Department of Information and Computer Science

Computational methods for comparison and exploration of event sequences

Jefrey Lijffijt

Aalto University
DOCTORAL DISSERTATIONS

Aalto University publication series
DOCTORAL DISSERTATIONS 205/2013

Computational methods for comparison and exploration of event sequences

Jefrey Lijffijt

A doctoral dissertation completed for the degree of Doctor of Science (Technology) to be defended, with the permission of the Aalto University School of Science, at a public examination held at the lecture hall T2 of the school on 16 December 2013 at 10 am.

Aalto University
School of Science
Department of Information and Computer Science

Supervising professor
Prof. Juho Rousu

Thesis advisor
Prof. Heikki Mannila

Preliminary examiners
Prof. Floris Geerts, Universiteit Antwerpen, Belgium
Prof. Jean-François Boulicaut, Institut National des Sciences Appliquées de Lyon, France

Opponent
Prof. Bart Goethals, Universiteit Antwerpen, Belgium

Aalto University publication series
DOCTORAL DISSERTATIONS 205/2013

ISBN 978-952-60-5474-2
ISBN 978-952-60-5475-9 (pdf)
ISSN-L 1799-4934
ISSN 1799-4934 (printed)
ISSN 1799-4942 (pdf)
http://urn.fi/URN:ISBN:978-952-60-5475-9

Unigrafia Oy
Helsinki 2013

Finland

441 697
Printed matter

Aalto University

Abstract

Aalto University, P.O. Box 11000, FI-00076 Aalto www.aalto.fi

Author
Jefrey Lijffijt

Name of the doctoral dissertation
Computational methods for comparison and exploration of event sequences

Publisher School of Science

Unit Department of Information and Computer Science

Series Aalto University publication series DOCTORAL DISSERTATIONS 205/2013

Field of research Information and Computer Science

Manuscript submitted 9 September 2013 **Date of the defence** 16 December 2013

Permission to publish granted (date) 4 November 2013 **Language** English

☒ **Monograph** ☐ **Article dissertation (summary + original articles)**

Abstract

Many types of data, e.g., natural language texts, biological sequences, or time series of sensor data, contain sequential structure. Analysis of such sequential structure is interesting for various reasons, for example, to detect that data consists of several homogeneous parts, that data contains certain recurring patterns, or to find parts that are different or surprising compared to the rest of the data. The main question studied in this thesis is how to identify global and local patterns in event sequences. Within this broad topic, we study several subproblems.

The first problem that we address is how to compare event frequencies across event sequences and databases of event sequences. Such comparisons are relevant, for example, to linguists who are interested in comparing word counts between two corpora to identify linguistic differences, e.g., between groups of speakers, or language change over time. The second problem that we address is how to find areas in an event sequence where an event has a surprisingly high or low frequency. More specifically, we study how to take into account the multiple testing problem when looking for local frequency deviations in event sequences. Many algorithms for finding local patterns in event sequences require that the person applying the algorithm chooses the level of granularity at which the algorithm operates, and it is often not clear how to choose that level. The third problem that we address is which granularities to use when looking for local patterns in an event sequence.

The main contributions of this thesis are computational methods that can be used to compare and explore (databases of) event sequences with high computational efficiency, increased accuracy, and that offer new perspectives on the sequential structure of data. Furthermore, we illustrate how the proposed methods can be applied to solve practical data analysis tasks, and describe several experiments and case studies where the methods are applied on various types of data. The primary focus is on natural language texts, but we also study DNA sequences and sensor data. We find that the methods work well in practice and that they can efficiently uncover various types of interesting patterns in the data.

Keywords pattern mining, event sequence, statistical significance, multiple testing, sliding window, window length

ISBN (printed) 978-952-60-5474-2 **ISBN (pdf)** 978-952-60-5475-9

ISSN-L 1799-4934 **ISSN (printed)** 1799-4934 **ISSN (pdf)** 1799-4942

Location of publisher Helsinki **Location of printing** Helsinki **Year** 2013

Pages 112 **urn** http://urn.fi/URN:ISBN:978-952-60-5475-9

Preface

I am deeply grateful to Prof. Heikki Mannila, for supervising my thesis and research for the past years, for his ability to always ask the right questions, and for his invaluable advice and encouragement. I am also grateful to Dr. Kai Puolamäki and Dr. Panagiotis Papapetrou for all the white-board sessions, discussions, lunches and many hours that we have spent together, and to Prof. Juho Rousu for his supervision of the final steps of my dissertation. I thank Prof. Terttu Nevalainen, Ms. Tanja Säily, and Mr. Turo Vartiainen for introducing me to the exciting field of corpus linguistics and thereby providing me with an endless source of computational problems. I thank the pre-examiners of this thesis, Prof. Floris Geerts and Prof. Jean-François Boulicaut, and my FICS-mentors, Prof. Hannu Toivonen and Dr. Salme Kärkkäinen, for their useful and valuable comments. I would like to thank all my colleagues, including the support staff, at the ICS Department, ALGODAN, HIIT, FICS, and the Aalto Doctoral Programme, for all the fun and the inspiring working environment. I gratefully acknowledge the funding provided by the Finnish Doctoral Programme in Computational Sciences (personal grant), the Academy of Finland (grants 118653 and 129282), and the EU PASCAL Network. I would like to thank my father for perpetually stimulating me to learn, and my friends and family for their support and happiness that they have provided me throughout the years. I thank Seppe for the fun that we have and Lotte for her high spirits, support and encouragement.

Espoo, November 20, 2013,

Jefrey Lijffijt

Contents

1. Introduction

1.1 Motivation

This thesis considers the problem of finding global and local patterns in event sequences. An event sequence is a sequence of event labels and can be used to represent the structure of an object or a series of events, such as a novel or a deoxyribonucleic acid (DNA) molecule. For example, $(A, C, T, G, G, C, G, G, A, T, T, A)$ is an event sequence with event labels A, C, G, and T that represents a part of the structure of a DNA molecule. A text may be represented as an event sequence by mapping the words to events, e.g., $(this, thesis, considers, the, problem, of, finding, global, and, local, patterns, in, event, sequences)$ is an event sequence, where the event labels are $this$, $thesis$, $considers$, etc.

Analysis of such event sequences is interesting for many reasons. For example, Mannila et al. [1997] study frequent patterns in alarm logs from a telecommunication network to discover relations in time between the alarms. Salmenkivi and Mannila [2005] study segmentation of similar alarm logs to detect variation over time in the frequency of events. Haiminen et al. [2008] study segmentation of DNA sequences in order to detect transcription factor binding sites. Hearst [1994] studies segmentation of text into coherent discourse units corresponding to subtopics, and many more applications have been considered.

The main question studied in this thesis is *how to identify global and local patterns in event sequences*. Within this topic, the following subproblems are addressed. One of the principal properties of an event sequence is the frequency at which the events occur in the sequence. Comparison of event sequences by comparing the frequencies at which the events occur can be a useful source of information. Such comparisons are, for

example, frequently employed by linguists. Word counts between two corpora are compared to identify linguistic differences, e.g., between groups of speakers [Rayson et al., 1997], varieties of the same language [Oakes and Farrow, 2007], or language change over time [Baker, 2011].

Typically, one assumes that the event sequences are generated by a stochastic process and that the observations are a sample from this process. The aim is to make inferences about the underlying stochastic processes, and not just to compare the counts observed in the samples. A statistical test can be used to assess the statistical significance of an observed difference, i.e., to compute a p-value for an observed frequency difference. If the p-value is very low, it is unlikely that the two generative processes are the same, and it is concluded that the difference is *significant*. The first question addressed in this thesis is which statistical test is most appropriate in this setting. In other words,

Question 1. *How to compare event frequencies across (databases of) event sequences?*

The average frequency of an event is a *global* pattern, and it may also be useful to identify *local* patterns in event sequences. For example, there may be dependencies between consecutive occurrences of an event, or the frequency of an event may change over time within an event sequence, causing the occurrence pattern of an event to have a non-uniform, or *bursty*, distribution. A prime question is how to identify such local patterns, i.e., how to find areas in the sequence where the event is substantially more or less frequent than in other parts.

The approach considered here is to compute the frequency of an event for all intervals of some given length, e.g., the frequency between events 1 and 1,000, events 2 and 1,001, events 3 and 1,002, etc. This technique is also known as the *sliding window* method. Then, for each interval (*window*), the statistical significance (p-value) is assessed using a statistical test. A problem with this approach is that one cannot conclude that there is local structure based on a single low p-value. Since many tests are conducted simultaneously, it is very likely that low p-values are observed.

In such *multiple testing* scenarios, a post-hoc correction can be used to adjust the p-values to make them easier to interpret. Several types of adjustments have been proposed, for example to guarantee that, when there is no additional structure, i.e., when the null hypothesis is true, at most one p-value is expected to be below a given threshold α. This guarantee is known as *control for the family-wise error rate at level* α.

In the sliding window setting, the intervals are overlapping, thus the p-values are highly dependent. It is preferable to take this into account in the post-hoc correction to prevent that the p-values become conservative, i.e., too high, and that structure present in the data is overlooked. The second question addressed in this thesis is aimed at this problem:

Question 2. *How to take into account the multiple testing problem when looking for local frequency deviations in event sequences?*

There are many algorithms for finding local patterns in event sequences or time series that use a sliding window, for example, for detecting bursts [Zhu and Shasha, 2003], detecting change-points [Kifer et al., 2004], or mining frequent patterns [Lin et al., 2005]. Often, the user has to choose the level of granularity at which the algorithm operates, and it is not clear how to choose that level, e.g., would it be optimal to use windows of length 100 events, 1,000 events, or something else?

Additionally, a single granularity does often not provide all the information that a user is interested in. Thus, it is often preferable to analyse event sequences using multiple window lengths concurrently. The question then arises how to select the best set of window lengths, best meaning most appropriate for the task at hand. The third question addressed in this thesis targets this problem:

Question 3. *Which granularities to use when looking for local patterns in an event sequence?*

The main question, *how to identify global and local patterns in event sequences*, is not answered completely in this thesis, but the three subproblems that are outlined above are addressed thoroughly. The main contributions of this thesis are computational methods that are faster, more accurate, and provide new information about event sequences. Each of the methods is also tested in practice, and we illustrate how empirical questions can be answered using the methods. The contributions made in this thesis are detailed further in the following section.

