# Using two-stage conditional word frequency models to model word burstiness and motivating TF-IDF

**Peter Sunehag**[*]
Statistical Machine Learning Program, National ICT Australia
Locked bag 8001, 2601 ACT
Australia

## Abstract

Several authors have recently studied the problem of creating exchangeable models for natural languages that exhibit word burstiness. Word burstiness means that a word that has appeared once in a text should be more likely to appear again than it was to appear in the first place. In this article the different existing methods are compared theoretically through a unifying framework. New models that do not satisfy the exchangeability assumption but whose probability revisions only depend on the word counts of what has previously appeared, are introduced within this framework. We will refer to these models as two-stage conditional presence/abundance models since they, just like some recently introduced models for the abundance of rare species in ecology, seperate the issue of presence from the issue of abundance when present. We will see that the widely used TF-IDF heuristic for information retrieval follows naturally from these models by calculating a cross-entropy. We will also discuss a connection between TF-IDF and file formats that seperate presence from abundance given presence.

## 1 Introduction

### 1.1 Review and Discussion of background litteraure

It is well known that a word that has been used once in a text has an increased probability of appearing again according to a power-law distribution. A discrete probability distribution is a power-law distribution if $\frac{P(X \geq x)}{Cx^{-\alpha}} \to 1$ as $x \to \infty$ where $\alpha > 0$. Already in 1932, Zipf noticed that word frequencies in natural languages are roughly inversely proportional to their rank in the frequency table and, therefore follows a power-law distribution [20]. It has since been discovered that there are many natural and man made quantities that follow power-law distributions [15]. Some examples are citations of scientific papers, copies of books sold, magnitudes of earthquakes, intensity of solar flares, wealth of richest Americans, frequencies of family names in the US and populations of cities. It is important to notice that the probabilities that follow power-laws are often conditional probabilities, e.g. given that an earth quake with a magnitude of at least 3.8 has occured, its magnitude is power-law distributed with exponent 3; given that there is a city with at least 40000 people in a particular square in a grid drawn on a US map, its population is power-law distributed with exponent 2.3; and given that a certain word has appeared in a text, its frequency follows a power-law with exponent 2.2. Because of this conditioning, it is logical to consider two-stage conditional models where we seperately model the probability that something will occur and what happens when something does occur. A two-stage conditional model has recently been used [5] by Cunningham and Lindenmayer to model the abundance of a rare species of Possum in Australia by separating the issue of presence from the issue of abundance when present.

In the context of language models, the power-law property was called ”word burstiness” in a recent paper by Madsen et. al. [11] who used Polya urn models to model it. In a Polya urn document model, every document is started with an urn with a specified number of balls of different colors where every color represents a word type, i.e. an item in the vocabulary. When a ball is drawn, we note its color and then we put it back together with an additional ball of the same color. Polya urn models have power-law behavior with exponent 2 and are, therefore, natural candidates for modeling word burstiness.

Polya urn models and their variations have been used for many other purposes, e.g. combat modeling [14] and modeling of markets, which are developing toward monopoly [2]. The parameters estimated for Polya urns are the initial number of balls of the different colors in the urn. Fractional numbers of balls are allowed and typically the pa-

[*] Peter.Sunehag@nicta.com.au

rameters associated with modeling text are much smaller than one. This is due to the fact that most words occur in a very small fraction of all documents but can be quite abundant when they are present. Similarly, some rare animal or plant species can be hard to find but when you find them it is not unlikely that you find many individuals around the same site. Consider a situation where we initially have a small number of red balls, e.g. $0.005$. If we draw a red ball, then the total number of red balls is increased by one and becomes $1.005 \approx 1$. If we instead had started with twice as many red balls, i.e. $0.01$, we would have ended up with $1.01 \approx 1$. Therefore, the probability of a red ball being drawn again has very little to do with the probability of its appearance in the first place. This is consistent with the empirical study by Church [4] of the appearance of the word ”Noriega”, and some other words, in the Brown corpus. He discovered that the probability of that word to appear at least twice was closer to $p/2$ than to $p^2$ if $p$ is the probability that it will appear at least once. He concluded that ”The first mention of a word obviously depends on frequency, but surprisingly, the second does not”. This is clearly indicating that it would be a good idea to use a two-stage conditional model that separates presence from abundance given presence.

A problem with using Polya urns for text modeling is the computational cost for performing maximum likelihood estimation. Elkan [6] introduced a probability distribution he denoted the Exponential family Dirichlet Compound Multinomial (EDCM), which he has later used for document clustering [7]. This distribution is an approximation of a Polya urn. Furthermore the EDCM is a member of the exponential family and parameter estimation can be performed efficiently.

