ when training PLDLMs, i.e., we set $\theta = 0$ in (6). We used the AMI-ASR system [12] as the baseline platform for our ASR experiments, using all LMs in the first pass decoding.

### 5.1. NIST Rich Transcription 2006 Evaluation

For the RT06s task we trained LMs on 1.8M words of transcribed meetings data (meetings-s1), 10.6M words of transcribed conversationaltelephone speech (fisher-03-p1), and web data matched to meeting (webmeet; 36.1M words) and conversational (webconv; 162.9M words) speech collected using the approach described by Wan and Hain [13]. In total this resulted in 211.4M words of LM training data (ALL-1). We performed experiments using a vocabulary of 50,000 words. Table 1 shows the perplexity results on the NIST RT06s test data *rt06seval* (31,810 words). We found that, in all cases, the PLDLM outperforms the IKNLM and the MKNLM, and has comparably similar results to the HPYLM. The PLDLM with three discount parameters ($\text{PLD}_3$) also results in a slightly lower perplexity compared to the MKNLM.

**Table 1**. Perplexity results on NIST RT06s *rt06seval*.

| DATA | IKN | MKN | HPY | $\text{PLD}_3$ | PLD |
|---|---|---|---|---|---|
| meeting-s1 | 110.1 | 106.5 | 101.2 | 105.7 | **104.3** |
| fisher-03-p1 | 134.0 | 128.5 | 121.4 | 128.1 | **122.6** |
| webmeet | 176.8 | 170.6 | 159.3 | 169.6 | **159.7** |
| webconv | 135.4 | 131.8 | 120.2 | 130.5 | **120.8** |
| ALL-1 | 107.0 | 105.2 | 98.9 | 104.6 | **100.7** |

Table 2 shows the speech recognition WERs for *rt06seval* using LMs trained on ALL-1. The PLDLM is significantly better than the IKNLM and the MKNLM (weak), with $p < 0.01$ and $p < 0.05$ respectively, but is not significantly different to the HPYLM. We also evaluated using individual data sets, and found that there is no significant reduction in WER on meeting-s1 and fisher-03-p1, but there are significant reductions in WER by the PLDLMs on the webmeetings and webconv conditions, compared to the MKNLM. This may suggest that the increment of discounts in the PLDLM in turn increases the generalization ability of LMs in if the domain is somewhat mismatched.

**Table 2**. WER (%) results on NIST RT06s *rt06seval*.

| LMS | SUB | DEL | INS | WER |
|---|---|---|---|---|
| IKNLM | 14.5 | 9.7 | 2.7 | 27.0 |
| MKNLM | 14.4 | 9.8 | 2.7 | 26.8 |
| HPYLM | 14.2 | 9.8 | 2.6 | 26.5 |
| PLDLM | **14.2** | **10.0** | **2.4** | **26.6** |

[1]The executable program for power law discounting language model is available from `http://homepages.inf.ed.ac.uk/s0562315/`.

### 5.2. The AMI Corpus

For the AMI corpus, we trained LMs on 1.7M words of meeting transcripts (meeting-s2), 3.5M words of conversational speech transcripts (h5etrain03v1), a further 21.2M words of conversational speech transcripts (fisher-03-p1+p2), and 130.9M words of broadcast news transcripts (hub4-lm96), totalling 157.3M words of LM training data (ALL-2). Again we used a vocabulary of 50,000 words in our experiments. Table 3 shows the perplexity results on a test set of 32 AMI scenario meetings *amieval* (175,302 words). We found similar observations as those for *rt06seval*. Moreover, the PLDLM even slightly outperforms the HPYLM on some corpora, i.e., on fisher-03-p1+p2 and hub4-lm96.

**Table 3**. Perplexity results on the AMI meetings *amieval*.

| DATA | IKN | MKN | HPY | $\text{PLD}_3$ | PLD |
|---|---|---|---|---|---|
| meeting-s2 | 114.7 | 112.0 | 106.9 | 110.9 | **110.9** |
| h5etrain03v1 | 234.6 | 223.3 | 210.6 | 220.5 | **210.8** |
| fisher-03-p1+p2 | 221.2 | 210.9 | 200.7 | 209.7 | **198.2** |
| hub4-lm96 | 321.1 | 301.3 | 289.1 | 303.3 | **282.5** |
| ALL-2 | 168.6 | 163.9 | 158.8 | 163.7 | **157.9** |

Table 4 shows the speech recognition WERs on *amieval* using LMs trained on ALL-2. The reduction in WER by the PLDLM is significant comparing to the IKNLM and the MKNLM, with $p < 0.001$. Again there is no significant difference between the PLDLM and the HPYLM ($p < 0.15$), which to some extent implies that the PLDLM well approximates the HPYLM.