1.2 Contributions

The main problem considered in this thesis is *how to find global and local patterns in event sequences*. Several aspects of this general problem are considered, specifically: (1) *how to compare event frequencies across*

(databases of) event sequences, (2) *how to take into account the multiple testing problem when looking for local frequency deviations in event sequences*, and (3) *which granularities to use when looking for local patterns in an event sequence*.

Our focus is on introducing new computational methods that address each of these questions. Furthermore, we illustrate how the proposed methods can be applied to solve practical data analysis tasks, and describe several experiments and case studies where the proposed methods are applied on various types of data: texts, DNA, and sensor data. We discuss the related work and compare the proposed methods with existing methods where applicable.

In Chapter 4, we study the question *how to compare event frequencies across databases of event sequences*. By modelling texts as event sequences and a text corpus as a database, the question can be mapped to questions such as "is word X more frequent in male than in female speech?". We introduce two statistical tests based on resampling and we compare and evaluate these methods, along with several existing methods, with respect to their suitability to the task.

We find that the choice of the test, or more specifically, the representation of the data that is used in the test, matters, both in theory and in practice, as evidenced by experiments and case studies on two text corpora. We conclude that frequently applied tests may lead to overestimating the significance of frequency differences, and demonstrate that the overestimation is related to the burstiness of words. We show that there exist bursty and non-bursty words at any frequency level, thus the overestimation also occurs at all frequency levels.

In Chapter 5, we study the question *how to take into account the multiple testing problem when looking for local frequency deviations in event sequences*. We introduce a new statistical test for assessing the significance of event frequencies in subsequences when using a sliding window, which provides strong control of the family-wise error rate and takes into account the dependency structure of overlapping subsequences. We argue that the exact p-values are difficult to compute, and base the test on an easy-to-compute upper bound.

We provide empirical evidence that the test offers substantially increased power compared to existing alternatives, and demonstrate the utility and practicality of the test on linguistic and biological sequences. We identify several new and interesting patterns, and find that meaningful results

can be obtained. Moreover, we find that the method remains sufficiently powerful even when testing hundreds of millions of hypotheses.

In Chapter 6, we study the question *which granularities to use when looking for local patterns in an event sequence*. We introduce a new optimisation problem that corresponds to selecting the most informative set of window lengths. We show that the optimisation problem can be efficiently approximated algorithmically, and solved analytically for certain simple statistics and data distributions. We explore the performance of the proposed optimisation algorithm, as well as the results for several statistics on both synthetic data and real data.

We demonstrate that the analytical and empirical results on synthetic data are useful as a baseline for practical use of the method. We show that sampling can be used to compute the set of window lengths more efficiently, making the method practical for (databases of) event sequences of any size. Finally, we illustrate that the window lengths themselves can reveal interesting properties of the data; among other findings, we identify relations between the optimal window lengths and (1) the structure of sequences composed of multiple interleaved sources and (2) the burstiness of events.

In short, the methods introduced in this thesis can be used to compare and explore (databases of) event sequences with high computational efficiency, increased accuracy, and in novel ways.

1.3 Author's contributions

Chapters 1, 2, 3, and 7 have been written independently by the author of this thesis, and all the content is new. Parts of Chapter 4 have been published in the following papers:

1. J. Lijffijt, P. Papapetrou, K. Puolamäki, and H. Mannila. Analyzing word frequencies in large text corpora using inter-arrival times and bootstrapping. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*, 2011.

2. J. Lijffijt, T. Säily, and T. Nevalainen. CEECing the baseline: Lexical stability and significant change in a historical corpus. In *Outposts of Historical Corpus Linguistics: From the Helsinki Corpus to a Prolifera-*

tion of Resources (Studies in Variation, Contacts and Change in English 10), 2012.

3. J. Lijffijt, T. Nevalainen, T. Säily, P. Papapetrou, K. Puolamäki, H. Mannila. Significance testing of word frequencies in corpora. *Forthcoming*.

Sections 4.1 to 4.4, the *introduction*, *related work*, *problem setting* and *methods* sections, are inspired by all three articles, but the text is mostly new. Sections 4.5 and 4.6, containing the *experiments* and *conclusion*, are mostly based on Publication 3, while Section 4.5.3 is based on Publication 2. The author of this thesis has formulated the initial hypothesis of the chapter (that the statistical test matters, and that using the bag-of-words model leads to overestimating the statistical significance of observed differences) and is the main author of all three publications. The current author has written most of the text of Publications 1 and 3, while Tanja Säily has contributed equally towards Publication 2. However, only a small part of Publication 2 is included in this thesis. The current author has developed and implemented the new methods, has designed the experiment in Section 4.5.1, has co-designed the experiments in Sections 4.5.2 and 4.5.3, and has conducted the experiments. The analysis and interpretation of the results in the case studies (Sections 4.5.2 and 4.5.3) have been conducted by the domain experts and co-authors Tanja Säily and Terttu Nevalainen. All co-authors of the articles have participated in writing, discussions, and provided feedback during all stages of the research.

Most of the content of Chapter 5 has been published in

4. J. Lijffijt. A fast and simple method for mining subsequences with surprising event counts. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*, 2013.

The author of this thesis is the sole author of the article and the work has been conducted independently. The author's advisor, Heikki Mannila, has provided feedback during various stages of the research. The text has been partly rewritten to adhere to the style of the thesis, the division between the problem formulation and the method is new, and the discussion of the results from the experiments has been slightly expanded.

Most of the content of Chapter 6 has been published in

5. J. Lijffijt, P. Papapetrou, and K. Puolamäki. Size matters: Finding the most informative set of window lengths. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*, 2012.

6. J. Lijffijt, P. Papapetrou, and K. Puolamäki. The k-windows problem: Finding the most informative set of window lengths. *Forthcoming*.

Publication 6 is an expanded version of Publication 5. The text has been partly rewritten to adhere to the style of the thesis. The initial hypothesis was formulated together with the co-authors Panagiotis Papapetrou and Kai Puolamäki. The current author has written most of the text of the articles, has developed and implemented the methods, has designed and conducted all the experiments presented in this thesis, and has analysed and interpreted the results of the experiments. The proof for an analytical solution in a special case was first derived by Kai Puolamäki, and both Panagiotis Papapetrou and Kai Puolamäki have participated in writing, discussions and provided feedback during all stages of the research.

1.4 Outline

This thesis is structured as follows. Basic definitions are presented in Chapter 2. However, these need not be read integrally, the reader is referred to the appropriate sections of Chapter 2 where necessary. Similarly, to avoid repetition, general introductions to all data sets used in this thesis are given in Chapter 3. These need not be read in advance either.

The three subproblems are each addressed in a separate chapter. The question *how to compare event frequencies across (databases of) event sequences* is addressed in Chapter 4. The question *how to take into account the multiple testing problem when looking for local frequency deviations in event sequences* is addressed in Chapter 5, and the question *which granularities to use when looking for local patterns in an event sequence* is addressed in Chapter 6. These chapters are self-contained and can be read without reading the others. Chapter 7 contains an overview and discussion of the main conclusions.

2. Preliminaries

This chapter introduces basic notation and definitions related to event sequences (Section 2.1), inter-arrival times of events (Section 2.2), burstiness and dispersion of events (Section 2.3), statistical significance testing (Section 2.4), and testing multiple hypotheses (Section 2.5).

All of the following definitions are used in multiple chapters and therefore collected in this chapter. However, it is not necessary to read these definitions in advance. The reader is referred back to specific sections of this chapter when the corresponding notation and definitions are used.

2.1 Event sequences

Definition 2.1 (Event sequence)**.** *Given a set of event labels L, an event sequence S of length n is defined as*

$$S = (s_1, \ldots, s_n)\,, \text{ where } s_i \in L \text{ for all } i \in \{1, \ldots, n\}\,.$$

For example, $S = (a, b, c, b)$ is an event sequence of length 4 with event labels $L = \{a, b, c\}$.

Definition 2.2 (Subsequence)**.** *Given an event sequence S of length n, the subsequence $S_{i,m}$ starting at position i with length m and $(i + m - 1) \leq n$ is the event sequence*

$$S_{i,m} = (s_i, , \ldots, s_{i+m-1})\,.$$

For example, if $S = (a, b, c, b)$, then $S_{3,2} = (c, b)$.

Definition 2.3 (Event sequence database)**.** *A database of event sequences $\mathcal{S}$ is an unordered set of r event sequences:*

$$\mathcal{S} = \{S_1, \ldots, S_r\}\,.$$

For example, $\mathcal{S} = \{(a, b, c, b)\,, (c, a, b)\}$ is a database with two event sequences.

Definition 2.4 (Event count)**.** *Let $\mathbf{1}_A(s_k)$ denote the indicator function that equals 1 if $s_k \in A$ and 0 otherwise. The count of a set of events $A \subseteq L$ in a subsequence $S_{i,m}$ is defined as*

$$\sigma_A(S_{i,m}) = \sum_{k=i}^{i+m-1} \mathbf{1}_A(s_k).$$

The count of a set of events A in an event sequence S of length n is given by

$$\sigma_A(S) = \sigma_A(S_{1,n}),$$

and the count of a set of events A in a database $\mathcal{S} = \{S_1, \ldots, S_r\}$ is

$$\sigma_A(\mathcal{S}) = \sum_{i=1}^{r} \sigma_A(S_i).$$

Definition 2.5 (Event frequency)**.** *The frequency of a set of events $A \subseteq L$ in a subsequence $S_{i,m}$ is defined as*

$$\zeta_A(S_{i,m}) = \frac{\sigma_A(S_{i,m})}{m}.$$

The frequency of a set of events A in an event sequence S of length n is given by $\zeta_A(S) = \zeta_A(S_{1,n})$. Let $|S|$ denote the length of an event sequence S, then the frequency of a set of events A in a database $\mathcal{S} = \{S_1, \ldots, S_r\}$ is

$$\zeta_A(\mathcal{S}) = \frac{\sigma_A(\mathcal{S})}{\sum_{i=1}^{r} |S_i|}.$$

For example, if $\mathcal{S} = \{(a, b, c, b), (c, a, b)\}$, then $\sigma_{\{b\}}(\mathcal{S}) = 2 + 1 = 3$ and $\zeta_{\{b\}}(\mathcal{S}) = \frac{2+1}{4+3} = \frac{3}{7}$. If the set of events A contains only one element, e.g., $A = \{a\}$, we generally write σ_a and ζ_a instead of $\sigma_{\{a\}}$ and $\zeta_{\{a\}}$. Often, the relevant set of events A is clear from the context, in which case A is omitted from the notation, e.g., $\sigma(\mathcal{S}) = \sigma_A(\mathcal{S})$.

