N-Poisson Document Modelling

Eugene L. Margulis
Institut für Informationssysteme, ETH Zentrum, CH-8092 Zürich, Switzerland

Abstract

This paper is a report of a study investigating the validity of the Multiple Poisson (nP) model of word distribution in document collections. An nP distribution is a mixture of n Poisson distributions with different means. We describe a practical algorithm for determining if a certain word is distributed according to an nP distribution and computing the distribution parameters. The algorithm was applied to every word in four different document collections. It was found that over 70% of frequently occurring words and terms indeed behave according to the nP distributions. The results indicate that the proportion of nP words depends on the collection size, document length and the frequency of the individual words. Most of the nP words recognised are distributed according to the mixture of relatively few single Poisson distributions (two, three or four). There is an indication that the number of single Poisson components in the mixture depends on the collection frequency of words.

1 Introduction

This paper describes the results of the current study attempting to find a useful statistical model of large collections of full text documents. We describe the results of several experiments that examine the Multiple Poisson[1] model of word distributions in document collections. The initial results indicate that over 70% of frequently occurring words indeed behave according to Multiple Poisson distributions and that the composition of the nP distribution functions depends on the collection frequency of words. The nP properties are also displayed by terms (words after applying a word reduction algorithm). The detailed analysis of the nP distribution functions composition results and the effects of various document length normalisation techniques on the nP properties of words appear in a related report (Margulis, 1991).

The Information Retrieval (IR) area of Computing Science deals with the problems associated with processing of large collections of usually unstructured documents. One of the traditional applications of IR is the retrieval of full text documents from large document collections. The major problem lies in finding documents within the collection that satisfy a certain query (a phrase or a set of words that describe the documents to be retrieved). To solve this problem the documents are assigned a set of identifiers (indexes) that describe what the documents are about. This process is called *indexing*. The indexes are then examined in order to determine which documents satisfy the query the most and should be retrieved. There exist many indexing/retrieval approaches and much research has been directed towards finding better ones. In spite of the abundance of IR approaches, there are several unresolved fundamental problems. It is very hard to judge the effectiveness of indexing/retrieval approaches objectively. A method that works well for one collection, may not work well for another. We are not capable of determining or justifying in advance the best method to use with a given collection. We know little of the characteristics of document collections that affect the effectiveness of IR functions.

Most of these problems arise due to the lack of a useful statistical model of a document collection. A useful model (with respect to IR) would provide us with a set of characteristics and measurements that could be used in determining the parameters of various information re-

[1] In this paper we use "n-Poisson model" or "nP model" terminology to refer to the Multiple Poisson model.

15th Ann Int'l SIGIR '92/Denmark-6/92
© 1992 ACM 0-89791-524-0/92/0006/0177...$1.50

trieval methods. A number of IR related models have been proposed in the past (e.g. vector space model, various probabilistic models, etc.)[2]. Most of these models introduce new formalisms to support a certain indexing and retrieval strategy. Only a few of the models attempt to describe the properties of a document collection irrespectively of a specific retrieval strategy.

This study analyses the validity of a statistical model described by Bookstein and Swanson (1974) who suggest that the frequency of occurrence of words in documents can be modelled by a sum of Poisson distributions. Several studies have attempted to test the validity of the model experimentally, but were unable to prove or disprove the validity of the model conclusively (Harter, 1975; Srinivasan, 1990). Another study used a special case of the nP model to develop an index term weighting scheme (Robertson et al., 1981).

The main goal of this study was to find a conclusive experimental evidence that either supports or disproves this model. The results of our experiments strongly support the Multiple Poisson model.

2 Notation Summary

In this section we summarise the notation and some of the basic definitions and assumptions used throughout this study.

2.1 Words and Terms

Documents in document collections can be partitioned into various *text tokens*: words, reduced words, n-grams, phrases, etc. Text tokens have meanings or concepts associated with them. The meanings of the text tokens define the meanings and topics of the documents.

In this study we deal with two types of text tokens: *words* and *terms*. A *word* is a sequence of nonblank alphabetic characters that appears in full text documents. In some IR literature a *word* is referred to as a "word token" or "text word". A *term* is a reduced word that results from the application of a word reduction (suffix/prefix removal) algorithm. We say that t is a *term* if it is a reduction of some word w, its length is greater than 2 and the word w does not belong to the list of commonly used "stop words"[3]. In some IR literature *terms* are referred to as "keywords", "word stems", "reduced words" or "index terms".

[2]Bärtschi (1985) provides a good overview of various IR models.

[3]Stop words are the words that occur very often but have little or no meaning for information retrieval purposes within the context of a specific document collection. For example: "a", "the", etc. are common stop words. In a collection of financial articles the word "dollar" might be a stop word.

We use the following notation to describe functions defined with respect to text tokens, words, terms and documents:

$ndoc(\mathbf{C})$ the number of documents in collection $\mathbf{C}$
$ntok(\mathbf{C})$ the number of unique tokens in collection $\mathbf{C}$
$wlen(D)$ the total number of words in document D
$tlen(D)$ the total number of terms in document D
$awlen(\mathbf{C})$ average number of words per document in $\mathbf{C}$
$atlen(\mathbf{C})$ average number of terms per document in $\mathbf{C}$
$wdfreq(D, w)$ number of times word w occurs in D
$tdfreq(D, t)$ number of times term t occurs in D
$tcfreq(\mathbf{C}, t)$ number of times term t occurs in $\mathbf{C}$
$dfreq(D, h)$ number of times token h occurs in D
$cfreq(\mathbf{C}, h)$ number of times token h occurs in $\mathbf{C}$

Note that $wdfreq$ and $tdfreq$ are special cases of $dfreq$ and that $wcfreq$ and $tcfreq$ are special cases of $cfreq$.

2.2 Extents of Coverage

Documents are written to provide information on a certain topic or a set of topics. The extent of coverage of various topics within the document varies from one topic to another. For example, this article is primarily about the n-Poisson text models but it also describes the estimation methods of parameters of probability distributions. This article, however, is to a very large extent about the n-Poisson models and to a lesser extent about parameter estimation. To estimate the extent of coverage we can use the following assumption:

Assumption 2.1 : The frequency of occurrence of a specific text token h in a particular document depends on the extent to which this document is related to the topic that is associated with the text token.

This assumption is a key assumption in IR and is related to the Luhn's proposal that the frequency of word occurrence is a useful measure of word significance (Luhn, 1958). Thus the extent of coverage of topics represented by a specific text token in the document can be approximated by the frequency of occurrence of the token within the document: $dfreq(D, h)$. For words and terms the extents of coverage are approximated with $wdfreq(D, w)$ and $tdfreq(D, t)$ respectively.