Goldwater et. al. [9] invented a method for constructing power-law distributions. It consists of two parts, one part called a generator and another called an adaptor. A sequence that does not exhibit burstiness is first drawn from the generator after which the adaptor creates a power-law distribution based on that sequence. Pitman-Yor processes were used as adaptors and a multinomial as a generator. The adaptor/generator framework will be reviewed in more detail later in this article but we would like to point out already here that the choice of a multinomial generator is what distinguishes the models used by Goldwater et. al. from models that separate presence from abundance given presence. A sequence drawn from a multinomial can contain the same word several times. If we replace the generator with a sampling without replacement scheme the result will be a model that separates presence from abundance given presence. We have, therefore, realized that the adaptor/generator framework is general enough to serve as a unifying framework for many distributions including two-stage conditional presence/abundance models. Goldwater et. al found that the best performing models for the application of morphology is the case when the Pitman-Yor process is close to a Chinese Restaurant Process (CRP).

### 1.2 Contributions of this article

In this article, we will prove that using a CRP adaptor and a multinomial generator results exactly in a Polya urn model. We will also show that the EDCM, as a probability distribution on count vectors (according to Elkan's definition), can be defined by using a CRP adaptor but with a generator that provides sequences of unique word types. We have discovered that the EDCM is equivalent to a two-stage conditional presence/abundance model. We will also define two new two-stage conditional presence/abundance models. In one of them, a CRP adaptor and a sampling without replacement scheme generator will be used. In the other, the CRP adaptor will be replaced by a generalized Polya urn of the kind defined by Chung et. al. [3] combined with a sampling without replacement generator. The generalized Polya urn depends on a parameter $\lambda \in [0, 1)$ representing the probability that the next word in the sequence will be of a type that has so far not been present. The generalized Polya urn follows a power-law with exponent $1 + \frac{1}{1-\lambda}$. Since it is known that word frequencies follow a power-law with exponent slightly larger than two, this is an interesting alternative to the CRP as an abundance distribution. Maximum likelihood estimation of $\lambda$ gives us a nice formula for estimating the power-law exponent for a particular corpus. The generalized Polya urn is not exchangeable, i.e. the probability of a sequence depends on the order. This means that if we want to calculate word count probabilities, the probabilities of all the ways that a particular count vector could have arisen must be added up. However, there is a weaker property than exchangeability that holds and that is the Commutativity Principle, named so by Wagner [19]. This says that when we are revising probabilities in the light of new evidence, the order of the evidence we have seen so far should not change the result, i.e. the probability revisions should commute. This concept has been studied extensively in the Philosophy of Science and in particular by people who are trying to understand the Bayesian concepts of subjective probability and belief. Bayesians usually define their models by choosing priors, which result in exchangeable models. However, if our aim is to measure the relative abundance of the various types in the context of the document in question, then the Commutativity Principle is exactly the condition required for representing the word sequence by a count vector.

To measure this relative abundance is essential for ranking documents in the field of information retrieval and in the ecological context, it is essential for monitoring rare and potentially endangered species. In this article, we will use the introduced models to provide a theoretical foundation for the Term Frequence-Inverse Document Frequency (TF-IDF) heuristic [18]. This heuristic states that if we want to rank documents according to how well they match a collection of distinct key words $w_1, ..., w_k$, we should rank a document $\tilde{d}$ according to the size of the expression

$$\sum_{i=1}^{k} \frac{n_{w_i}}{n}(\tilde{d}) \log \frac{|D|}{\sum_d I(n_{w_i}(d)) \geq 1)}$$

where $I$ is the indicator function , $n_w(d)$ is the number of times that the word type $w$ appeared in document $d$ and $n$ is the total number of word tokens in $d$. When Robertson [16] was reviewing various existing attempts to create a theoretical foundation for the practically so succesfull TF-IDF heuristic, he pointed out that a complication with using modeling and information theoretic approaches to motivate IDF relevance weighting is defining document probabilities based on word type presence probabilities. Furthermore, Robertson also discussed the non-triviality of motivating the TF part. The two-stage conditional presence/abundance approach takes care of the first problem, while we deal with the second one by focusing on the probabilities resulting from the total revisions caused by the word counts. Our approach enables us to formulate a simple derivation of the TF-IDF heuristic. In addition, our approach can also be connected to file formats for lossless compression of word count data.

In this context, it should be mentioned that Elkan [7] proved that if we use a Fisher kernel as a similarity measure between Polya urn distributions for the purpose of topic classification, the resulting measure is approximately equal to a variation of TF-IDF often used for this purpose.

### 1.3 Outline

Section two reviews existing word burstiness models and introduce new models within a unifying framework. Section three describes how to estimate the parameters of the new models. Section four shows how to derive the TF-IDF heuristic from one of those models and section five discuss how our modeling assumptions link TF-IDF to file formats that compress word count vectors. Section six contains a summary and future plans.