2.2 Inter-arrival times

Definition 2.6 (Event occurrence)**.** *Let $S = (s_1, \ldots, s_n)$ be an event sequence, then the event $a \in L$ occurs at position j if and only if*

$$s_j = a.$$

Definition 2.7 (Set of occurrence positions)**.** *Given an event sequence S and an event $a \in L$, denote the event count as $k = \zeta_a(S)$. The set of all occurrence positions of the event a is defined as*

$$\Omega_a = \left\{\omega_a^1, \ldots, \omega_a^k\right\},$$

where ω_a^1 is the first position where a occurs in S, ω_a^2 is the second position, ω_a^3 is the third position, …, and ω_a^k is the last position.

For example, given $S = (a, b, c, b)$, the event b occurs at positions 2 and 4 and $\Omega_b = \{2, 4\}$.

Definition 2.8 (Set of inter-arrival times)**.** *Let Ω_a be a set of occurrence positions, and let k denote the number of occurrence positions $k = |\Omega_a|$, then the corresponding set of inter-arrival times Π_a is given by*

$$\Pi_a = \left\{ \pi_a^1, \ldots, \pi_a^k \right\},$$

where

$$\pi_a^i = \begin{cases} \omega_a^{i+1} - \omega_a^i, & \textit{if } i \leq k-1, \\ n + \omega_a^1 - \omega_a^i, & \textit{if } i = k. \end{cases}$$

For example, given $S = (a, b, c, b)$, the inter-arrival times for event b are

$$\pi_b^1 = \omega_b^2 - \omega_b^1 = 4 - 2 = 2,$$

and

$$\pi_b^2 = n + \omega_b^1 - \omega_b^2 = 4 + 2 - 4 = 2.$$

Thus, the set of inter-arrival times for event b is $\Pi_b = \{2, 2\}$. The previous definitions could be generalised for sets of events, but only sets of inter-arrival times for single events are considered in this thesis.

2.3 Burstiness and dispersion

We use the terms *burstiness* and *dispersion* to refer to measures of the variability of the frequency of an event. An event that is *bursty* or that has *low dispersion* tends to be frequent in some parts of an event sequence and infrequent in all other parts of an event sequence. In this thesis, burstiness refers to a statistic computed from the set of inter-arrival times of an event, while dispersion refers to a statistic computed from the distribution of frequencies across event sequences.

We use the method introduced in Altmann et al. [2009] to measure the burstiness of an event. The measure is based on fitting a *Weibull distribution* [Weibull, 1951] to the set of inter-arrival times. The probability density function of the Weibull distribution is

$$f(x; \alpha, \beta) = \begin{cases} \frac{\beta}{\alpha} \left(\frac{x}{\alpha}\right)^{\beta-1} e^{-(x/\alpha)^\beta} & x \geq 0, \\ 0 & otherwise, \end{cases}$$

The parameters $\alpha > 0$ and $\beta > 0$ are known as the scale and shape parameters, respectively. When $\beta = 1$, the Weibull distribution is equal

to an exponential distribution. As β decreases, the probability of x having a low or a high value increases, and the probability for values around the mean decreases. Functions for maximum likelihood estimation of the parameters are available in most statistical software packages.

Definition 2.9 (Burstiness of an event)**.** *Given a set Π_a of inter-arrival times and the corresponding maximum-likelihood estimate $(\hat{\alpha}, \hat{\beta})$ for the parameters of the Weibull distribution, the burstiness of the event a is defined as the value $\hat{\beta}$.*

In corpus linguistics, it is more common to study dispersion of word frequencies. Dispersion is in interpretation similar to the measure of burstiness defined above, but is quantified using event frequencies over texts. There is no consensus on what is the best quantification of dispersion, a recent survey is Gries [2008]. The measure adopted here is the *normalised index of dispersion*, which was introduced in Gries [2008] and refined in Lijffijt and Gries [2012]. The measure is defined as follows.

Definition 2.10 (Event dispersion)**.** *Given an event $a \in L$ and a database of event sequences $\mathcal{S} = \{S_1, \ldots, S_r\}$ with lengths $n_1, \ldots, n_r$. Let $N = \sum_{i=1}^{r} n_i$ be their sum and let $f_1, \ldots, f_r$ be the relative frequencies: $f_i = \sigma_a(S_i)/\sigma_a(\mathcal{S})$. The dispersion of the event a in database $\mathcal{S}$ is defined as*

$$DP_{norm} = \frac{\sum_{i=1}^{r} \left| f_i - \frac{n_i}{N} \right|}{2 \cdot \left(1 - \frac{\min_i(n_i)}{N}\right)}.$$

For example, let $\mathcal{S}$ be a database with three event sequences, each with length 1,000 events, and assume that $\sigma_a(S_1) = 100$, $\sigma_a(S_2) = 200$, and $\sigma_a(S_3) = 300$. Then, $N = 3{,}000$, $\min_i(n_i) = 1{,}000$, $f_1 = 100/600$, $f_2 = 200/600$, $f_3 = 300/600$, and thus

$$DP_{norm} = \frac{|1/6 - 1/3| + |2/6 - 1/3| + |3/6 - 1/3|}{2 \cdot (1 - 1/3)} = \frac{2/6}{4/3} = 0.25.$$

DP_{norm} takes values in the range $[0, 1]$. It equals 1 when all occurrences of the event are in the shortest event sequence, and 0 if all relative event frequencies are equal to the relative event sequence lengths, regardless of the distribution of the event sequence lengths [Lijffijt and Gries, 2012]. An event has low dispersion if DP_{norm} is high.

2.4 Statistical significance testing

Statistical significance testing, also called *hypothesis testing*, can be used to aid a decision making process involving data whose generative process

is (partly) unknown. Using hypothesis testing in decision making, for example to reject a scientific theory, is a complex topic with many aspects. For a comprehensive view on the topic of hypothesis testing, see, for example, Lehmann and Romano [2005].

We consider the following aspects: in Chapters 4 and 5, novel methods are introduced for assessing the significance of variations in event counts across databases, and within event sequences. In the first case the problem is to specify and appropriate null hypothesis, while in the second case the problem is to compute the probabilities corresponding to the null hypothesis as accurately and efficiently as possible[1]. This section introduces a few basic concepts, while all relevant details are presented in the corresponding chapters.

Definition 2.11 (Test statistic)**.** *The test statistic is the quantity of interest in a data set, for example the frequency of an event in an event sequence.*

Definition 2.12 (Null hypothesis)**.** *Let T denote a random variable that corresponds to the value of the test statistic. The null hypothesis is a probability distribution over the test statistic:* $\Pr(\{T = k\})$.

Definition 2.13 (One-tailed p-value)**.** *Let k denote the value of the test statistic in the data. The one-tailed p-value for the data is the probability that the test statistic is $\geq k$, under the null hypothesis:*

$$p_H = \Pr(\{T \geq k\}).$$

Alternatively, the one-tailed p-value for the data may correspond to the test statistic being smaller than or equal to k:

$$p_L = \Pr(\{T \leq k\}).$$

We indicate the direction with subscript H (high) and L (low).

Definition 2.14 (Two-tailed p-value)**.** *Let k denote the value of the test statistic in the data. The two-tailed p-value is defined as:*

$$p_T = \Pr(\{T = k\}) + 2 \cdot \min(\Pr(\{T > k\}), \Pr(\{T < k\})).$$

Which type of p-value to use, one-tailed or two-tailed, depends on the application and the question that the user aims to answer.

[1]The topic of designing computationally efficient algorithms for computing probabilities (or p-values) as accurately as possible could be argued to be probability theory, or algorithm design, but the aim here is to use these probabilities in hypothesis testing and analysis of data.

Definition 2.15 (Statistically significant)**.** *An observation is statistically significant if and only if its p-value is less than or equal to a prespecified threshold α:*

$$p \leq \alpha.$$

For the sake of readability, the terminology used in this thesis is that an observation is "statistically significant" instead of the traditional statement that "the null hypothesis is rejected".

2.5 Testing multiple hypotheses

Due to the probabilistic nature of statistical significance testing, two types of errors may occur in an inference process that is based on statistical testing. Table 2.1 gives an overview of each of the possible situations.

Definition 2.16 (False positive)**.** *A false positive, also known as type I error, is the event that an observation is declared significant, while the null hypothesis is true. This corresponds to situation FP in Table 2.1.*

Definition 2.17 (False negative)**.** *A false negative, also known as type II error, is the event that an observation is not declared significant, while the null hypothesis is false. This corresponds to situation FN in Table 2.1.*

Table 2.1. When using statistical significance testing for inference, it is possible to make errors. Situations FP and FN correspond to *type I* and *type II errors*, respectively, while situations TN and TP correspond to correct inferences.

	Observation not declared significant	Observation declared significant
Null hypothesis is true	TN	FP
Null hypothesis is false	FN	TP

Several forms of *control* have been proposed in order to make p-values easier to interpret when testing multiple hypotheses concurrently. For an overview see, for example, Shaffer [1995]. Depending on the problem setting, we use methods for control of the *family-wise error rate*, or control of the *false discovery rate*.

Let TN, FP, FN, and TP denote random variables that correspond to the number of times each of the situations occurs in a multiple testing scenario.

Definition 2.18 (Family-wise error rate)**.** *The family-wise error rate is the*

probability that at least one false positive occurs:

$$\Pr\left(\{FP > 0\}\right).$$

Definition 2.19 (False discovery rate)**.** *The false discovery rate is the expected number of false positives over the total number of observations declared significant:*

$$E\left[\frac{FP}{FP+TP}\right].$$

A method for control provides a guarantee on the rate, typically an upper bound, for example the probability that at least one observation is falsely declared significant is at most t: $\Pr\left(\{FP > 0\}\right) \leq t$, or the expected rate of false positives over all positives is at most t: $E\left[FP/(FP+TP)\right] \leq t$. Such methods and guarantees are presented where used.

Note that the word *rate* has a slightly different meaning in family-wise error rate as in false discovery rate. In the first case, it is a *probability* that a certain event (falsely declaring one or more observations as significant) occurs while conducting an experiment and the following inference. Thus, it is a rate over the number of times the entire experiment is repeated. In the second case, it is a ratio (between the number of false positives and the total number of positives).