Document collections can be divided into subsets of documents with respect to the extent of coverage of a certain topic. Therefore, a collection $\mathbf{C}$ can be divided into subsets $\mathbf{C}_i$ with respect to the extent of coverage of topics associated with a certain text token. These subsets are usually referred to as "levels" or "classes" of coverage.

3 nP Model Overview

In this section we present the definition of the n-Poisson model. We also summarise previous studies that at-

tempted to validate various special cases of the nP model.

3.1 nP Models

n-Poisson models of documents are based on the following assumption:

Assumption 3.1 : The frequency of occurrence of text tokens within the full text documents in a document collection can be described by a sum of Poisson distributions.

Each summand in this sum is an independent single Poisson distribution that describes the frequency of occurrence of the text token within the subset of documents that belong to the same level of coverage of the topics related to that text token (Bookstein & Swanson, 1974). The probability that a randomly chosen document D contains k occurrences of a certain text token h is given by:

$$P(dfreq(D,h)=k) = \sum_i \pi_i \frac{\lambda_i^k}{k!} e^{-\lambda_i}$$

Here i denotes the class of coverage of the topics related to the text token; λ_i is the average extent of coverage of topics related to the text token within the class $\mathbf{C}_i$; π_i is the probability that the document belongs to the class $\mathbf{C}_i$, and $\sum_i \pi_i = 1$.

The process of generation of documents can be viewed as a stochastic process where the documents are created by randomly selecting text tokens. For every document D, each text token h has a certain probability of being selected. This probability depends on the extent of coverage of topics associated with h in D. Since the number of various text tokens (e.g. words) is very large and the probability of being selected for every token is very small, the process of document generation can be viewed as a poisson process. The mean of this poisson process (λ) is the average number of occurrences of the text token h per document and represents the extent of coverage of topics associated with h in the document D. A document collection can be partitioned into a number of classes of documents w.r.t. the extents of coverage of topics related to a specific text token h. The distribution of the text tokens h within each class $\mathbf{C}_i$ is governed by a single poisson process with a mean of λ_i. Thus the distribution of a certain text token h in documents within the whole collection is governed by the sum of poisson distributions, one for each class of coverage.

For example, consider a collection of documents on vacation travel ($\mathbf{C}$). Suppose we can divide the collection into two subclasses: $\mathbf{C}_1$ – the documents about air travel, and $\mathbf{C}_2$ – all other documents. The probability of the word "aircraft" occurring in $\mathbf{C}_1$ documents is higher than the probability of the word "aircraft" occurring in $\mathbf{C}_2$ documents. In $\mathbf{C}_1$ the word "aircraft" occurs on average λ_1 times per document and λ_2 times per document in $\mathbf{C}_2$. The probabilities that a randomly chosen document D from $\mathbf{C}_1$ or $\mathbf{C}_2$ contains k occurrences of the word "aircraft" are:

$$\begin{aligned} P_1(k) &= P(D \in \mathbf{C}_1 \wedge wdfreq(D, \text{"aircraft"}) = k) \\ &= \frac{\lambda_1^k}{k!} e^{-\lambda_1} \end{aligned}$$

$$\begin{aligned} P_2(k) &= P(D \in \mathbf{C}_2 \wedge wdfreq(D, \text{"aircraft"}) = k) \\ &= \frac{\lambda_2^k}{k!} e^{-\lambda_2} \end{aligned}$$

Thus the probability that a randomly chosen document from the whole collection $\mathbf{C}$ contains k occurrences of the word "aircraft" can be computed as:

$$\begin{aligned} P(k) &= P(D \in \mathbf{C} \wedge wdfreq(D, \text{"aircraft"}) = k) \\ &= P((D \in \mathbf{C}_1 \vee D \in \mathbf{C}_2) \\ &\quad \wedge wdfreq(D, \text{"aircraft"} = k)) \\ &= P((D \in \mathbf{C}_1 \wedge wdfreq(D, \text{"aircraft"}) = k) \\ &\quad \vee (D \in \mathbf{C}_2 \wedge wdfreq(D, \text{"aircraft"}) = k)) \\ &= P(D \in \mathbf{C}_1) \cdot P_1(k) + P(D \in \mathbf{C}_2) \cdot P_2(k) \\ &= \frac{|\mathbf{C}_1|}{|\mathbf{C}|} \cdot \frac{\lambda_1^k}{k!} e^{-\lambda_1} + \frac{|\mathbf{C}_2|}{|\mathbf{C}|} \cdot \frac{\lambda_2^k}{k!} e^{-\lambda_2} \end{aligned}$$

Note: $\frac{|\mathbf{C}_1|}{|\mathbf{C}|} + \frac{|\mathbf{C}_2|}{|\mathbf{C}|} = 1$

The discussion above demonstrates a possible analytical interpretation of the Multiple Poisson model. In the rest of this paper we present experimental results that support the validity of the model.

The Multiple Poisson model can be defined with respect to any text token. In this study we recognise two types of poisson models: nPw based on words and nPt based on terms. We define these models as follows:

nPt : $P(tdfreq(D,t)=k) = \sum_i \pi_i \frac{\lambda_i^k}{k!} e^{-\lambda_i}$

nPw : $P(wdfreq(D,w)=k) = \sum_i \pi_i \frac{\lambda_i^k}{k!} e^{-\lambda_i}$, where $\sum_i \pi_i = 1$

If the distribution of a certain text token (e.g. word or term) complies with a Multiple Poisson model, then we say that this text token is nP. We define a Boolean function $np(h)$ that determines whether the token h is nP:

$$\begin{aligned} np(h) = {} & \exists n, \exists \{\lambda_i, \pi_i\}_{i=1}^n, \forall D \in \mathbf{C} \\ & P(dfreq(D,h)=k) = \sum_i \pi_i \frac{\lambda_i^k}{k!} e^{-\lambda_i} \\ & \text{where } \sum_i \pi_i = 1 \end{aligned}$$

3.2 Previous Work

Harter attempted to verify the validity of a special case of the nPw model – the $2P$ model (Harter, 1975). The $2P$ model is the nP model with only two coverage classes. The $2P$ model used by Harter can be described in our notation as follows:

$$P(wdfreq(D,w)=k)=\pi\frac{\lambda_1^k}{k!}e^{-\lambda_1}+(1-\pi)\frac{\lambda_2^k}{k!}e^{-\lambda_2}$$

Harter's goal was to find a better indexing strategy based on the $2P$ model. He used a collection of 650 Sigmund Freud abstracts for his experiments. The average length of an abstract was 223 words. Harter used the Method of Moments to estimate the $2P$ parameters π, λ_1 and λ_2 for the index terms considered "good" by human indexers. 38% of these terms fitted a $2P$ distribution according to the χ^2 goodness of fit test. The experiment showed that a significant number of good index terms were $2P$, but it did not provide a conclusive evidence of the validity of assumption 3.1 for the following reasons:

- Only "good index terms" were analysed
- Only the special case ($2P$) was examined
- The collection size was too small for many words to have high enough frequencies for χ^2 test
- Only relatively short abstracts and not complete documents were used
- The Method of Moments is not the best available method for parameter estimation (Breiman, 1973, pp. 84-85)

Srinivasan investigated the possibility of extending Harter's $2P$ model to $3P$ (Srinivasan, 1990). She used a variation of the nPt model that consist of the three poisson terms. The model was tested with 59,917 documents from the INSPEC database. Only abstracts and citation details were used for each document with an average length of 58 words. Index terms that were selected from the identifying phrases of the documents and were determined "good" in another study were used for the approximation of $2P$ and $3P$ parameters using the Method of Moments. The study confirmed Harter's results by finding that 43% of the terms tested (85 out of 196) were $2P$, but failed to find any $3P$ terms. Although this experiment confirmed Harter's results, it failed to provide a conclusive evidence of the validity of assumption 3.1 for the following reasons:

- Only "good index terms" were analysed
- Only special cases ($2P$ and $3P$) were examined
- Only very short abstracts ($awlen = 58$) and not complete documents were used
- The Method of Moments is not the best available method for parameter estimation (Breiman, 1973, pp. 84-85)

The studies described above provide strong evidence that nP models could play an important role in describing the word occurrence distribution in full text documents. Both studies, however, investigate only special cases of the nP model with respect to preselected "good" index words/terms and use collections of relatively short abstracts rather than complete documents.

Harter's $2P$ model was used by Robertson, Rijsbergen and Porter (1981) to develop a $2P$ based index term weighting schema. The authors mention that the performance of the weighting schema in the experiments conducted was "slightly disappointing". The experiments conducted had most of the drawbacks of the Harter's study: only a special case ($2P$) was examined, the documents used were short (average length was 19.96 terms) and the Method of Moments was used for the parameter estimation. We believe that a term weighting schema based on the complete nP model would be more successful.

The inconclusive results of these studies prompted us to examine the validity of the complete nP model in large collections of full text documents.

4 Experiment Description

The goal of the current study was to find a convincing evidence that would either support or disprove the assumption 3.1 with respect to nPw and nPt models. In order to find such evidence we estimate the λ_i and π_i parameters of the nP distribution for every word and term in our test collections and test the goodness of the estimation using the χ^2 goodness of fit test.

4.1 Data Description

In this experiment we analyse four document collections:

FT85: collection of articles that appeared in the "Financial Times" newspaper in 1985

WS87: collection of articles that appeared in the "Wall Street Journal" newspaper in 1987

RAMR: on-line collection of film reviews from usenet news group *rec.arts.movie.reviews*

INYT: on-line collection of summer 1991 "New-York Times" articles[4]

Two larger collections, *FT85* and *WS87*, were used for the major portion of tests in this study. *RAMR* and *INYT*, considerably smaller collections, were used to verify the results obtained from *FT85* and *WS87*. *FT85*

[4]The articles are electronically distributed by the MIT BCIS project. The description of the project appears in (Giford, 1990).

Collection	*ndoc*	*awlen*	*atlen*	*mwlen*
FT85	6750	772	342	400
WS87	13938	942	496	500
RAMR	1034	655	300	175
INYT	1084	366	190	135

Figure 1: Test Collection Characteristics

collection was only used for nPt model related tests, in that collection we only had access to the term occurrence data and the document length data, but not to the full text documents. The table in Fig. 1 shows some of the characteristics of these collections. In this table *mwlen* is the *wlen* of the smallest document in the collection.

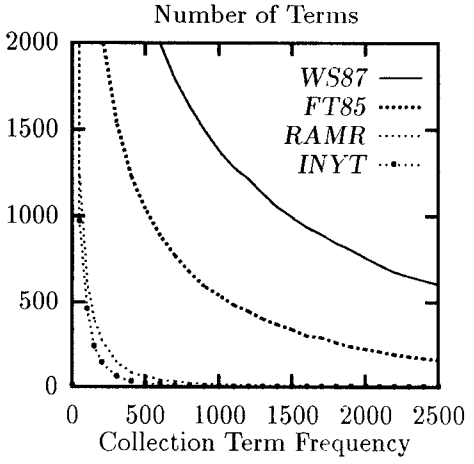

Figure 2: Term Frequencies

In this study we were interested in the analysis of complete full text documents (as opposed to the relatively short abstracts used in the previous studies). Therefore, only the documents with length greater than 400 words (approximately one typed page of English text in 12pt font) were selected from the complete collection of the Financial Times articles for 1985 to comprise the *FT85* test collection. Only documents with length greater than 500 words were selected from the collection of "Wall Street Journal" 1987 articles to comprise *WS87* collection. Smaller documents were not excluded from the *RAMR* and *INYT* collections since since these collections are considerably smaller.

Many tests in this study are based on the collection frequency of text tokens: words and terms. The graphs in Fig. 2 (terms) and Fig. 3 (words) show the number of text tokens in the collections tested whose collection frequency is greater than a certain value. The X axis in this graph is the collection frequency of the

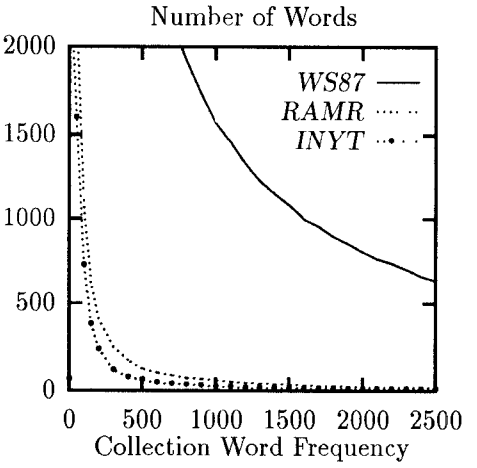

Figure 3: Word Frequencies

text tokens and the Y axis is the number of text tokens whose frequency is greater than x: $y = |T|$, $T = \{h|cfreq(\mathbf{C}, h) > x\}$

Porter's stemming algorithm (Porter, 1980) was used for word stemming to create terms. Words (and terms) shorter than 2 characters were not used for testing.

4.2 Experiment Procedures

The experiment involved the estimation of the nP parameters for each word and term in our test collections and then testing the estimated parameters with the χ^2 goodness of fit test. All experiments in this study were run on a Sun 4/490 SPARC server.

The estimation of the parameters of nP distribution is done using the Maximum Likelihood Estimate (MLE) based algorithm described by Hasselblad (1969). Our implementation of this algorithm is based on the implementation suggested by Agha and Ibrahim (1984) in algorithm AS203.