## 2 Word burstiness models

In this section we will review the models we want to compare and prove that they can all be expressed through the adaptor/generator framework [9]. We will also introduce a new model within this framework. First we will, however, discuss the basic assumption imposed.

### 2.1 Exchangeability and the Commutativy Principle

There are several application areas of language modeling including information retrieval and topic classification, where the most common words of the language in question are removed before an algorithm is used. These words are often called the stop words. They are words like ”and” and ”the” with little content. The data is then typically compressed by representing the documents by their word counts. This is a reason for using distributions satisfying the exchangeability condition, i.e. the condition that requires that the probability of a sequence does not depend on the order which is sufficient for motivating such compression. It is, however, not necessary if we are only interested in the resulting probability revisions. In ecology, we would be interested in calculating the abundance, or relative abundance, we believe that a species has at a certain site. This would be more important than calculating the probability that we would see what we have seen. With that aim, the Commutativity Principle [19], which is implied by, but not equivalent to exchangebility, is the necessary assumption for motivating the representation of a sequence of observations by a count vector. In formal mathematics the Commutativity Principle holds if and only if

$$P(X_{N+1}|X_1, ..., X_N) = P(X_{N+1}|X_{\pi(1)}, ..., X_{\pi(N)})$$

holds for any permutation $\pi$. The new two-stage conditional presence/abundance models will satisfy the Commutativity Principle but not the exchangeability assumption.

### 2.2 The Polya urn as a CRP augmented with labels sampled with replacement

As mentioned in the background, in a Polya urn document model, we start every document with an urn with a specified number of balls of different colors where each color represents a word type, i.e. an item in the vocabulary. When we draw a ball, we write down the word that corresponds to its color and then we put it back together with an additional ball of the same color. Thus if we started with $\beta_w$ balls of the color $w$ and we have so far drawn $n_w$ balls of color $w$, the probability that the next color is $w$ is equal to

$$\frac{n_w + \beta_w}{n + \beta}$$

where $\beta = \sum_w \beta_w$ and $n = \sum_w n_w$.
The Chinese Restaurant Process (CRP) is often described as follows: Suppose that we have a restaurant with an infinite number of tables, each of infinite size. Let the first customer sit at table number one. Then, when the second customer arrives, he or she sits down at the same table with probability $\frac{1}{1+\beta}$ and at table number two with probability $\frac{\beta}{1+\beta}$ where $\beta > 0$. Suppose that the first $n$ customers have sat down at $m$ different tables and $n_k$ are sitting at table number $k$. We then decide that the next customer will sit down at table $k$ with probability $\frac{n_k}{n+\beta}$ if $1 \leq k \leq m$ and at table number $m + 1$ with probability $\frac{\beta}{n+\beta}$. This scheme is called the CRP and as a probability distribution on allocations of customers to tables, it is exchangeable.

The CRP gives us a seating arrangement, which can be represented as a sequence of numbers $t_1, t_2, ..., t_n$ where $t_k$ for $1 \leq k \leq n$ is the number of the table that the $kth$ customer sits down at. If we want to use the CRP for modeling text, we have to attach labels, i.e. word types, to the tables. A type is an item in a vocabulary, while an occurrence of a word in a document is called a token. If we attach a word type to every table, the seating arrangement is transformed into a document, and each customer represents a word token. The following table explains the relationship between the different analogies that are being used:

| Polya urn | CRP | Language | Ecology |
|---|---|---|---|
| Ball | Customer | Word token | Individual |
| Color | Label of Table | Word type | Species |

The distribution that we draw the seating arrangement from, Goldwater et. al. [9] referred to as the *adaptor* and the distributions which we draw the labels from was called the *generator*. The generator decides the word types that will appear in the document and the order of their first appearance, and the adaptor decides how many times each word type will appear. Goldwater et. al. [9] used a multinomial as a generator resulting in a combined adaptor-generator model that is exchangeable. We will here prove that if we use a multinomial generator and a CRP adaptor we have exactly a Polya urn. It can be viewed as a way of expressing the well known equivalence between the CRP and a Polya urn [1].

**Theorem 1.** *Suppose that we use a CRP with parameter $\beta > 0$ as an adaptor and a multinomial with parameters $p_w$ as a generator. Furthermore, suppose that we so far have seen the word $w$, $n_w$ times and we have in total seen $n$ words. Then the probability that the next word is $w$ is equal to*

$$\frac{n_w + \beta p_w}{n + \beta} \tag{1}$$

*which is the probability that a Polya urn with parameters $\beta_w = \beta p_w$ would assign to the event.*

*Proof.* The word $w$ can be drawn in two different ways. We will use the restaurant analogy to describe the two. Either the new customer sits down at one of the already occupied tables with label $w$, which has probability $\frac{n_w}{n+\beta}$ , or the new customer sits down at a previously unoccupied table labeled $w$. The latter option has probability $\frac{\beta}{n+\beta} p_w$. If we let $\beta_w = \beta p_w$, the sum of the two equals $\frac{n_w + \beta_w}{n+\beta}$. □