Also, given a statistical test, the family-wise error rate depends on the total number of hypotheses tested, while the false discovery rate does not. Degenerate cases aside, the probability of falsely declaring at least one observation significant grows with the total number of hypotheses, but the expected ratio between the number of false positives over all positives is constant.

3. Data

To investigate the utility of the methods proposed in this thesis, we have applied them to data sets from several domains. This chapter provides brief introductions to all data sets.

3.1 British National Corpus

The British National Corpus [2007] is the largest annotated text corpus that is currently available in full-text format. The texts have have been annotated with information such as author gender, age, and target audience, and all texts have been classified into genres [Lee, 2001]. The corpus is available for a fee through the BNC website[1]. The corpus is used in Chapter 4 to study vocabulary differences between male and female-authored fictional prose, and to study the impact of burstiness/dispersion of words on the statistical significance of vocabulary differences for various statistical tests. The corpus is also used in Chapter 6, where hapax legomenon ratios throughout texts are compared between several genres.

3.2 Corpus of Early English Correspondence

The Corpus of Early English Correspondence is a corpus of letters dated between 1410 and 1681 that was compiled with the aim to study language change over time and sociolinguistic phenomena. A version with standardised spelling[2] was created recently to facilitate diachronic comparisons. The original corpus is freely available through the Oxford Text Archive[3]. The Standardised-spelling Corpus of Early English Correspon-

[1] http://www.natcorp.ox.ac.uk
[2] http://www.helsinki.fi/varieng/CoRD/corpora/CEEC/standardized.html
[3] http://ota.ahds.ac.uk

dence [2012] is used in Chapter 4 to study language change around the English Civil War (1642–1651) and the differences between statistical tests that may be employed to that end. The part used in the study consists of the 3,055 letters (1.2 million words) dated in the 17th-century.

3.3 Pride and Prejudice

Project Gutenberg[4] offers digitised books that are freely available because their copyright has expired. One of the most popular novels available, "Pride and Prejudice" by Jane Austen, is used in Chapter 5 to study the local frequency variation of words, and in Chapter 6 to analyse the relation between burstiness of words and optimal window lengths for studying frequency variations.

3.4 Reference genomes

In Chapter 6, we analyse and compare the spatial occurrence patterns of nucleotides and dinucleotides in the reference genomes of Homo Sapiens (human) [Venter et al., 2001] and Canis Lupus Familiaris (dog) [Kirkness et al., 2003], and in Chapter 5, we study local variation in GC-content in the Homo Sapiens genome. The reference genomes are freely available through the NCBI data repository[5]. The reference genomes are good examples of large event sequences, as they have lengths of ca. 3,200 and 2,500 million bases.

3.5 Strain sensor time series of the Hollandse Brug

In Chapter 6, we study event sequences and time series with multi-scale structure. An example of such a time series is the Infrawatch data[6] [Knobbe et al., 2010, Vespier et al., 2012]. The time series consists of 24 hours of data (860,953 measurements) from a strain sensor on the "Hollandse Brug", a bridge in the Netherlands. The data contains structure at multiple time scales: individual cars and trucks passing on the bridge, traffic jams, and weather effects. The data is freely available.

[4] http://www.gutenberg.org/
[5] http://www.ncbi.nlm.nih.gov
[6] http://infrawatch.liacs.nl

4. Comparing word frequencies between text corpora

In this chapter, we consider the problem how to compare word frequencies across corpora. We model texts as event sequences and a text corpus as a database of event sequences. The problem is then mapped to the question of *how to compare event frequencies across (databases) of event sequences*. This problem is relevant, for example, when a linguist wants to test a hypothesis such as "word X is more frequent in male than in female speech". This can be accomplished by comparing the frequency of word X between two corpora, one containing transcribed male speech and the other containing transcribed female speech.

We introduce two methods based on resampling and we evaluate and compare these methods, along with several existing methods, with respect to their suitability to this task. We analyse the methods theoretically, and present two case studies where the practical differences are investigated. We show that the choice of how to represent the data, and thus the statistical method, matters, and argue that using only total word counts, which is common practice for comparing corpora, may lead to overestimating the significance of frequency differences. We find that the overestimation is related to the spatial distribution of words and that the overestimation occurs at all frequency levels.

4.1 Introduction

Comparing event frequencies across data is an important task in many applications and scientific disciplines. For example, a linguist may want to test the hypothesis "word X is more frequent in male than in female speech", which can be accomplished by comparing the frequency of word X in two text corpora containing conversations by males and females. The problem considered in this chapter is how to assess the statistical signifi-

cance of observed differences in event frequencies between two databases, which turns out to be surprisingly intricate.

The statistical significance depends on the choice of the null hypothesis, i.e., the probability distribution that is assumed to describe the frequency of a word. The norm is to represent the data by total word counts, implicitly assuming that all words in a corpus are independent samples, and thus that the count of each word follows a binomial distribution. This model is known as the *bag-of-words model*, and has been pervasively used in both data mining [Kleinberg, 2003, Lappas et al., 2009] and linguistics [Leech and Fallon, 1992, Rayson et al., 1997, Oakes and Farrow, 2007] for finding words with significantly elevated occurrences in a text.

We argue that the bag-of-words model should not be used when computing the statistical significance of observed difference between text corpora, because the bag-of-words model does not take into account the structure of a corpus. Typically, a corpus is a collection of texts, and while the texts can be regarded as independent samples, the words inside a text are certainly not (statistically) independent.

An example of the fit of the bag-of-words model to two words in a large corpus (taking into account the varying lengths of texts in the corpus) is given in Figure 4.1. Both *for* and *I* are highly frequent and approximately equally frequent, yet their distribution is very different and neither is modelled well by the predicted distribution, because the bag-of-words model does not take into account the structure of the corpus. In Section 4.5, it is shown that almost none of the words found in this large corpus follows a binomial distribution.

Another example is illustrated in Table 4.1, which contains p-values for the hypothesis that the name *Matilda* is used at an equal frequency by male and female authors in the prose fiction subcorpus of the British National Corpus (see Section 3.1). With more than 500 occurrences (408 for male, 169 for female), the well-known chi-squared and log-likelihood ratio tests could be expected to be reliable. However, the other two tests give very different results. The reason that the methods disagree is that the frequency distribution is highly skewed; there are only five texts where the word occurs, of which one text, by a male author, has 408 occurrences. The bag-of-words model, used in the chi-squared and log-likelihood ratio tests, does not account for the uneven distribution[1].

[1]NB. It is not claimed here that the chi-squared or log-likelihood ratio tests are bad tests, but only that their application in this context is inappropriate.

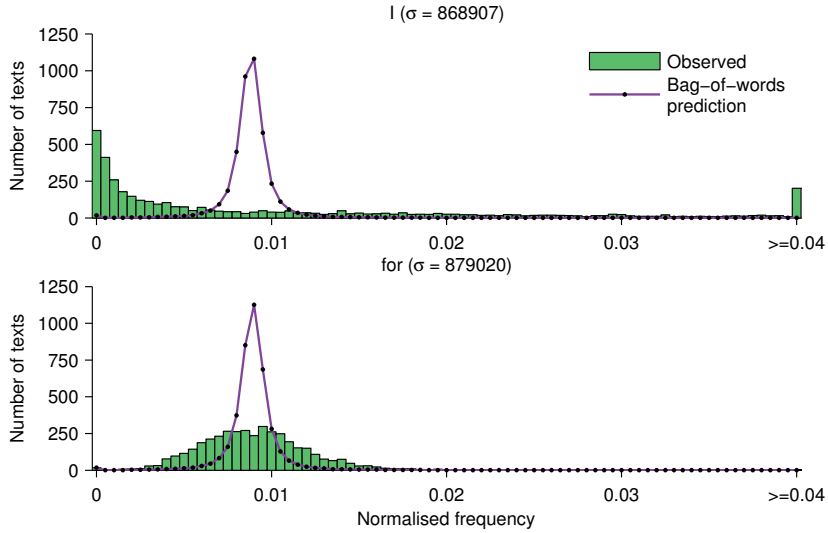

Figure 4.1. Histograms of normalised frequencies of the words *I* and *for* in the British National Corpus, compared to the distribution expected under the bag-of-words model, i.e., when we assume that all words are independent samples and do not take into account the structure of the corpus.

Table 4.1. P-values for the hypothesis that male and female authors use the name *Matilda* at an equal frequency, based on the prose fiction subcorpus of the British National Corpus.

Method	Chi-squared test	Log-likelihood ratio test	Welch's t-test	Wilcoxon rank-sum test
p	< 0.0001	< 0.0001	0.4393	0.1866

In linguistics, frequencies of words and other phenomena, such as n-grams, proverbs, semantic tags, etc., are widely used to study how people communicate. It has been pointed out previously that the bag-of-words model is a poor descriptor of word occurrences. Due to unintuitive outcomes from the chi-squared test, Kilgarriff [2005] claimed that hypothesis testing of word frequencies is rarely useful for finding associations and often leads to misleading results. Rayson and Garside [2000] argued that manual investigation of all differences is required, while Gries [2005b] concluded that each significant result should be checked using an effect size measure. Oakes and Farrow [2007] and Gries [2008] suggest that besides testing the significance, a measure of *dispersion* should be used.

In information retrieval, the fraction of documents where a word occurs is used to detect content-related words. The *inverse document frequency*, used in *tf-idf*, but also in more recent approaches such as *Okapi BM25* [Robertson and Walker, 1994, Baeza-Yates and Ribeiro-Neto, 1999], is useful because content-related words are less dispersed than words with

a grammatical function. Usually the statistical significance of word frequencies is irrelevant, because the task is to rank documents according to their relevance to a given set of query words, and not to find words that describe the documents. The problem setting considered here is different and thus the methods are not directly comparable.

The contextual behaviour of words in texts varies and is affected by several factors, such as topic and genre [Biber, 1995], and author characteristics (gender, age, social class) [Rayson et al., 1997]. For example, in written language, especially in newspaper texts, there is avoidance of repeating a word, due to stylistic ideals, whereas in conversation, priming of words and syntactic structures plays an important role [Gries, 2005a, Szmrecsanyi, 2005]. Hence, it is evident that natural language is non-homogeneous. The variation in frequency depends on the specific word, but as shown later, is almost always much larger than expected under the bag-of-words model.