The performance of the estimation algorithm depends on the initial values of the parameters being estimated (λ_i and π_i). We generate the initial values of these parameters randomly and if the estimation attempt fails we use the "Abundance of witnesses" approach (Karp, 1990) to determine whether to stop the estimation process or make another attempt with a different set of initial values. If the estimation or the goodness of fit test fails for one set of initial values, another set is generated and the estimation and goodness of fit test is repeated, up to 10 times. If, after 10 times, no estimates that satisfy the goodness of fit test are found, the number of poisson components is increased and the estimation process repeated.

The simplified description of the estimation process

```
for each token do
    npcomponents := 1
    estimated := FALSE
    while (¬estimated)∧(npcomponents < 8) do
        npcomponents := npcomponents + 1
        nattempts := 0
        while (¬estimated)∧(nattempts < 10) do
            nattempts := nattempts + 1
            generate initial values of nP parameters
            estimate parameters with MLE algorithm
            estimated := χ² goodness of fit test
                 is satisfied
        endwhile
    endwhile
endfor
```

Figure 4: Estimation Process

is shown in Fig. 4. *estimated* is the Boolean variable that indicates successful estimation. *npcomponents* is the number of single Poisson components in the nP distribution. *nattempts* is the number of the estimation attempts.

4.2.1 Accuracy of the Estimation Algorithm

The accuracy of this randomised estimation approach was determined experimentally by generating various nP distributions and then approximating their parameters. The results are summarised in Fig. 5.

These results were obtained by first generating 1000 random nP distributions for each n in nP and each sample size class, and then approximating their parameters. The table shows that for sample sizes of over 500 (for the tokens that occur in 500 documents or more in the collection) the accuracy of our estimation process is greater than 95%. Therefore the chance of failing to recognise an nP token is less than 5%.

4.2.2 Efficiency of the Estimation Algorithm

The analysis and the experimental measurements indicate that the time complexity of the algorithm is in the worst case proportional to the product of the sample size (*ndoc* in our case) and the number of iterations required by the MLE algorithm. For sample sizes under 20000, the time complexity is proportional to the square of the sample size in the worst case. The following discussion of the time complexity of the algorithm is based on a number of measurements performed during the accuracy testing described above. The results[5] are shown in Figures 6*a*,6*b* and 6*c*.

The time complexity of the algorithm can be described by the following formula:

$$NT \cdot NA \cdot (ET + OT) + OA$$

where NT is the number of tokens to be processed; NA is the average number of estimation attempts per token; ET is the time complexity of the parameter estimation process; OT is the handling overhead per term and OA is the handling overhead per document collection.

NT is the number of tokens to be processed and is equal to the number of unique tokens in the collection – *ntok*. This number grows with the collection size, but asymptotically it approaches the number of the unique tokens in the collection vocabulary. Therefore, NT is asymptotically a constant, albeit large. For smaller collections, however, NT is equal to *ntok*.

NA is the average number of estimation attempts required per term. This number is computed by the *nattempts* variable (Fig. 4). Large samples provide more information about the underlying distribution, thus estimation of the distribution parameters for large samples is more likely to be correct the first time. Therefore, as the sample size increases, the number of estimation attempts decreases and asymptotically approaches 1. This suggestion is supported by the graphs in Fig. 6*a* that show the relation between *nattempts* (Y-axis) and the sample size (X-axis) for various nP distributions.

ET is the time complexity of the MLE algorithm used for the parameter estimation of nP distribution. From the description of the algorithm (Agha & Ibrahim, 1984) we determine that its time complexity is proportional to the product of the number of single Poisson components in the distribution (N), the sample size (S), and the number of iterations (I) required to compute the maximum likelihood estimate of the nP parameters:

$$ET = N \cdot S \cdot I$$

Although the number of single Poisson components may vary, in our case it is bounded by the maximum number of Poisson components in the nP distribution that we recognise (which is equal to 8). Thus N is a constant. The sample size is the number of documents where a given token occurs and is proportional to the total number of documents in the collection[6]: $S = O(ndoc)$.

[5] The efficiency results for sample sizes of 3000 and under are based on the execution of 1000 tests for each sample size and every n in nP. For sample sizes of over 3000 the results are based on the execution of 200 tests for each sample size and every n in nP.

[6] The sample size, although proportional to *ndoc*, is usually much smaller than *ndoc*. It is equal to *ndoc* only for the tokens that occur in every document in the collection.

	Sample Size	1P	2P	3P	4P	5P	6P
P(successful estimation)	100	0.993	0.862	0.830	0.788	0.772	0.729
	500	0.998	0.957	0.960	0.951	0.963	0.970
	1000	1.000	0.968	0.965	0.976	0.978	0.986
	3000	1.000	0.983	0.981	0.982	0.985	0.992
P(successful estimation on 1st attempt)	100	0.993	0.841	0.769	0.701	0.647	0.569
	500	0.998	0.943	0.917	0.917	0.925	0.943
	1000	1.000	0.952	0.925	0.937	0.949	0.957
	3000	1.000	0.967	0.929	0.929	0.938	0.947

Figure 5: Accuracy of the Estimation Algorithm

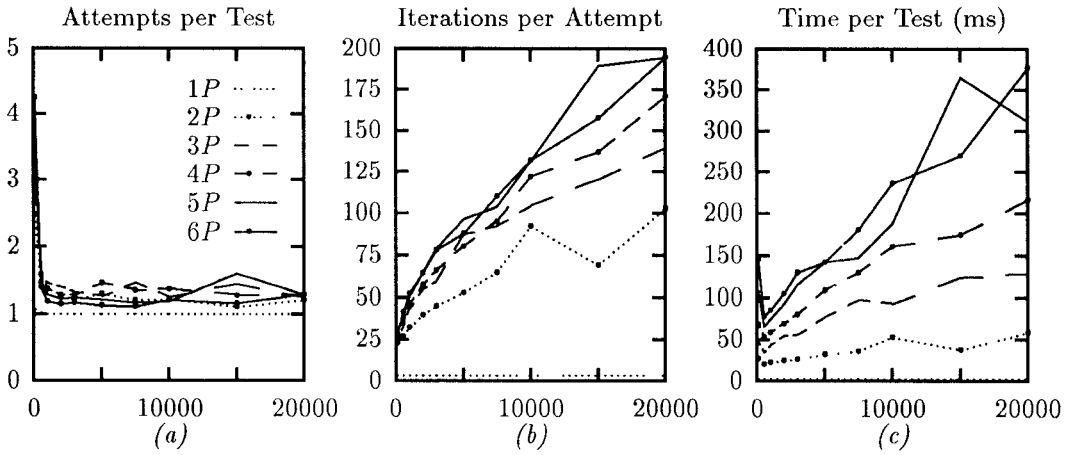

Figure 6: Efficiency of the Estimation Algorithm

I is the number of iterations within the MLE algorithm. To the best of our knowledge there is no analytical study of the relationship between the number of iterations and the sample size for the MLE algorithm. During the accuracy testing we also monitored the number of iterations within this algorithm. The results are shown in Fig. 6*b* where Y-axis represents the average number of iterations per estimation attempt and X-axis represents the sample size. The results indicate that estimation of $1P$ distribution parameters requires a constant number of iterations independent of the sample size. For nP distributions with $n > 1$, the results indicate that within our sample size range (100 to 20000) the number of iterations required is not worse than linear with respect to the sample size: $I = O(S) = O(ndoc)$. We suspect that the relationship between the number of iterations and the sample size is not worse than linear even for the larger samples. Therefore the time complexity of the MLE algorithm is:

$$\begin{aligned} ET &= C \cdot O(ndoc) \cdot I = O(ndoc) \cdot I \\ &\qquad \text{for } ndoc > 20000 \\ &= C \cdot O(ndoc) \cdot O(ndoc) = O(ndoc^2) \\ &\qquad \text{for } 100 < ndoc < 20000 \end{aligned}$$