### 2.3 New model A:The CRP augmented with labels sampled without replacement

In the model described in the previous section, several tables can have the same label. An alternative would be to sample without replacement. That is we start with an urn with $p_w$ balls of type $w$ where $\sum_w p_w = 1$. When a ball has been drawn, we remove all balls of that type from the urn. Then we will have unique labels for all the tables. This model is not exchangeable but the probabilities for what we will see next only depends on the counts of what we have seen so far and, therefore, the model satisfies the Commutativity Principle. We will from now on make extensive use of the gamma function $\Gamma(z) = \int_0^\infty t^{z-1} e^{-t} dt, z \geq 0$, which has the properties that $\Gamma(z+1) = z\Gamma(z)$ and $\Gamma(n+1) = n!$ for integers $n$.

**Theorem 2.** *For the distribution defined above with CRP parameter $\beta > 0$, the probability of a specific sequence of $n$ words where we have seen exactly the $m \leq n$ different word types $w_1, ..., w_m$, appearing for the first time in that order and where the word type $w_j$ has been seen exactly $n_{w_j}$ times is*

$$\frac{\beta^{m-1}}{(\beta - \beta_{w_1}) \cdot ... \cdot (\beta - (\beta_{w_1} + ... + \beta_{w_{m-1}}))} \cdot \frac{\Gamma(\beta)}{\Gamma(\beta + n)} \cdot$$

$$\cdot \prod_{j=1}^{m} \beta_{w_j} (n_{w_j} - 1)! \tag{2}$$

*where $\beta_w = p_w \beta$ and $p_w$ is the initial probability of the word $w$.*

To prove the theorem we need the following well known result, see e.g. [1].

**Lemma 1.** *The probability that a CRP with parameter $\beta$ generates a specific seating arrangement for $n$ customers seated at $m$ tables where $n_k$ customers are sitting at table $k$ is*

$$\beta^m \frac{\Gamma(\beta)}{\Gamma(\beta + n)} \prod_{k=1}^{m} (n_k - 1)!.$$

We can now present a proof of Theorem 2.

*Proof.* The probability of drawing the word types $w_1, w_2, ..., w_m$ from the urn of labels without replacement in that order is

$$\frac{\beta_{w_1} \cdot ... \cdot \beta_{w_m}}{\beta(\beta - \beta_{w_1}) \cdot ... \cdot (\beta - (\beta_{w_1} + ... + \beta_{w_{m-1}}))}.$$

If we have chosen a sequence of labels and a seating arrangement, we can permute both without changing the word-counts but we would always change the order of the first appearances. Therefore, the probability we seek is the product of the latest expression and the expression from Lemma 1. This yields the result in the theorem. □

### 2.4 Approximating the Polya urn, the EDCM

A Polya urn with parameters $\beta_w$ assigns probability

$$\frac{\Gamma(\beta)}{\Gamma(\beta + n)} \prod_w \frac{\Gamma(n_w + \beta_w)}{\Gamma(\beta_w)}$$

to a document of length $n$ where the word $w$ has appeared $n_w$ times. Elkan [6] tried to get around the problem that it is very difficult to maximize this expression. He used the approximation $\frac{\Gamma(x+\alpha)}{\Gamma(\alpha)} \approx \Gamma(x)\alpha$ for small $\alpha$ and because the parameters resulting from MLE on a Polya urn on a corpus of documents are in general very small, much smaller than one, he discovered that the parameters we get when optimizing the approximate expression are close to optimal for the original expression. The expression Elkan maximized was

$$\frac{\Gamma(\beta)}{\Gamma(\beta+n)} \prod_{w:n_w \geq 1} \beta_w (n_w - 1)!, \tag{3}$$

except that he worked with distributions on count vectors, i.e. he had an extra factor which equaled $\frac{n}{\prod_w n_w!}$, the number of documents with that count vector. Elkan proved that this is a much simpler optimization problem. By requiring that all partial derivatives should be zero, he arrived at the equations

$$\beta_w = \frac{\sum_d I(n_w(d) \geq 1)}{\sum_d (\Psi(\beta + n(d)) - \Psi(\beta))} \tag{4}$$

where $\Psi(x) = \frac{d}{dx}\log(\Gamma(x))$, $n_w(d)$ is the number of times the word $w$ has appeared in a document $d$ and $n(d)$ is the number of words in document $d$. The right hand side depends on $\beta = \sum \beta_w$ and, therefore, the parameters can not be directly computed. However, if the right and left sides of the equations are summed up, we end up with an equation where $\beta$ is the only unknown variable and, can therefore, be solved efficiently numerically. When we have $\beta$ we can calculate the $\beta_w$ parameters by using Expression 4.