**Table 4**. WER (%) results on the AMI meetings *amieval*.

| LMS | SUB | DEL | INS | WER |
|---|---|---|---|---|
| IKNLM | 22.2 | 10.7 | 5.7 | 38.6 |
| MKNLM | 22.0 | 10.8 | 5.6 | 38.5 |
| HPYLM | 21.9 | 10.8 | 5.5 | 38.2 |
| PLDLM | **22.0** | **10.9** | **5.5** | **38.3** |

## 6. ANALYSIS AND DISCUSSIONS

**Absolute Discount.** To demonstrate the power law property of discounts in the PLDLM and the HPYLM, we trained a PLDLM using meeting-s1, and an HPYLM for 300 iterations on the same data. We plotted average discounts as a function of trigram counts in Fig. 1. The average discounts of the PLDLM approaches the expected values of discounts in the HPYLM.

**Effect of Strength Parameter $\theta$.** To study the effect of strength parameter $\theta$, we trained a trigram PLDLM using $\theta$ on fisher-03-p1, which has lower perplexity (121.7) than the PLDLM without $\theta$ (122.6). The initial values of $\theta$ for were obtained using the auxiliary sampling method, resulting in $\theta_1 = 3.3$, $\theta_2 = 2.25$, and $\theta_3 = 2.0$. Each time we ranged one $\theta$ over 0 to 100, while keeping the other two fixed. We evaluated the perplexity on *rt06seval* as the value of $\theta$ change. Results in Fig. 2 show that the PLDLM is more sensitive to $\theta_2$ and $\theta_3$ than $\theta_1$, and smaller values of $\theta_2$ and $\theta_3$ work better.

**Effect of Discount Parameter Numbers.** To investigate the effect of discount parameter numbers, we trained various trigram PLDLMs on fisher-03-p1+p2 by setting $p$ from 1 to $\infty$ (none), and evaluated

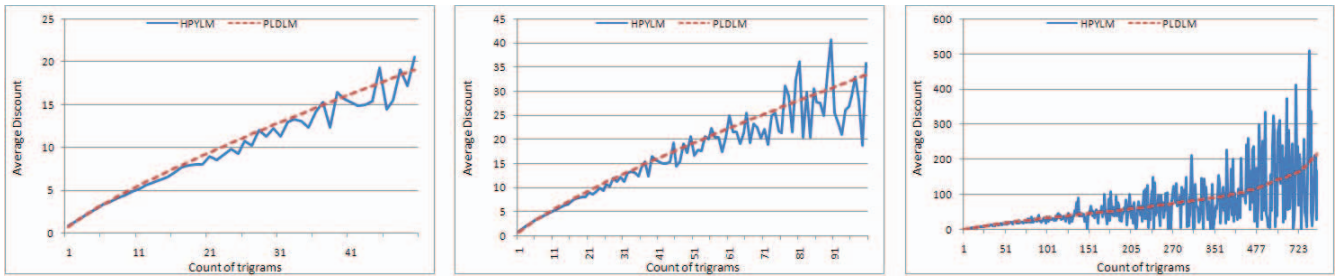


**Fig. 1**. Average discounts as a function of trigram counts in the PLDLM and the HPYLM trained on meeting-s1, with different scales for horizontal axis: first 50 counts (left), first 100 counts (middle), and first 1000 counts(right).