OT is the handling overhead per token during the estimation process. This overhead includes data table mangement, χ^2 test, and the generation of the randomised initial values of nP parameters. The time complexity of the data table management and the χ^2 test is proportional to the sample size. The time complexity of the the generation of the initial values is proportional to the number of single Poisson components in nP distribution and is bounded by a constant – the maximum number of such components recognised. Therefore $OT = O(ndoc)$.

The graphs in Fig. 6*c* show the average amount of CPU time (in *ms*) that was spent in the inner **while** loop of the algorithm (Fig. 4). The complexity of this portion of the algorithm can be described as: $NA \cdot (ET + OT)$. The Y-axis represents the time in milliseconds and the X-axis represents the sample size. These graphs confirm that the time complexity of the algorithm depends on NA (Fig. 6*a*) and $(ET + OT)$ (I, the main component of ET, is shown in Fig. 6*b*)

OA is the handling overhead per document collection and is proportional to the management of two tables: document table and token table. The size of the token table is the number of unique tokens and is asymptotically a constant. The size of the document table is proportional to the number of documents. Therefore the handling overhead is: $OA = O(ndoc)$.

Substituting the values of NT, NA, ET, OT, and

OA in the original complexity formula, we determine that the complexity of the algorithm is:

$$
\begin{aligned}
& NT \cdot NA \cdot (ET + OT) + OA = \\
&= C \cdot C \cdot (O(ndoc) \cdot I + O(ndoc)) + O(ndoc) \\
&= O(ndoc) \cdot I \text{ for } ndoc > 20000 \\
&= O(ndoc^2) \text{ for } 100 < ndoc < 20000
\end{aligned}
$$

ntok	*ndoc*	*Time*	Collection
2062	1084	46	*INYT_t*
2678	1034	55	*RAMR_t*
2419	1084	57	*INYT_w*
3154	1034	75	*RAMR_w*
6540	3269	377	$WS87\_t_1$
6582	3316	399	$WS87\_t_2$
7917	6750	609	*FT85_t*
11099	11212	1049	$WS87\_t_3$
11034	8151	1121	$WS87\_t_4$
10012	5177	1167	$WS87\_t_6$
12127	12463	1333	$WS87\_t_5$
12294	9111	1571	$WS87\_t_7$
13641	13938	1967	*WS87_t*
20565	13938	2001	*WS87_w*

Figure 7: Processing Times

Note that it is more expensive (time-wise) for the estimation process to fail than to succeed: if a token is not nP then the algorithm would fail only after exhausting all attempts (up to 10) for all n in nP (up to 8). If a token is nP, then it is likely to be recognised early in the process since the probability of successful estimation on the first attempt is relatively high (Fig. 5). Therefore, the proportion of nP tokens as well as the composition of various nP functions affects the performance of the algorithm for a given collection.

The initial experimental results on the test collections used in this study support the timing complexity analysis above. The timing results of applying the algorithm to various collections are shown in Fig. 7. In this table the time is given in CPU seconds; "*_t*" and "*_w*" suffixes after the collection name indicate whether terms or words were processed; subscripts denote various subsets of the original collections. The sizes of collections tested are relatively small to assume that the number of unique tokens in the collection is a constant (see *ntok* column in Fig. 7), thus $NT = O(ntok)$. Therefore the time complexity of the algorithm for our test collections is expressed as: $O(ntok \cdot ndoc^2)$.

The graph in Fig. 8 supports the complexity analysis. It shows that for the tests performed in this study, the time complexity of the algorithm is not worse than $O(ntok \cdot ndoc^2)$.

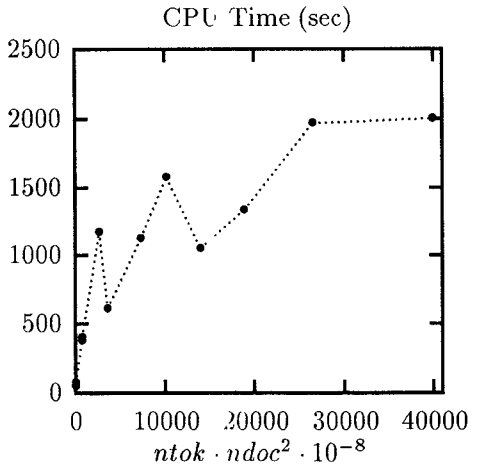

Figure 8: Processing Times

5 Experiment Results

The goal of our experiment was to test assumption 3.1. First we describe the results of the basic nP analysis to verify the assumption 3.1 with respect to words and terms. The results indicate that most tokens that occur frequently are indeed nP.

The second part of the experiment is the initial analysis of the composition of nP tokens. Here we investigate what proportion of all nP tokens are $1P$, $2P$, etc. The results of this part of the experiment suggest that the composition of nP tokens in a collection depends on the token frequency and the collection size.

5.1 Basic nP Analysis

The first and the main phase of the experiment was to test the assumption 3.1 by analysing the *WS87*, *FT85*, *RAMR* and *INYT* collections. The result of the analysis is not a binary "yes" or "no" answer, but depends on the collection frequency of tokens analysed. Large samples of data result in better statistical analysis. The more often a token occurs in the collection, the more likely its frequency is high enough for the χ^2 goodness of fit test. We represent the results of the study as a 2D graph where the X axis is the collection frequency of tokens and the Y axis is the percentage of nP tokens whose frequency is greater than x:

$$
y = \frac{|TNP|}{|T|} \cdot 100\%, \qquad T = \{h | cfreq(\mathbf{C}, h) > x\}
$$

$$
TNP = \{h | h \in T \wedge np(h)\}
$$

The graphs in Figures 9 (terms) and 10 (words) represent the main results of this study. The graphs show

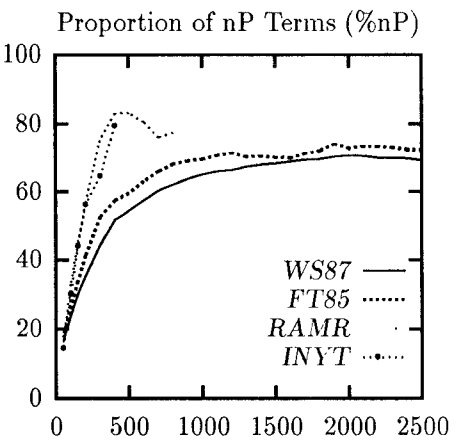

Figure 9: Proportion of nP terms

that over 65% of terms or words that occur more than 1000 times in the two larger collections (*FT85*, *WS87*) are nP. In smaller collections (*RAMR* and *INYT*) over 80% of terms occurring more than 500 times are nP.