The Expression 3 above does not directly define a probability distribution since it is not normalized. Elkan, however, points out that the corresponding normalized probability distribution on word count vectors is a probability distribution in the exponential family with interesting properties for modeling text. He also managed to succesfully use Expression 3 for clustering documents [7]. Instead of trying to calculate that probability directly, we define a probability distribution on subsets of the vocabulary of a specific size $m$, by letting the probability for a set consisting of the types $w_1, ..., w_m$ be proportional to $\beta_{w_1}\beta_{w_2} \cdot ... \cdot \beta_{w_m}$ where we have defined a parameter $\beta_w > 0$ for every word $w$ in the vocabulary. The probability of the set $\{w_1, ..., w_m\}$ of $m$ different word types from a vocabulary of $M$ word types is then equal to

$$\frac{\beta_{w_1}\beta_{w_2} \cdot ... \cdot \beta_{w_m}}{P_m(\beta_1, \beta_2, ..., \beta_M)} \tag{5}$$

where $P_m$ is the $m^{th}$ elementary symmetric polynomial with $M$ variables

$$P_m(\beta_1, ..., \beta_M) = \sum_{1 \leq i_1 < ... < i_m \leq M} \beta_{i_1} \cdot ... \cdot \beta_{i_m}.$$

We will define a probability distribution on documents by first sampling a seating arrangement from a Chinese Restaurant Process. We then know how many word types will appear in the document, e.g. $m$, and the document is, at this stage, defined as a sequence of numbers between 1 and $m$. We have, however, not decided which word types in our vocabulary correspond to those numbers but this can be done using the distribution that we just defined.

**Theorem 3.** *For the distribution defined above, the probability of a document where word $w$ has appeared $n_w$ times given that the document is of length $n$ is*

$$\frac{\beta^m}{m! P_m(\beta_1, ..., \beta_M)} \frac{\Gamma(\beta)}{\Gamma(\beta+n)} \prod_{w:n_w \geq 1} \beta_w (n_w - 1)!.$$

*Proof.* The probability of a specific seating arrangement and a specific sequence of word types is the product of the expression from Lemma 1 and $\frac{1}{m!}$ times the expression 5. The $\frac{1}{m!}$ is due to the fact that Expression 5 is a set probability and that the set can be ordered in $m!$ different ways. □

This expression defines the exchangeable normalization called the EDCM by Elkan [6]. The first factor $\frac{\beta^m}{P_m(\beta_1, ..., \beta_M)}$ represents the needed normalization for Expression 3. Unfortunately, the denominator is expensive to compute. Note that if we would remove the requirement on the order of the indices $i_1, ..., i_m$ from the definition of $P_m$, and only keep the requirement that they are different, then every term would appear $m!$ times. Furthermore, if we would include additional terms by also removing the condition that the indices should be different, we would end up with a sum that is equal to $\beta^m$. From this follows that $m! P_m(\beta_1, ..., \beta_m) < \beta^m$ and that the normalizing constant is larger than one.

### 2.5 New model B: A Generalized Polya urn augmented with labels sampled without replacement

The next new model we will consider is using a generalized Polya urn as an adaptor instead of the CRP. There are several different generalizations of the Polya urn scheme and we will here use the one by Chung et. al. [3]. This scheme depends on a choice of $\lambda \in [0, 1]$ and $\gamma \geq 0$. Given that we at the current stage have $a_k > 0$ balls of color $k$ in the urn, the probability that the next one will be of color $k$ is $(1-\lambda)\frac{a_k^\gamma}{\sum_k a_k^\gamma}$ and the probability that a ball of a previously unseen color will appear is $\lambda$. If we chose $\gamma = 1$ and $\lambda < 1$, as we will always do in this article, we get a power-law distribution with exponent $1 + \frac{1}{1-\lambda}$. $\gamma > 1$ leads to convergence to a situation where one colour has probability one, i.e. monopoly in the language of economists. $\gamma < 1$ leads to convergence towards uniformity. Growth rate increases with size for $\gamma > 1$ and decreaes with size for $\gamma < 1$, see the article by Chung at. al. [3]. If we let $\gamma = 1$, the main difference to the CRP is that the probability that the customer in question will sit down at the next table that none of the customers in the sequence of observations has so far used is always $\lambda$. When we use the generalized Polya urn as an abundance distribution, i.e. adaptor, we will, as we do with the CRP, start with an empty restaurant and let the first customer sit down at the first table and draw a label from the generator.