To model the contextual behaviour of words, we consider their spatial distribution throughout texts. The primary unit used in modelling is the interval between two occurrences of a word. This interval is referred to as the *inter-arrival time* between two instances. Altmann et al. [2009] argued that *inter-arrival times* of word occurrences in natural language can be modelled to a good accuracy using a Weibull distribution. This parametric distribution gives rise to a parameter β that can be interpreted as the *burstiness* of a word. Bursty words tend to exhibit long inter-arrival times followed by short inter-arrival times, while the inter-arrival times for non-bursty words have smaller variance. The lower the burstiness parameter, the burstier the word: for example, $\beta_{for} = 0.93$ and $\beta_I = 0.57$.

Summary of contributions. We introduce two methods to asses the significance of observed differences in word frequencies between two text corpora. The first method is based on randomisation testing and takes into account the word frequency at the text level, while the second method is based on the inter-arrival time distribution of individual words. We compare these methods to existing methods in a series of experiments, and present case studies on two large corpora: the British National Corpus (BNC) and the Corpus of Early English Correspondence (CEEC), see Sections 3.1 and 3.2, respectively. The case studies are based on comparing word frequencies between genders in the BNC and over time in CEEC.

We find that the choice of the statistical test, and more generally, the choice of how the data is represented, matters: for most words, the thresh-

old for a word to be reported as significant increases substantially when taking into account its distribution throughout the corpus. Therefore, fewer words are reported as significant. We also find that Welch's t-test, and the introduced randomisation and inter-arrival time tests give similar results, and conclude that each of these tests is a viable choice for comparing word frequencies across corpora.

Outline. We continue by discussing the related work in Section 4.2. The formal statistical problem is defined in Section 4.3, and the methods are presented in Section 4.4. Results from the experiments and case studies are presented in Section 5.5. Conclusions are given in Section 5.6.

4.2 Related work

Research on graphs and networks has shown that many natural phenomena and patterns in human activity exhibit bursty behaviour [Faloutsos et al., 1999, Barabási, 2005, Kumar et al., 2005, Leskovec et al., 2007]. The discovery of power-law distributions occurred in the study of natural language; Zipf's law [Zipf, 1949], relating the rank of words and their frequencies, describes the oldest known example of a power-law. However, it seems that for comparing word frequencies across text corpora, no heavy-tailed modelling has been attempted.

The bag-of-words model has been pervasively used in data mining and linguistics communities for finding words with significantly elevated occurrences in a text, see, e.g., Leech and Fallon [1992], Rayson et al. [1997], Kleinberg [2003], Oakes and Farrow [2007], and Lappas et al. [2009]. The chi-squared test for independence was introduced by Pearson [1900]. Hofland and Johansson [1982] were the first to apply the chi-squared test to study differences between text corpora. They compared American and British English using the Brown and LOB corpora.

Theoretical works on statistical testing under the bag-of-words assumption include Dunning [1993] and Rayson et al. [2004]. Dunning [1993] proposed a log-likelihood ratio test to find associations between words, because the chi-squared test is based on a normal approximation to the binomial and thus inaccurate for small sample sizes. The log-likelihood ratio test can also be applied to compare word frequencies between two corpora. Rayson et al. [2004] discussed Yates' correction when using the chi-squared test with small samples, but, based on results from simulations, proposed to use Cochran's rule with an extension for the purpose of

comparing word frequencies between text corpora.

Several studies have argued that statistical tests based on comparing vectors of relative word counts per text are more appropriate than statistical tests based exclusively on counts per corpus. Kilgarriff [2001] concluded that the Wilcoxon rank-sum test[2] is preferable to the chi-squared test after studying the most significant differences between the Brown and the LOB corpus, and the most significant differences between males and females in the transcribed conversations in the BNC. Paquot and Bestgen [2009] favour the t-test over the log-likelihood ratio test and the Wilcoxon rank-sum test, based on a study of word frequency differences between the academic prose and fiction prose genres in the BNC.

In data mining, modelling burstiness of words has received attention with respect to several applications. For example, Kleinberg [2003] proposes to model the frequency of a word over time in a hierarchical fashion, using an infinite-state automaton, with the aim of detecting bursts. Lappas et al. [2009] introduced a search framework to identify bursts of words over time in a corpus of time-stamped news articles. The method is based on computing a score under the bag-of-words model, but values are then processed further to identify bursts. Madsen et al. [2005] and Elkan [2006] use the DCM distribution, a multinomial distribution with one additional parameter to take into account burstiness, for classification and clustering of texts, respectively. Blei and Frazier [2011] proposed the distance dependent CRP for taking account word burstiness and the structure of texts while learning topic models. However, none of these methods are directly applicable to the problem studied in this chapter.

The fact that the bag-of-words model poorly describes word frequencies in text corpora is not surprising, as it is well known that words do not occur at random [Church and Gale, 1995, Katz, 1996, Kilgarriff, 2005, Evert, 2006]. The materialisation of linguistic utterances depends on many factors, such as text genre [Biber, 1995], and target audience [Bell, 1984]. Besides these, there are other immediate cognitive features that play a role, such as word priming [Gries, 2005a, Szmrecsanyi, 2005].

[2]The statistical test that we refer to as the Wilcoxon rank-sum test has several names in the literature. For example, Kilgarriff [2001] uses the name Mann-Whitney U-test, while Paquot and Bestgen [2009] use the name Wilcoxon Mann-Whitney test. The name used here is the same as in Matlab and R.

4.3 Problem statement

Relevant preliminaries, definitions regarding event sequences and statistical significance testing, are presented in Sections 2.1 and 2.4. Our aim is to compare the frequency of a word between two corpora and to assess the significance of an observed difference.

Each text in a corpus is represented as an event sequence and the event labels are the word types. All punctuation is ignored and, in the experiments, words are lowercased, such that there is no difference between a word that occurs at the start of a sentence. For example, the sentence "The aim is to compare the frequency." is mapped to the event sequence $(the, aim, is, to, compare, the, frequency)$. Lowercasing is not required, and, in contrast, it would also be possible to enrich the words before mapping them to event labels, for example using part-of-speech or semantic tagging, such that each event correspond to a word+tag, or a unique semantic identifier.

Let $\mathcal{S}$ and $\mathcal{T}$ be two text corpora, i.e., databases of event sequences. We assume that the corpora are representative samples of the language varieties that should be compared. The research question is: is the frequency of a word w different in the two language varieties? The true mean frequencies of the word w in the language varieties, denoted as $\theta_{w,\mathcal{S}}$ and $\theta_{w,\mathcal{T}}$, are unobserved. However, the observed mean frequency is the maximum-likelihood estimate for the true mean frequency: $\hat{\theta}_{w,\mathcal{S}} = \zeta_w(\mathcal{S})$.

Under the bag-of-words assumption, the probabilistic models for the counts of w in $\mathcal{S}$ and $\mathcal{T}$ are fully specified by $\theta_{w,\mathcal{S}}$ and $\theta_{w,\mathcal{T}}$. Under that assumption, several statistical tests exist that assess the statistical significance of the observed difference: Pearson's chi-squared test, the log-likelihood ratio test (Dunning's G^2), Fisher's exact test, or the binomial test. For large sample sizes, these tests are similar, but for small sample sizes there are apparent differences, see, e.g., Dunning [1993].

We consider two alternative approaches: (1) counting the frequency of w per text/event sequence and comparing the frequency distributions of the two corpora, and (2) counting the inter-arrival times of w in the two corpora and comparing the inter-arrival time distributions of the two corpora. We study both parametric and non-parametric tests for both representations, and investigate the differences between the methods using simulations and in two case studies.

4.4 Methods

We discuss six methods in this section, the first four have been applied or reviewed in several studies in corpus linguistics. These are: the chi-squared, log-likelihood ratio, Wilcoxon rank-sum, and t-test. The other two methods are both based on resampling, but use different sources of information, either vectors of per-text frequencies, or vectors of inter-arrival times. Both of these methods come in two flavours. As is common in corpus linguistics, all methods provide two-tailed p-values.

4.4.1 Pearson's chi-squared test

Pearson's chi-squared test [Pearson, 1900], also known as the chi-squared test for independence, or simply as the chi-squared test, can be used to test if two (categorical) variables are statistically independent of each other. The application of the test to the problem considered in this chapter is based on the assumption that the whole corpus can be modelled as a sequence of independent Bernoulli trials, i.e., represented in the bag-of-words model.

The test is conducted as follows. Let w be the word of interest. The two variables that are tested for having a significant association are the word w and the corpus. Both variables are binary: given the set of all words in both corpora, every word is either the word w, or not, and every word is either in corpus $\mathcal{S}$, or corpus $\mathcal{T}$.

Let $n_{\mathcal{S}}$ and $n_{\mathcal{T}}$ be the total number of words in corpora $\mathcal{S}$ and $\mathcal{T}$, and define $\zeta_w(\mathcal{S} \cup \mathcal{T}) = \frac{\sigma_w(\mathcal{S})+\sigma_w(\mathcal{T})}{n_{\mathcal{S}}+n_{\mathcal{T}}}$, i.e., $\zeta_w(\mathcal{S} \cup \mathcal{T})$ is the average frequency of word w over the two corpora. The expected count of w in corpus $\mathcal{S}$, under the assumption of independence, is $\hat{\sigma}_w(\mathcal{S}) = n_{\mathcal{S}} \cdot \zeta_w(\mathcal{S} \cup \mathcal{T})$, and likewise $\hat{\sigma}_w(\mathcal{T}) = n_{\mathcal{T}} \cdot \zeta_w(\mathcal{S} \cup \mathcal{T})$. Using these definitions, the test statistic X^2 for the chi-squared test with Yates' correction [Yates, 1934], is

$$X^2 = \frac{(|\sigma_w(\mathcal{S}) - \hat{\sigma}_w(\mathcal{S})| - 0.5)^2}{\hat{\sigma}_w(\mathcal{S})} + \frac{(|\sigma_w(\mathcal{T}) - \hat{\sigma}_w(\mathcal{T})| - 0.5)^2}{\hat{\sigma}_w(\mathcal{T})} + \frac{(|\hat{\sigma}_w(\mathcal{S}) - \sigma_w(\mathcal{S})| - 0.5)^2}{n_{\mathcal{S}} - \hat{\sigma}_w(\mathcal{S})} + \frac{(|\hat{\sigma}_w(\mathcal{T}) - \sigma_w(\mathcal{T})| - 0.5)^2}{n_{\mathcal{T}} - \hat{\sigma}_w(\mathcal{T})}.$$

The test statistic asymptotically follows a chi-squared distribution with one degree of freedom. The p-value can be obtained by comparing the test statistic to a table of chi-squared distributions.