Examining the graphs in Figures 9 and 10 one can detect an important similarity between the proportion of nP terms (Fig. 9) and the proportion of nP words (Fig. 10). This similarity is not surprising, since the terms are derived from words via word reduction (stemming) process and elimination of stop words. This similarity indicates that the word reduction process does not significantly affect the proportion of nP tokens in the collection.

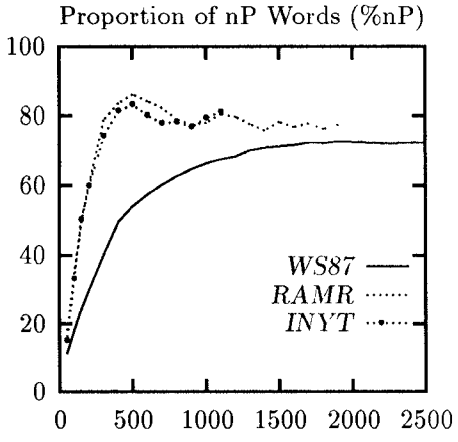

Figure 10: Proportion of nP tokens

The main difference between the word based nP analysis and the term based nP analysis is demonstrated by examination of the results for the smaller collections (*INYT* and *RAMR*). The graphs suggest that most frequently occurring stop words are nP. The right portions of the nP words proportion graphs for these collections (Fig. 10) reflect the proportion of the frequently occurring stop words which are excluded from the term based analysis (that is why the *INYT* and *RAMR* graphs are significantly shorter in Fig. 9).

The results also suggest a relation between the collection size and the proportion of the nP tokens. Consider graphs for smaller collections (*INYT* and *RAMR*) in Fig. 9 and 10. These graphs display higher proportion of nP tokens (words and terms) than the graphs for the larger collections (*FT85* and *WS87*). *WS87* collection is larger than the *FT85* collection and the graph showing the proportion of nP terms in *WS87* is very close to the *FT85* but is consistently below the *FT85* graph (Fig. 9). This examination suggests that the smaller document collections have higher proportion of nP tokens. More analysis is required to explain this phenomenon analytically. Note that for the very small collections this suggestion does not hold - in such collections tokens do not occur frequently enough to facilitate any reasonable nP testing.

The graphs in the Figures 9 and 10 provide positive evidence that strongly supports assumption 3.1: approximately 70% of the frequently occurring terms in document collections are nP. It is possible that a high proportion of terms occurring less frequently are also nP, but their low frequency prevents us from testing the goodness of the parameter estimation with the χ^2 test.

The results of this part of the study suggest that:

- Approximately 70% of the frequently occurring terms and words in document collections are nP.
- The proportion of nP terms is similar to the proportion of nP words.
- The proportion of nP tokens is higher in the smaller collections and is lower in the larger collections.

5.2 Analysis of Collections of Similar Length Documents

In this section we describe the results of nP analysis of collection of documents of similar length. For this purpose we have created four collection subsets: $FT85_w$, $FT85_t$, $WS87_w$ and $WS87_t$. These subcollections contain documents with similar *tlen* and similar *wlen*. The table in Fig. 11 shows the criteria for selecting documents in these subset collections. The intervals for *wlen* and *tlen* were chosen around the average values of *wlen* and *tlen* for each collection so that the subcollections contain a "reasonably large" number of documents. The proportion of nP tokens in these subcollec-

Collection	Selection Criterion	*ndoc*
$FT85_w$	$672 < wlen < 872$	1521
$FT85_t$	$300 < tlen < 400$	1729
$WS87_w$	$850 < wlen < 1050$	3269
$WS87_t$	$450 < tlen < 550$	3316

Figure 11: Similar Document Length Collections

tions was compared with the proportion of nP tokens in the similar size subcollections of randomly chosen documents from *FT85* and *WS87*, namely $FT85_s$ containing 1650 documents and $WS87_s$ containing 3300 documents. The results of this analysis are shown in Figures 12 and 13.

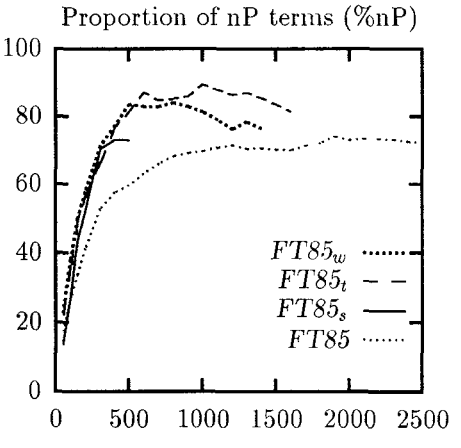

Figure 12: Similar Document Length *FT85* Subset

The graphs indicate that the collections of documents of similar lengths (either *tlen* or *wlen*) have a higher proportion of nP terms than the similar size collections of documents of different lengths. Although the subcollections do not contain documents of strictly equal lengths, the lengths within the subcollections $FT85_{w,t}$ and $WS87_{w,t}$ are more uniform than in $FT85_s$ and $WS87_s$ or the original *FT85* and *WS87* collections and we suspect that the results of the analysis of the subcollections show the general trend.

$FT85_t$ and $FT85_w$ (Fig. 12) as well as $WS87_t$ and $WS87_w$ (Fig. 13) have very similar proportions of nP terms as indicated by the closeness of the graphs representing the nP analysis of the subcollections. This suggests that the proportion of nP terms in the collections of documents with similar *tlen* is very close to that in the collections of documents with similar *wlen*.

The results described above are obtained by analysing the nP proportion of *terms* in collections of similar

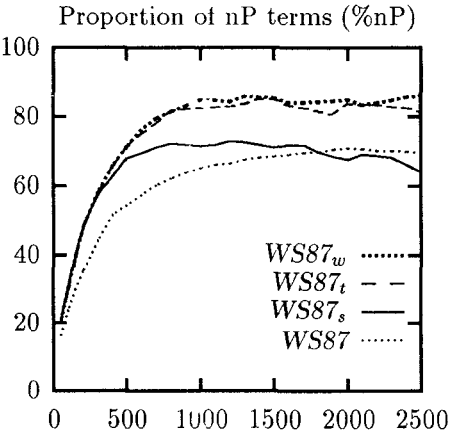

Figure 13: Similar Document Length *WS87* Subset

length documents. Other tests analysing nP proportions of *words* in collections of similar length documents indicate that the proportion of nP words is also higher in collections of documents of similar lengths.