**Lemma 2.** *A generalized Polya urn with $\gamma = 1$ and parameter $\lambda$ and $a_k$, assigns probability*

$$\frac{\lambda^{m-1}(1-\lambda)^{n-m}}{(n-1)!}\prod_{k=1}^{m}(n_k-1)!x_k$$

*to a seating arrangement with $n$ customers occupying $m$ tables and the first customer to sit at table $k$ was customer number $x_k$.*

*Proof.* It has happened exactly $m-1$ times that a customer, excluding the first, has chosen a previously unoccupied table, and exactly $n-m$ customers have sat down at an already occupied table. The probability that customer number $x_k$ will sit down at the next empty table is always $\lambda$ which can also be written as $\lambda\frac{x_k}{x_k}$, and the probability that customer $\tilde{n}$ will sit down at table number $k$ if it is already occupied by $\tilde{n}_k$ customers is $(1-\lambda)\frac{\tilde{n}_k}{\tilde{n}}$. The expression in the lemma is the product of all of those probabilities. □

Suppose that we draw our labels from an urn with initial probability $p_w$ for $w$, and that we have so far drawn $n$ word tokens and $n_w$ of those are of the word type $w$, then if $n_{\tilde{w}} = 0$ the probability for $\tilde{w}$ is $\lambda\frac{p_{\tilde{w}}}{1-p_{w_1}-\ldots-p_{w_m}}$ where $w_1, ..., w_m$ are the word types that have appeared so far. Otherwise the probability is $(1-\lambda)\frac{n_{\tilde{w}}+a_{\tilde{w}}}{n+a}$.

**Theorem 4.** *A generalized Polya urn with $\gamma = 1$ and parameter $\lambda$ augmented with labels drawn without replacement from an urn, where word type $w$ has initial probability $p_w$, assigns probability*

$$\frac{\lambda^{m-1}(1-\lambda)^{n-m}}{(n-1)!\prod_{i=1}^{m-1}(1-\sum_{j=1}^{i}p_{w_j})}\prod_{k=1}^{m}(n_{w_k}-1)!p_{w_k}x_k$$

*to a document with $n$ word tokens of the $m+$ different types $w_1, ..., w_m$, indexed in the order of appearance, $n_w$ of the word tokens are of type $w$ and the first word token of type $w_k$ was word number $x_k$.*

# 3 Performing Maximum Likelihood Estimation

We will now turn to the problem of parameter estimation for our two new models. Since the data model is complete for both models, the information that is extracted from every document $d$ in the training data is the word types present in the document and the number of types $m(d)$ and tokens $n(d)$ it contains.

## 3.1 New model A

For every word type that has occurred in a specific document, it is always true that exactly one, the first, of the word tokens of that type was drawn from the generator and the rest came from the adaptor. We, therefore, have a complete data model. Since we only have types occuring in a small fraction of the documents, there is very little difference between parameter estimation for sampling with or without replacement. Even if we sample with replacement, we will have very few repetitions. Therefore, if we let the parameters $p_w$ be proportional to $\sum_d I(n_w(d) \geq 1)$, we will have close to optimal parameters for the generator and the lemma below provides us with the $\beta$ for the adaptor. If we let $\beta_w = \beta p_w$ we have exactly the parameters from Equation 4 in section 2.4.

**Lemma 3.** *Suppose that we have $D$ seating arrangements and that seating arrangement $d$ consists of $n(d)$ customers occupying $m(d)$ tables. Then the $\beta$ maximizing the likelihood defined by a CRP with parameter $\beta$, satisfies the equation*

$$\frac{1}{\sum_d(\Psi(\beta+n(d))-\Psi(\beta))} = \frac{1}{\beta}\sum_d m(d)$$

*where $\Psi$ as before denotes the digamma function $\Psi(x) = \frac{d}{dx}\log(\Gamma(x))$.*

*Proof.* Using the formula from Lemma 3 for the joint likelihood of these seating arrangements $L(\beta)$, we find that

$$\frac{d}{d\beta}L(\beta) = \sum_d(\frac{d}{d\beta}\log(\frac{\Gamma(\beta)}{\Gamma(\beta+n(d))}) + \frac{1}{\beta}m(d))$$

$$= \sum_d(\Psi(\beta) - \Psi(\beta+n(d)) + \frac{1}{\beta}m(d)).$$

It is now obvious that the equation in the lemma we are proving is equivalent to the equation $\frac{d}{d\beta}L(\beta) = 0$. □

## 3.2 New model B

Since the data model is complete also for Model B we only need to know how to find the parameters for the two classes given what have been drawn from them. It is only the adaptor that is different from Model A.