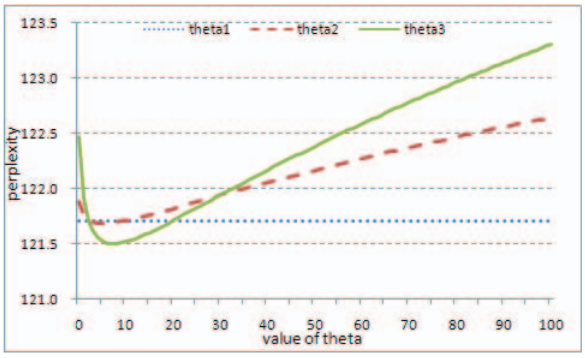


**Fig. 2**. Effect of strength parameter $\theta$ on *rt06seval*.

perplexity on *amieval*. As shown in Fig. 3, the perplexity decreases as the number of free discount parameters increases, which implies that it is better to use more discount parameters if we have a coherent way to estimate them, as done in the PLDLM.

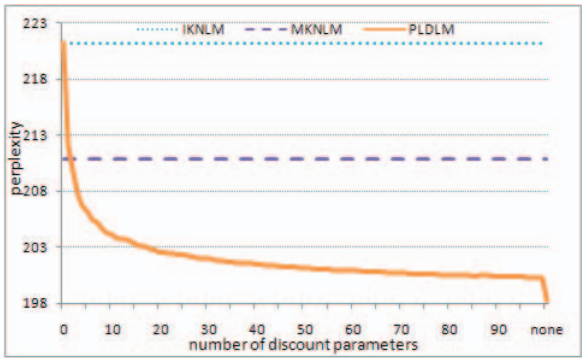


**Fig. 3**. Effect of discount parameter numbers on *amieval*.

## 7. CONCLUSIONS

We present in this paper a simple but efficient smoothing technique for language modeling that makes use of the power law distribution. The PLDLM estimates a smoothing parameter using power law discounting directly, thus avoiding expensive approximate inference, while maintaining comparable computational complexity to the IKNLM and the MKNLM. On the other hand, the PLDLM is an approximation to the HPYLM, producing similar performance to the HPYLM and in turn outperforming the IKNLM and the MKNLM. We conclude that the PLDLM is ready for use in practical large vocabulary speech recognition systems.

## 8. REFERENCES

[1] Stanley F. Chen and Joshua Goodman, "An empirical study of smoothing techniques for language modeling," *Computer Speech & Language*, vol. 13, no. 4, pp. 359–393, 1999.

[2] Reinhard Kneser and Hermann Ney, "Improved backing-off for $m$-gram language modeling," in *Proc. of ICASSP'95*, 1995, pp. 181–184.

[3] Jesús Andrés-Ferrer and N. Ney, "Extensions of absolute discounting (Kneser-Ney method)," in *Proc. of ICASSP '09*, 2009, pp. 4729–4732.

[4] Yee Whye Teh, "A hierarchical Bayesian language model based on Pitman-Yor processes," in *Proc. of the Annual Meeting of the ACL*, 2006, vol. 44.

[5] Songfang Huang and Steve Renals, "Hierarchical Pitman-Yor language models for ASR in meetings," in *Proc. of IEEE ASRU'07*, Dec. 2007, pp. 124–129.

[6] Songfang Huang and Steve Renals, "A parallel training algorithm for hierarchical Pitman-Yor process language models," in *Proc. Interspeech'09*, Sep. 2009.

[7] Jim Pitman and Marc Yor, "The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator," *The Annals of Probability*, vol. 25, no. 2, pp. 855–900, 1997.

[8] Sharon J. Goldwater, Thomas L. Griffiths, and Mark Johnson, "Interpolating between types and tokens by estimating power-law generators," in *Proc. of NIPS 18*, 2006.

[9] David J. C. MacKay and Linda C. Bauman Peto, "A hierarchical Dirichlet language model," *Natural Language Engineering*, vol. 1, no. 3, pp. 1–19, 1994.

[10] Yee Whye Teh, "A Bayesian interpretation of interpolated Kneser-Ney," Tech. Rep. TRA2/06, School of Computing, National University of Singapore, 2006.

[11] Andreas Stolcke, "SRILM - an extensible language modeling toolkit," in *Proc. of ICSLP'02*, September 2002.

[12] Thomas Hain, *et al.*, "The AMI system for the transcription of speech in meetings," in *Proc. of ICASSP'07*, April 2007, pp. 357–360.

[13] Vincent Wan and Thomas Hain, "Strategies for language model web-data collection," in *Proc. of ICASSP'06*, May 2006.