The results of the analysis of collections of similar length documents are summarised as follows:

- Collections of documents with similar lengths (either *tlen* or *wlen*) have higher proportion of nP terms (or words) than the collections of documents with different lengths.
- Collections of documents with similar *tlen* have proportion of nP terms corresponding to that in the collections of documents with similar *wlen*.

5.3 Composition of nP Distribution Functions

In this section we describe the analysis of the composition of the nP distribution functions: the proportions of $1P$, $2P$, $3P$, etc. tokens among all nP tokens in a document collection. The results of this part of the study are presented as bar-graphs showing the proportions of nP terms with a specific n with respect to all nP tokens. The proportions are shown for all token frequencies ($cfreq$) using the intervals of 100. For each n in nP, the graph is formally described as follows:

$$y = \frac{|TnP|}{|TNP|} \cdot 100\%$$

$$T = \{h | low100(cfreq(\mathbf{C}, h)) \leq x < low100(cfreq(\mathbf{C}, h)) + 100\}$$

$$low100(z) = z - (z \; rem \; 100)$$

$$TNP = \{h | h \in \; ? \wedge np(h)\}$$

$$TnP = \{h | h \in TNP \wedge N = n\}$$

The diagrams in Figures 14, 15 and 17 show the proportions of specific nP terms in *FT85*, *WS87*, *RAMR* and *INYT* collections. The diagrams in Figures 16 and 18 show the proportions of specific nP words in *WS87*, *RAMR* and *INYT* collections. The proportion of $6P$ tokens is not shown since there are only two $6P$ terms found in the *FT85* collection; only 11 $6P$ terms and 6 $6P$ words found in the *WS87* collections.

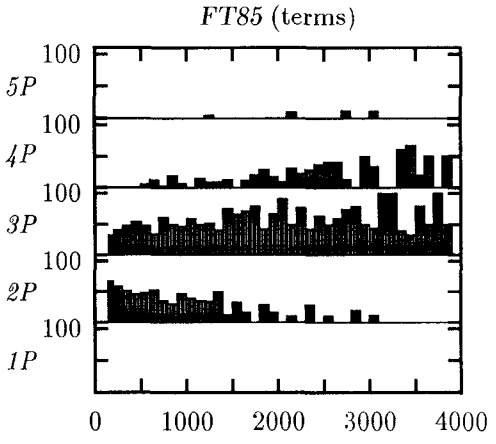

Figure 14: *FT85* nP composition (terms)

The diagrams show that the n in nP tokens is relatively low: most nP tokens are $2P$, $3P$ and $4P$. A small number of $5P$ and $6P$ tokens was found in the two larger collections (*FT85* and *WS87*) and a small number of $1P$ tokens was found in the two smaller collections (*RAMR* and *INYT*). We can therefore conclude that most terms in the collections examined can be described by $2P$, $3P$ or $4P$ distributions.

High frequency nP tokens tend to be the tokens with a relatively high value of the n in nP. For example, in *FT85* collection (Fig. 14) there are no $2P$ terms among the terms whose collection frequency is higher than 3100 and the proportion of $4P$ terms among the nP terms with frequencies less than 2000 is relatively low. Examination of term and word distributions in the *WS87*, *RAMR* and *INYT* collections (Figures 15, 16, 17 and 18) confirms the *FT85* analysis.

The results also suggest that the two larger collections (*FT85* and *WS87*) contain higher proportion of nP tokens with relatively large n in nP than the two smaller collections (*RAMR* and *INYT*). The smaller collections, on the other hand, have higher proportion of nP tokens with relatively small n: both *RAMR* and *INYT* have a significantly higher proportion of $1P$ and $2P$ tokens than the two larger collections. This suggests that

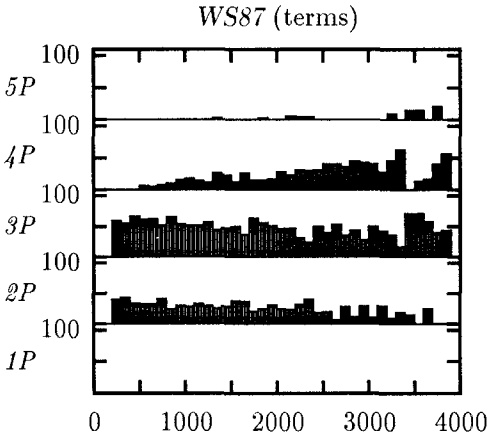

Figure 15: *WS87* nP composition (terms)

the collection size affects the nP composition of tokens in the collections: large document collections are more likely to have nP tokens with large n. nP tokens with small n are more likely to occur in the small document collections.

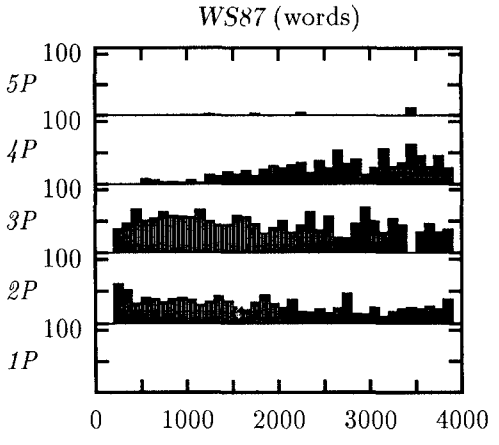

Figure 16: *WS87* nP composition (words)

Another issue related to the composition of the nP distribution functions is the lack of $1P$ tokens (terms or words) in the *FT85* and *WS87* collections (Figures 14, 15 and 16). This seems to contradict the suggestion of Damerau (1965) and Stone and Rubinoff (1968) that the stop words are distributed according to a single Poisson distribution ($1P$).

Although the most common stop words were excluded from the term based analysis (Figures 14 and 15) it is very unlikely that we have excluded *all* of the stop words.