**Lemma 4.** *The parameter $\lambda$ that maximizes the joint likelihood, defined by a generalized Polya urn with parameter $\lambda$, of a set of seating arrangements, where seating arrangement $d$ consists of $n(d)$ customers sitting at $m(d)$ unoccupied tables is*

$$\lambda = \frac{\sum_d(m(d)-1)}{\sum_d(n(d)-1)}.$$

*Proof.* If we let $L(\lambda)$ be the likelihood, given some choice of the values $x_k$ and $n_k$ in Lemma 2 for every document $d$, it follows from the formula in Lemma 2 that

$$\frac{d}{d\lambda}\log L(\lambda) = \sum_d (\frac{m(d)-1}{\lambda} - \frac{n(d)-m(d)}{1-\lambda})$$

regardless of the choices of $x_k$ and $n_k$. From this follows that $\frac{d}{d\lambda}\log L(\lambda) = 0$ if and only if $\lambda = \frac{\sum_d (m(d-1))}{\sum_d (n(d)-1)}$. □

## 4 Deriving TF-IDF as a cross-entropy

In information retrieval, the aim is to rank documents according to how well they match a certain query. A query $Q$ is a set of distinct word types $\{w_1, ..., w_k\}$. To define a ranking measure we will, in this article, utilize the cross-entropy concept that has arisen in rare event simulation. Rare event simulation aims at accurately estimating very small probabilities. The cross entropy $CE(p, q) = \sum_w p(w) \log \frac{1}{q(w)}$ can be interpreted as the average number of bits needed to identify an event sampled from $p$, using a coding scheme based on $q$. It is closely related to the Kullback-Leibler divergence. Let $q$ be the ur-distribution we begin the sampling of every document with, i.e. for both of our new models, $q(w)$ would be proportional to the number of documents that $w$ is present in. If we let $IDF(w) = \frac{|D|}{\sum_d I(n_w(d) \geq 1)}$ and $N = \sum_w \sum_d I(n_w(d) \geq 1)$, then

$$log \frac{1}{q(w)} = \log \frac{N}{|D|} + \log IDF(w).$$

$\frac{N}{|D|}$ is the average number of word types per document. Let $p(x)$ be the probabilities resulting from the revision scheme that defines Model B, applied to the word counts for the document that we are analyzing. $p(w)$ is, therefore, equal to $(1-\lambda)TF(w)$, where $TF(w)$ is the frequency of $w$ in $d$, if $w$ is present in $d$, and miniscule otherwise. It follows that $CE(p, q) \approx$

$$(1-\lambda)\sum_w TF(w)(\log \frac{N}{|D|} + \log IDF(w)).$$

Given a query $Q$, we can define a matching measure $M_D(d, Q)$ for how well $d$ matches the query $Q$ in the context of corpus $D$ by instead defining $p$ by only taking into account the information about how many times every word in the query has appeared in the document and the total number of words in the document. Using revision scheme B, the result will actually only depend on the frequency of the words in the query and not on the length of the document. If we let $M_D(d, Q)$ be proportional (with $\frac{1}{1-\lambda}$ as the proportionality constant) to $CE(p, q)$ with the $p$ resulting from those revisions, it follows that $M_D(d, Q) \approx$

$$\sum_{w \in Q} TF(w)(\log \frac{N}{|D|} + \log IDF(w)). \qquad (6)$$

This is a variation of the TF-IDF heuristic where $TD \log IDF$ has been replaced by $TD(C + \log IDF)$. For $C > 0$ we have a measure that is based on a more moderate term (i.e. word type) weighting than the original measure, which corresponds to $C = 0$. If you are creating a search engine, you might be interested in choosing $C$ from the preferences of a test group of users instead of by the formula above. The formula above depends on how many words you have chosen to exclude to begin with. If you would choose to also exclude all words that are not in the query, you would end up with a small $C$. Thus, we actually have a foundation for choosing almost any $C > 0$. The original TF-IDF, which corresponds to $C = 0$, has been reported to be more prone to ranking a document containing $k-1$ out of $k$ query words above documents containing all of them than is preferred by search engine users. This is called non-coordination. The level of non-coordination for the measure presented here will be decreasing with increasing $C$. Many of the variations of TF-IDF in use contain a moderating constant of some sort to improve the coordination level [10].

The main idea of this section is to base a coding scheme on the ur-distribution $q$. All the discussed two-stage conditional models give us the same approximately optimal $q$. Instead of the $p$ used above, we could plug in the document's word fequency vector but with zeros instead of the coefficients corresponding to words that are not in the query. $CE(p, q)$ would then equal Expression 6. This can be interpreted as the number of bits needed to store the relevant part of the frequency vector.

## 5 TF-IDF compression of word count data

Cross entropy has been proven useful for estimation of probabilities for rare events, and $CE(p, q)$ can be interpreted as the average number of bits needed to idenify an event sampled from $p$, using a coding scheme based on $q$. This is related to the close connection between statistical modeling and compression techniques. If we want to store a collection of word count vectors in a compressed format, the first thing that we should do is to store every vector as a set of pairs $(w_i, n_i)$, where we have discarded the pairs corresponding to word types that have not occured. The next step is to find one coding scheme for the word types, based on their probability of being present in a document, and one coding scheme for the numbers representing the counts. Using document frequency for the first part and a power-law with exponent 2 for the second are natural choices. In this article, TF-IDF has been explained by implementing the two ideas of separating presence from abundance when present and that word counts are power-law distributed. Thus, it is reasonable to use the name TF-IDF compression for the resulting file format. However, in contrast to the models introduced in this article, the described file format does not incorporate a model for the number of present word types.