Occasionally, studies have used the goodness-of-fit test, rather than the test for independence, which corresponds to omitting the last two terms in the test statistic. One such study is Rayson et al. [1997]. For most words,

the expected event counts $\hat{\sigma}_w(\mathcal{S})$ and $\hat{\sigma}_w(\mathcal{T})$ are small compared to $n_\mathcal{S}$ and $n_\mathcal{T}$, in which case the difference between the tests is also small.

4.4.2 Log-likelihood ratio test

Pearson's chi-squared test is based on two approximations: a normal distribution approximates the binomial distribution, and the test statistic follows a chi-squared distribution only asymptotically. Due to the double approximation, the chi-squared test is inaccurate when the word frequency is small. Positive bias caused by the second approximation can be avoided by applying Yates' correction [Yates, 1934], but negative bias may then be a result. For this reason, Dunning [1993] introduced a new test, based on a likelihood ratio. This test is called the log-likelihood ratio test and is also known as the G^2 test.

The test is conducted as follows. The log-likelihood ratio tests is based on comparing the likelihood of the data, under the bag-of-words assumption, using separate estimates for the frequency of a word w with the likelihood of the data using a single estimate. Let $Bin\left(k; n, p\right) = \binom{n}{k} p^k (1-p)^{n-k}$ denote the probability mass function of the binomial distribution. Using the definitions from Section 4.4.1, the likelihood ratio λ is

$$\lambda = \frac{Bin\left(\sigma_w(\mathcal{S}); n_\mathcal{S}, \zeta_w(\mathcal{S} \cup \mathcal{T})\right) \cdot Bin\left(\sigma_w(\mathcal{T}); n_\mathcal{T}, \zeta_w(\mathcal{S} \cup \mathcal{T})\right)}{Bin\left(\sigma_w(\mathcal{S}); n_\mathcal{S}, \zeta_w(\mathcal{S})\right) \cdot Bin\left(\sigma_w(\mathcal{T}); n_\mathcal{T}, \zeta_w(\mathcal{T})\right)}.$$

A full derivation can be found in Dunning [1993].

The test statistic is $-2 \log \lambda$, which asymptotically follows a chi-squared distribution with one degree of freedom. Dunning [1993] claims that this test statistic approaches its asymptotical distribution much faster than the test statistic in the chi-squared test and is thus preferable, especially when the expected frequency is low. The p-value is computed by comparing the test statistic $-2 \log \lambda$ to a table of chi-squared distributions.

There exist other bag-of-words tests that are not based on approximations, but are directly based on the summation of values under a binomial or hypergeometric distribution; these are the binomial test and Fisher's exact test. These provide more accurate probabilities, especially for small frequencies, under the bag-of-words model. These tests are not studied further here because their results are expected to be similar to the chi-squared and log-likelihood ratio tests, and these tests have not been used by linguists.

4.4.3 Welch's t-test

Welch's t-test is a statistical test based on the assumption that the quantity of interest, in this case the mean frequency of a word w, follows a normal distribution. Welch's t-test is more generally applicable than Student's t-test because it does not assume equal variance in the two populations. Welch's t-test provides a p-value for the hypothesis that the means of the two distributions are equal.

The test is conducted as follows. Let $\zeta_w(S_1), \ldots, \zeta_w(S_{|\mathcal{S}|})$ denote the frequency of word w in each of the texts in corpus $\mathcal{S}$, and likewise define $\zeta_w(T_1), \ldots, \zeta_w(T_{|\mathcal{T}|})$ for corpus $\mathcal{T}$. The sample means are given by $\overline{\zeta}_w(\mathcal{S}) = \frac{1}{|\mathcal{S}|}\sum_{i=1}^{|\mathcal{S}|}\zeta_w(S_i)$ and $\overline{\zeta}_w(\mathcal{T}) = \frac{1}{|\mathcal{T}|}\sum_{i=1}^{|\mathcal{T}|}\zeta_w(T_i)$, while the variances equal $s^2(\mathcal{S}) = \frac{1}{|\mathcal{S}|}\sum_{i=1}^{|\mathcal{S}|}\left(\zeta_w(S_i) - \overline{\zeta}_w(\mathcal{S})\right)^2$ and $s^2(\mathcal{T}) = \frac{1}{|\mathcal{T}|}\sum_{i=1}^{|\mathcal{T}|}\left(\zeta_w(T_i) - \overline{\zeta}_w(\mathcal{T})\right)^2$. The test statistic t is given by

$$t = \frac{\overline{\zeta}_w(\mathcal{S}) - \overline{\zeta}_w(\mathcal{T})}{\sqrt{\frac{s^2(\mathcal{S})}{|\mathcal{S}|} + \frac{s^2(\mathcal{T})}{|\mathcal{T}|}}}.$$

The test statistic follows a t-distribution with ν degrees of freedom, where ν depends on the variance of the populations. Although it is not known how to compute ν exactly, Welch [1947] introduced an approximate solution. Implementations of this test are available in statistical software programs, including R and Matlab.

The null hypothesis for the t-test is that the mean frequencies $\overline{\zeta}_w(\mathcal{S})$ and $\overline{\zeta}_w(\mathcal{T})$ are equal, which are not necessarily the same frequencies as those that are compared in the bag-of-words tests, which are $\zeta_w(\mathcal{S})$ and $\zeta_w(\mathcal{T})$. The estimates coincide, for example, when all texts are equally long, and they are known as the average-of-average $\overline{\zeta}_w(\mathcal{S})$ and pooled $\zeta_w(\mathcal{S})$ frequency [Hinneburg et al., 2007].

4.4.4 Wilcoxon rank-sum test

The Wilcoxon rank-sum test, also known as the Mann-Whitney U-test, can be used to test if two distributions are equal. It is based on the fact that if two sets of observations are generated from the same distributions, then it is possible to induce a probability distribution over the rank orders [Wilcoxon, 1945, Mann and Whitney, 1947]. No assumptions are made on the shape of the distribution.

The test is conducted as follows. As before, let $\zeta_w(S_1), \ldots, \zeta_w(S_{|\mathcal{S}|})$ and $\zeta_w(T_1), \ldots, \zeta_w(T_{|\mathcal{T}|})$ denote the per-text frequencies. A combined ranking of all frequencies is produced, and for each frequency it is marked from

which corpus it came. This produces a ranked series, such as is illustrated in Table 4.2.

Table 4.2. Example of a ranked series.

Rank	1	2	3	4	5	6	7	8	9	10
Corpus	$\mathcal{S}$	$\mathcal{T}$	$\mathcal{T}$	$\mathcal{T}$	$\mathcal{S}$	$\mathcal{S}$	$\mathcal{S}$	$\mathcal{T}$	$\mathcal{T}$	$\mathcal{S}$

The test statistic U is the sum of the ranks of texts of the smaller corpus. In the example in Table 4.2, both corpora contain 5 texts, thus U can be based on either $\mathcal{S}$ or $\mathcal{T}$. For example, $U_{\mathcal{S}} = 1 + 5 + 6 + 7 + 10 = 29$. When there are only few samples in the smaller corpus, the p-value is obtained by comparing the test statistic with a statistical table, and when the number of samples is greater than 20, the distribution of the test statistic is well approximated by a normal distribution. Many statistical software programs contain an implementation of this test.

Particularly for infrequent words, numerous texts in a corpus may have a frequency of zero. In the case of any ties, each text is assigned the average rank over all equal-frequency texts. For example, if there are five texts with a frequency of zero, these texts would have ranks 1 to 5, thus each text is assigned a rank of $(1 + 2 + 3 + 4 + 5)/5 = 3$.

Rayson [2003], Rayson et al. [2004], and Baron et al. [2009] report that 92% of the types in their study have to be omitted because the Wilcoxon rank-sum test cannot handle ties. They base their statement on Kilgarriff [2001], but this conclusion seems to be based on a misunderstanding, as the test can handle ties. At worst, the test has limited power in case of many ties.

The Wilcoxon rank-sum test is often considered as the alternative to the t-test in case some of the criteria of the t-test are not met, see, e.g., Ruxton [2006]. However, the two methods are not truly comparable, as they test different hypotheses. The t-test compares the two means, while the Wilcoxon rank-sum test compares the full distributions.

4.4.5 Randomisation (permutation and bootstrap) tests

A randomisation test is a statistical test based on resampling of data. There are various methods for data randomisation, each with its own null hypothesis. The aim here is to test the significance of the difference between the observed mean frequencies for a given word w.

The tests are conducted as follows. As before, let $\zeta_w(S_1), \ldots, \zeta_w(S_{|\mathcal{S}|})$ and $\zeta_w(T_1), \ldots, \zeta_w(T_{|\mathcal{T}|})$ denote the per-text frequencies, and let their means be

$\overline{\zeta}_w(\mathcal{S}) = \frac{1}{|\mathcal{S}|}\sum_{i=1}^{|\mathcal{S}|}\zeta_w(S_i)$ and $\overline{\zeta}_w(\mathcal{T}) = \frac{1}{|\mathcal{T}|}\sum_{i=1}^{|\mathcal{T}|}\zeta_w(T_i)$. The test statistic d is the difference between the means:

$$d = \left|\overline{\zeta}_w(\mathcal{S}) - \overline{\zeta}_w(\mathcal{T})\right|.$$

The essential difference between randomisation tests and other statistical tests is that the null hypothesis has no explicit form. Instead, the probability distribution for the test statistic under the null hypothesis is defined by the randomisation method and the given data. Two types of randomisation are considered here: permutations and bootstraps.

Randomisation by permutation implies that $|\mathcal{S}|$ texts are drawn randomly without replacement from the set of all texts to form a corpus $\mathcal{S}'$, and the rest is assigned to a corpus $\mathcal{T}'$. Randomisation by bootstrap implies that $|\mathcal{S}|$ texts are drawn randomly with replacement from the set of all texts to form a corpus $\mathcal{S}'$, and likewise $|\mathcal{T}|$ texts to form a corpus $\mathcal{T}'$. Depending on the type of randomisation used, the method is referred to as the permutation test or the bootstrap test.