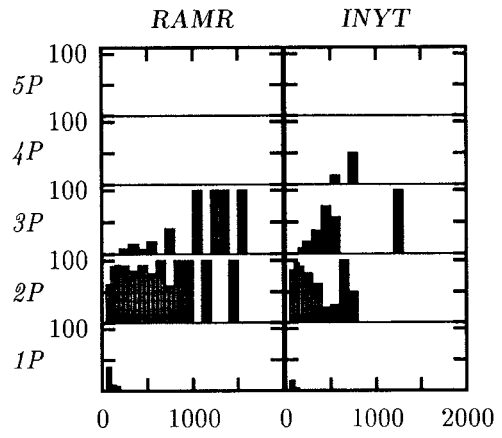

Figure 17: *RAMR, INYT* nP composition (terms)

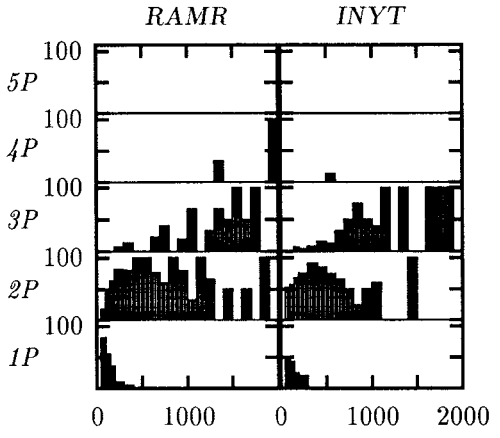

Figure 18: *RAMR, INYT* nP composition (words)

No stop words were excluded from the word based analysis (Fig. 16) These stop words should have been classified as $1P$ tokens. The lack of $1P$ terms in the two larger collections and the small number of $1P$ terms in the two smaller collections can be explained as follows.

In large collections documents are written using different writing styles and deal with a wide variety of topics. This variety of writing styles and topics introduces a certain "noise" in the otherwise perfect $1P$ distribution. If the collection size is relatively small then the noise is low and it is possible to recognise the stop word as a $1P$ term (that is why there are still some $1P$ tokens in the *RAMR* and *INYT* collections). If the collection size is large then the effects of the noise are strong and the stop words are either recognised as nP terms with $n > 1$ or not recognised as nP terms at all.

The main results of this portion of the study are summarised as follows:

- The distribution of most of the nP tokens have relatively few single Poisson components: 2, 3 or 4.
- Composition of nP terms in a collection is similar to the composition of nP words in the same collection.
- High frequency nP terms tend to be the terms with a relatively high value of n in nP.
- Large document collections are more likely to have nP tokens with the large n and less likely to have nP tokens with the small n.
- Stop words are unlikely to be distributed according to $1P$ in large collections.

6 Summary

In this paper we described the results of the recent study of the n-Poisson distribution of words and terms in full text document collections. We presented a practical algorithm for determining if a certain token is distributed according to an nP distribution. The time complexity of the algorithm is in the worst case asymptotically proportional to the product of the number of documents and the number of iterations required by the maximum likelihood estimation algorithm. For collections where individual terms occur in less than 20000 documents, the time complexity is in the worst case quadratic with respect to the number of documents.

The algorithm was applied to test the nP properties of every word and term in four full text document collections. We determined that over 70% of frequently occurring words and terms are indeed distributed according to the Multiple Poisson distribution.

The analysis of composition of the nP distribution functions was performed to determine the proportion of the specific n-Poisson distributions ($1P$, $2P$, etc) among all of the nP tokens in our collections. The distributions of most of the nP tokens were found to have relatively few single Poisson components: two, three, or four. The results also suggest the relation between the nP composition and the collection size and the token frequency.

In this study we presented strong evidence that the n-Poisson distribution could be used as a basis for an accurate and useful statistical model of large document collections. There are many open questions, however, and further research is required for the development of such a model. The following are some of the issues that would require further analysis:

- Determining if the n-Poisson distribution parameters of individual text tokens can be used as characteristics of document collections.
- Determining if not only words and terms, but other

text units as well behave according to the n-Poisson distribution: phrases, n-grams, etc.
- Investigating the n-Poisson properties of non-English language document collections.
- Correlating nP properties of tokens with their "traditional" IR characteristics: inverse document frequency, discrimination value, etc.

Another important issue for future research is the development of an nP indexing and retrieval strategy. This strategy is based on the following. Each nP token divides a document collection into n subsets corresponding to n classes of coverage. Therefore each nP token is associated with $(n-1)$ "division points". These division points can be easily computed. We believe that these points can be used for determining good index terms. This approach is based on the sound statistical analysis and we feel it will improve the efficiency of the search and retrieval process.

We believe that further research in using the n-Poisson distribution to model full text document collections will help in developing a comprehensive and useful statistical model for such collections.

Acknowledgements

I would like to thank Dr. M. Agha for his advice on the estimation procedures; Prof. H. P. Frei and Dr. P. Schauble for their comments on the previous drafts; Iain Campbell for many discussions of the problem of document collection modelling, comments on the previous drafts and help with the test collection data processing.

References

Agha, M., & Ibrahim, M. T. (1984). Maximum Likelihood Estimation of Mixtures of Distributions. *Applied Statistics, Journal of the Royal Statistical Society*, 33,327–332.

Bärtschi, M. (1985). An Overview of Information Retrieval Subjects. *Computer*, 18(5),67–84.

Bookstein, A., & Swanson, D. (1974). Probabilistic Models for Automatic Indexing. *Journal of the American Society for Information Science*, 25(5),312–318.

Breiman, L. (1973). *Statistics: With a View Towards Applications*. Houghton-Mifflin Company.

Damerau, F. J. (1965). An Experiment in Automatic Indexing. *American Documentation*, 16(4).

Giford, D. K. (1990). Polychannel Systems for Mass Digital Communication. *Communications of the ACM*, 33(2),141–151.

Harter, S. P. (1975). A Probabilistic Approach to Automatic Keyword Indexing: Part I. *Journal of the American Society for Information Science*, 26(4),197–206.

Hasselblad, V. (1969). Estimation of Finite Mixtures of Distributions from the Exponential Family. *American Statistical Association Journal*, 64,1459–1471.

Karp, R. M. (1990). An Introduction to Randomized Algorithms. Technical Report TR-90-024, Computer Science Division, University of California, Berkeley.

Luhn, H. P. (1958). The Automatic Creation of Literature Abstracts. *IBM Journal of Research and Development*, 2,156-165.

Margulis, E. L. (1991). N-Poisson Document Modelling Revisited. Technical Report 166, ETH-Zürich, Departement Informatik.

Porter, M. (1980). An Algorithm for Suffix Stripping. *Program*, 24(3),130–137.

Robertson, S. E., van Rijsbergen, C. J., & Porter, M. F. (1981). Probabilistic models of indexing and searching. In Oddy, R. N. Robertson, S. E., van Rijsbergen, C. J., & Williams, P. W., editors, *Information Retrieval Research*. Butterworth & Co Ltd.

Srinivasan, P. (1990). On Generalizing the Two-Poisson Model. *Journal of the American Society for Information Science*, 41(1),61–66.

Stone, D. C., & Rubinoff, M. (1968). Statistical Generation of a Technical Vocabulary. *American Documentation*, 19(4).