## 6 Summary and future plans

We argue that when word counts are modeled, the issue of presence should be separated from the issue of abundance given presence. For the latter issue word burstiness, i.e. the power-law behavior of word tokens, should be taken into account. We use the adaptor/generator framework introduced by Goldwater et. al as a unifying framework for word burstiness models and we present new two-stage conditional presence/abundance models within it. We use the separation of presence from abundance given presence to derive the TF-IDF heuristic for information retrieval and we also use this separation to provide a connection between TF-IDF and file formats that compress word count vectors. The author believes that there are many opportunities for using NLP techniques for biodiversity information analysis and management, in particular for multiple species inventories and monitoring [12]. Models like Latent Dirichlet Allocation that take co-occurences into account could turn out to be useful.

Future plans also involve topic classification and deriving new and/or existing variations of TF-IDF by using generalized entropy and Bregman divergences together with the introduced models.

## Acknowledgements

The author would like to thank Tiberio Caetano and Alex Smola for their constructive criticism and useful comments.

## References

[1] D. Aldous, (1981) *Exchangeability and Related Topics, in l'cole d't de probabilits de Saint-Flour, XIII1983*, Berlin: Springer.

[2] A.W. Brian, Increasing Returns and Path Dependence in the Economy, Chapter 4., Ann Arbor, MI: University of Michigan Press, 1994

[3] F. Chung , S. Handjani and D. Jungreis (2003) , Generalizations of Polya's urn problem , *Annals of Combinatorics* **7**:141-154

[4] K. Church (2000) , Empirical Estimates of Adaptation: The chance of Two Noriega's is closer to $p/2$ than $p^2$. , *Coling*:173–179

[5] R. Cunningham and D. Lindenmayer (2005) , Modeling count data of rare species: some statistical issues , *Ecology* **86(5)**:1135–1142

[6] C. Elkan (2005) , Deriving TF-IDF as a Fisher kernel , *Proceedings of the International Symposium on String Processing and Information Retrieval (SPIRE'05), Buenos Aires, Argentina, November 2005*, 296-301

[7] C. Elkan (2006) Clustering documents with an exponential-family approximation of the Dirichlet compound multinomial distribution , *Proceedings of the 23rd International Conference on Machine Learning (ICML), 2006* 289-296

[8] A. Ellison and A. Agrawal (2005) , The Statistics of Rarity, Special Feature Summary , *Ecology* **86(5)**:1079–1080

[9] S. Goldwater, T. Griffiths and M. Johnson (2006) , Interpolating Between Types and Tokens by Estimating Power-Law Generators , *Proceedings of Neural Information Processing Systems (NIPS) 2005*

[10] D. Hiemstra (1998) , A Linguistically Motivated Probabilistic Model of Information Retrieval , *European Conference on Digital Libraries*:569–584

[11] R.E. Madsen, , E. Kauchak, and C. Elkan, (2005) , Modeling Word Burstiness using the Dirichlet distribution , *Proceedings of the 22nd International Conference on Machine Learning (ICML), 2005*, 545–552

[12] Manley P. and Horne B. (2004) , The Multiple Species Inventory and Monitoring protocol: A population, community and biodiversity monitoring solution for National Forest System lands , *Proceedings of Unifying Knowledge for Sustainability in the Western Hemisphere 2004*

[13] R. Kuhn and R. Mori (1990) , A Cache-Based Natural Language Model for Speech Recognition. , *IEEE Transactions on Pattern Analysis and Machine Intelligence* **12(6)**:570–583.

[14] P. Kwam and D. Day (2001) , The multivariate Polya distribution in combat modeling , *Naval Research Logistics* **48**:1–17.

[15] M.E.J. Newman (2005) , Power laws, Pareto distributions and Zipf's law , *Contemporary Physics* **46(5)**:323–351

[16] S. Robertson (2004) , Understanding Inverse Document Frequence: On theoretical arguments for IDF , *Journal of Documentation* **60(5)**:503–520

[17] H.A. Simon (1955) , On a class of skew distribution functions , *Biometrika*, **42**(3/4): 425-440

[18] K. Spark Jones (1972) , A statistical interpretation of term specificity and its applications in retrieval , *Journal of Documentation*, **28**: 11–21

[19] C. Wagner (2003) , Commuting probability revisions: the uniformity rule , *Erkenntnis*, **59**: 349-364

[20] G. Zipf (1932) , *Selective Studies and the Principle of Relative Frequency in Language* , Harvard University press, MA