The p-value is defined by comparing the test statistic on the observed data (d) with the test statistic in the randomised data:

$$p = \Pr\left(\left\{\left|\overline{\zeta}_w(\mathcal{S}') - \overline{\zeta}_w(\mathcal{T}')\right| \geq d\right\}\right). \tag{4.1}$$

For small sample sizes it may be possible to enumerate all possible assignments of samples to $\mathcal{S}'$ and $\mathcal{T}'$ and solve the p-value exactly. Typically, it is impractical to compute these p-values exactly, in which case they can be estimated using sampling, as follows. Let d denote the test statistic on the observed data and $d'_1, \ldots, d'_N$ the test statistics in N randomisations. Let $\delta(x)$ denote a step function that equals 1 if $x \geq 0$ and 0 otherwise. The empirical p-value is given by [North et al., 2002]

$$\hat{p} = \frac{1 + \sum_{i=1}^{N}\delta(d'_i - d)}{1 + N}. \tag{4.2}$$

It is expected that the differences between the permutation and bootstrap test are small in practice, and the p-values under the two sampling schemes converge as the total number of texts $|\mathcal{S}| + |\mathcal{T}|$ grows. Moreover, the null hypothesis of the permutation test has a clear interpretation; the permutation just breaks the relationship between the grouping in two corpora and the observed frequencies. The bootstrap test is considered here as well, because it has a computational advantage.

For sufficiently large samples, random sampling can be avoided under the bootstrap formulation as follows. By the central limit theorem, as

$|\mathcal{S}'| \to \infty$, $\overline{\zeta}_w(\mathcal{S}')$ approaches a normal distribution with mean

$$\mathrm{E}\left[\overline{\zeta}_w(\mathcal{S}')\right] = \mathrm{E}\left[\frac{1}{|\mathcal{S}'|}\sum_{i=1}^{|\mathcal{S}'|}\zeta_w(S'_i)\right] = \mathrm{E}\left[\zeta_w(S'_i)\right],$$

and variance

$$\mathrm{Var}\left[\overline{\zeta}_w(\mathcal{S}')\right] = \mathrm{Var}\left[\frac{1}{|\mathcal{S}'|}\sum_{i=1}^{|\mathcal{S}'|}\zeta_w(S'_i)\right] = \frac{1}{|\mathcal{S}'|^2}\sum_{i=1}^{|\mathcal{S}'|}\mathrm{Var}\left[\zeta_w(S'_i)\right] = \frac{\mathrm{Var}\left[\zeta_w(S'_i)\right]}{|\mathcal{S}'|}.$$

Similarly, $\mathrm{E}\left[\overline{\zeta}_w(\mathcal{T}')\right] = \mathrm{E}\left[\zeta_w(T'_i)\right]$ and $\mathrm{Var}\left[\overline{\zeta}_w(\mathcal{T}')\right] = \frac{\mathrm{Var}\left[\zeta_w(T'_i)\right]}{|\mathcal{T}'|}$.

Using these equations and the fact that $\overline{\zeta}_w(\mathcal{S}')$ and $\overline{\zeta}_w(\mathcal{T}')$ are independent, the expected difference between the means is

$$\mathrm{E}\left[\overline{\zeta}_w(\mathcal{S}') - \overline{\zeta}_w(\mathcal{T}')\right] = \mathrm{E}\left[\overline{\zeta}_w(\mathcal{S}')\right] - \mathrm{E}\left[\overline{\zeta}_w(\mathcal{T}')\right] = \mathrm{E}\left[\zeta_w(S'_i)\right] - \mathrm{E}\left[\zeta_w(T'_i)\right] = 0,$$

and the variance is

$$\begin{aligned}\mathrm{Var}\left[\overline{\zeta}_w(\mathcal{S}') - \overline{\zeta}_w(\mathcal{T}')\right] &= \mathrm{Var}\left[\overline{\zeta}_w(\mathcal{S}')\right] + \mathrm{Var}\left[\overline{\zeta}_w(\mathcal{T}')\right] \\ &= \frac{\mathrm{Var}\left[\zeta_w(S'_i)\right]}{|\mathcal{S}'|} + \frac{\mathrm{Var}\left[\zeta_w(T'_i)\right]}{|\mathcal{T}'|}.\end{aligned}$$

The p-value is based on the absolute value of the difference (Equation 4.1), which can be taken into account as follows. The difference between the means follows a normal distribution with zero expectation. As the normal distribution is symmetric around the mean, the probability that the absolute difference is larger than some value is exactly twice the probability that the directed difference is larger than that value:

$$\begin{aligned}p &= \Pr\left(\left\{\left|\overline{\zeta}_w(\mathcal{S}') - \overline{\zeta}_w(\mathcal{T}')\right| \geq d\right\}\right) \\ &= \Pr\left(\left\{\overline{\zeta}_w(\mathcal{S}') - \overline{\zeta}_w(\mathcal{T}') \geq d\right\}\right) + \Pr\left(\left\{\overline{\zeta}_w(\mathcal{T}') - \overline{\zeta}_w(\mathcal{S}') \geq d\right\}\right) \\ &= 2 \cdot \Pr\left(\left\{\overline{\zeta}_w(\mathcal{S}') - \overline{\zeta}_w(\mathcal{T}') \geq d\right\}\right).\end{aligned}$$

As $\overline{\zeta}_w(\mathcal{S}') - \overline{\zeta}_w(\mathcal{T}')$ follows a normal distribution with known parameters, the p-value can be readily computed using the cumulative distribution function for the normal distribution, which concludes the test.

4.4.6 Inter-arrival time test

Arguably, viewing each text as a single sample, regardless of the number of occurrences of a word, is overly conservative. It is possible to use each occurrence as a sample while accounting for the burstiness of a word, by using the spatial distribution of the word. One way to model the spatial distribution is by counting the *inter-arrival times* of a word. The concepts of inter-arrival times and burstiness are presented in Sections 2.2 and 2.3.

The test is conducted as follows. The test is similar to the permutation test (Section 4.4.5), but uses a different representation of the data. The test statistic is $d = |\sigma_w(\mathcal{S}) - \sigma_w(\mathcal{T})|$. For the randomisation, all texts from $\mathcal{S}$ and $\mathcal{T}$ are ordered randomly and concatenated into a single long event sequence. The set of inter-arrival times Π_w for the given word w is then computed as in Definitions 2.7 and 2.8.

The data is randomised by randomly permuting the inter-arrival times. Let Π'_w denote the randomised set of inter-arrival times, and $n_\mathcal{S}$ the total number of words in corpus $\mathcal{S}$. The set Π'_w induces a set of occurrence positions in an event sequence. Let $k_\mathcal{S}$ denote the number of occurrences of w in the first $n_\mathcal{S}$ positions and $k_\mathcal{T}$ the number of occurrences of w in the rest of the induced event sequence, the test statistic for the randomisation is $d' = |k_\mathcal{S} - k_\mathcal{T}|$.

The p-value is again defined by Equation 4.1, but as the p-value can only be computed exactly for small samples, it is instead estimated using N randomisations and Equation 4.2. This test is referred to as the empirical inter-arrival time test.

Altmann et al. [2009] proposed to model the inter-arrival times with a Weibull distribution. To assess the accuracy of this model, we use that model as the basis for a statistical test. In general the setting is the same as the empirical inter-arrival time test, but the randomisation is performed by sampling inter-arrival times from a Weibull distribution with the maximum likelihood parameters.

Let $f(x)$ denote the probability density function for the Weibull distribution (Equation 2.3). The first inter-arrival time is a special case; at the beginning of the event sequence, it is possible to be at any point in any inter-arrival time. However, it is more likely we are at some point in a long inter-arrival than in a short one, proportional to the length of the inter-arrival. Thus, the first inter-arrival time should be sampled uniformly from $g(x) = C \cdot x \cdot f(x)$, where C is a normalisation constant, such that $\int_0^\infty g(x)\, dx = 1$.

Within an inter-arrival time, any position is equally likely. After the first inter-arrival time, inter-arrival times are sampled from $f(x)$ until the sum of the inter-arrival times is greater than $n_\mathcal{S} + n_\mathcal{T}$. The test statistic and p-value are then computed as before. This method is referred to as the Weibull inter-arrival time test.

Table 4.3. Summary of the assumptions underlying the six methods discussed in this chapter.

Test	Assumptions on frequency distribution
Chi-squared test	All words in all texts are i.i.d. samples (bag-of-words model)
Log-likelihood ratio test	All words in all texts are i.i.d. samples (bag-of-words model)
Welch's t-test	All texts are i.i.d. samples, and mean frequencies follow a normal distribution
Wilcoxon rank-sum test	All texts are i.i.d. samples
Randomisation tests	All texts are i.i.d. samples, and for bootstrap: mean frequencies follow a normal distribution
Inter-arrival time tests	Number of words between occurrences of the same word are i.i.d. samples

4.4.7 Summary of methods

A summary of the assumptions on the frequency distribution for each of the six methods is given in Table 4.3. The Wilcoxon rank-sum and permutation tests make the weakest assumptions and are thus the most generally applicable. However, as stated previously, the Wilcoxon rank-sum test compares the ranks between two distributions and not the means, and is thus not directly comparable to the other tests.

4.5 Experiments

To compare the tests and evaluate the consequences of choosing a particular test, we designed three experiments. In the first experiment, we compare the p-values from all methods on randomised data for various null hypotheses with the theoretically optimal distributions. The results of this experiment are presented in Section 4.5.1.

The second and third experiments are case studies: a comparison of fictional prose by male and female authors, and a comparison of personal letters before and after the English Civil War. In both case studies, we have applied multiple methods to find all words that have significantly different frequencies in order to study how the tests differ in a practical situation. The case studies are presented in Sections 4.5.2 and 4.5.3.

4.5.1 Uniformity of p-values under various null hypotheses

The first experiment was designed to investigate the relation between the performance of the methods and the assumptions on the data that a user may be willing to make. The applicability of a test can be evaluated based on the criterion that, if the data follow the distribution that is assumed in the null hypothesis, then the p-values should have a uniform distribution in the $[0, 1]$ range.

This criterion is applicable by definition of the p-value: the probability of encountering a p-value of x or less is x itself. If this criterion is not fulfilled, then the test is either anti-conservative, i.e., the probability of encountering a p-value of x or smaller is more than x, or conservative, i.e., the probability of encountering a p-value of x or smaller is less than x. We use the Kolmogorov-Smirnov test [Massey, 1951] to test the uniformity of a set of p-values.

We used the same data as in the first case study: all texts from the British National Corpus classified as fiction prose for which the gender of the author is known, see Section 3.1 for a general introduction. The subcorpus contains 200 texts written by men, and 205 texts written by women of length $\geq$ 2,000. To make the data more homogeneous, we included only the first 2,000 words of each text. We preprocessed the texts by lowercasing all words and ignored punctuation and other markup.

We considered three assumptions, which correspond to the assumptions underlying the various tests, and we evaluated all methods under each assumption. The assumptions are: (A) the texts are independent samples, (B) the inter-arrival times are independent samples, and (C) all words are independent samples. We generated randomised data as follows. First, we discarded 5 texts at random to make a corpus of exactly 400 texts. Then, for a given word, depending on the assumption, we computed the frequency per text (A), the inter-arrival time distribution (B), or the total word counts (C). Then, we we repeatedly divided the corresponding samples into two sets $\mathcal{S}'$ and $\mathcal{T}'$.

We repeated the randomisation process 500 times, and each of the methods was applied on each randomisation. This was done for all the 3,269 words that have a count of 20 or higher. For each word, for each method, we tested the 500 p-values for uniformity. This resulted in $8 \cdot 3{,}269 = 26{,}152$ uniformity p-values per assumption. We applied Bonferroni correction in each setting to correct for multiple hypotheses and declared all

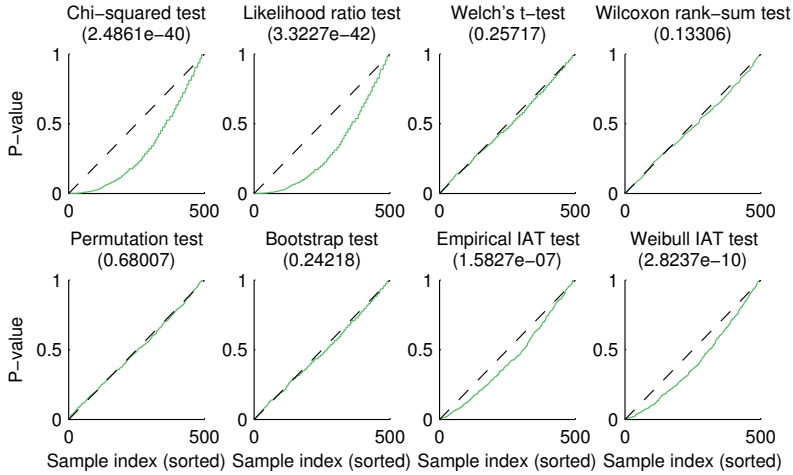

Figure 4.2. P-values over a set of data randomisations for the word *would* for all eight methods under the assumption that all texts are i.i.d. samples. The dashed lines correspond to the optimal uniform distribution and between parentheses are the p-values from the Kolmogorov-Smirnov test whether the p-values differ significantly from the optimal uniform distribution.

p-values ≤ 0.01 as significant.

An example of the p-values for the word *would* under the assumption that texts are i.i.d. samples (assumption A) is given in Figure 4.2. The word *would* occurs 2,590 times in the subcorpus. We find that the distribution of the p-values over the randomisations is uniform for the t-test, Wilcoxon rank-sum test, permutation test, and bootstrap test, while the p-values from inter-arrival time tests are too low, and the p-values of the chi-squared and log-likelihood ratio test even more biased.

The full result for assumption A, for words with count ≥ 100, is presented in Figure 4.3. For the chi-squared and log-likelihood ratio test, the p-values are significantly non-uniform in most cases. The tests that use the representation corresponding to the independence assumption (t-test, Wilcoxon rank-sum test, permutation test, and bootstrap test) clearly outperform the other methods.

To learn whether the p-values in the rejected samples are conservative or anti-conservative, we applied a one-tailed Kolmogorov-Smirnov test in the anti-conservative direction as well. Figure 4.4 shows the result for assumption A, for words with count ≥ 100. By comparing Figures 4.3 and 4.4, we find that the permutation and inter-arrival time tests have a tendency to be conservative, while the chi-squared and log-likelihood ratio tests have a strong anti-conservative bias under assumption A.

We also observe a clear inter-action between the uniformity of p-values

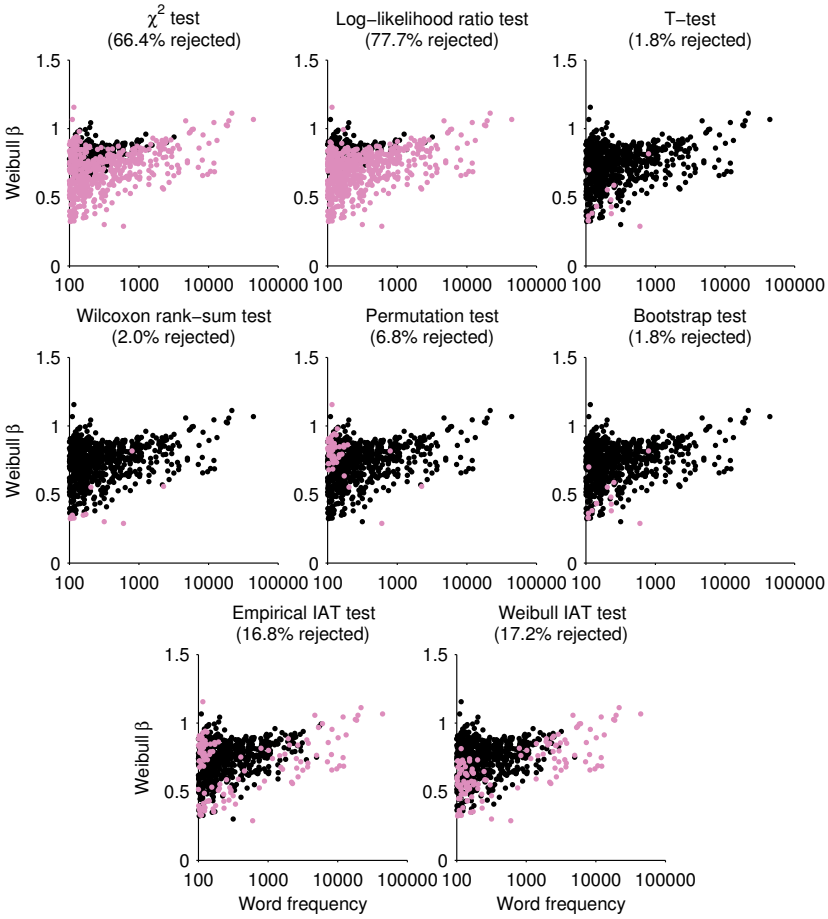

Figure 4.3. Results of the uniformity test for all eight methods under the assumption that all texts are i.i.d. samples. Each dot corresponds to a word, which has a frequency (x-axis) and burstiness (y-axis). Pink dots correspond to rejected samples, that is, the adjusted p-value for the uniformity test is ≤ 0.01.

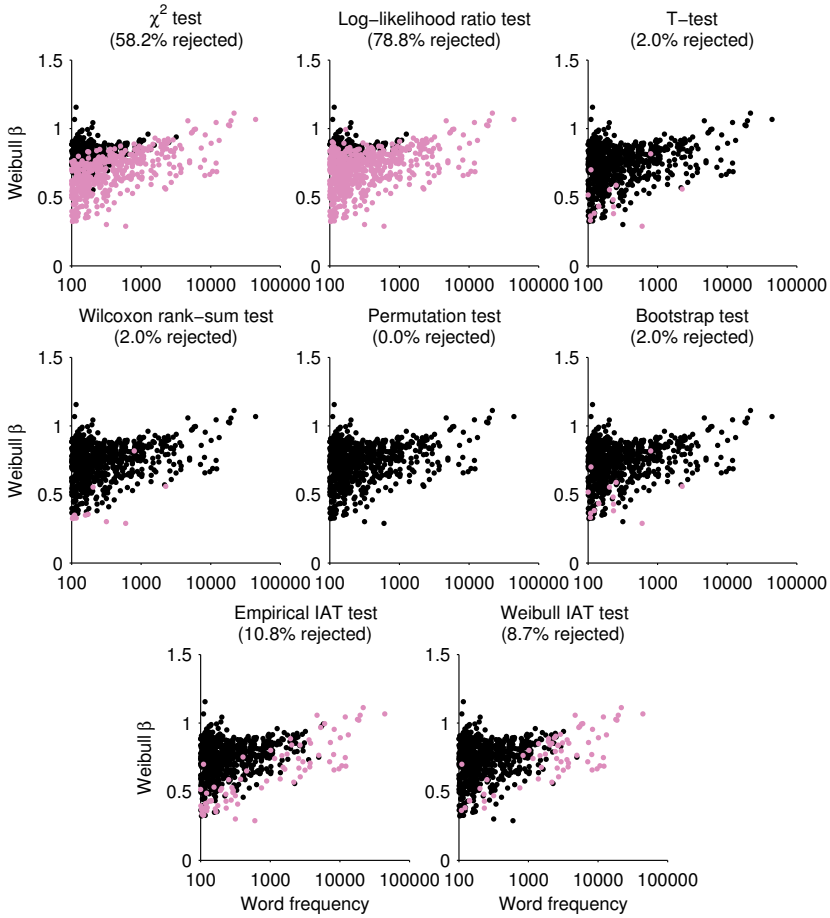

Figure 4.4. Results of the uniformity test in the anti-conservative direction for all eight methods under the assumption that all texts are i.i.d. samples. Each dot corresponds to a word, which has a frequency (x-axis) and burstiness (y-axis). Pink dots correspond to rejected samples, that is, the adjusted p-value for the uniformity test in the anti-conservative direction is ≤ 0.01.

and the burstiness of words: the burstier the word, the more likely the p-values are anti-conservative, and vice versa, the less bursty the word, the more likely the p-values are conservative. This effect is most clear in the results of the empirical inter-arrival time test, where the p-values for a cluster of words in the top-left corner were non-uniform in the conservative direction, and the p-values for the words that are most bursty were non-uniform in the anti-conservative direction. We also observe that there are bursty and non-bursty words at all frequency levels, thus over and underestimation of the significance may occur at any frequency.

Results for all three assumptions for both the two- and one-tailed test and for counts ≥ 20 as well as counts ≥ 100 are listed in Table 4.4. We observe several patterns. When taking into account all words with counts ≥ 20, essentially all methods fail to produce uniform p-values for many words under all assumptions, with one exception: the Wilcoxon rank-sum test under assumption C, that all words are independent samples. That all tests fail shows that a count of 20 is too low for this experiment to succeed. The likely reason is that, for low word counts, there are only few distinct outputs that the methods can give, rendering the p-value distribution non-uniform, regardless of the test.

Considering only the uniformity of p-values for words with counts ≥ 100 provides clearer information: the chi-squared, log-likelihood ratio and Weibull IAT test are appropriate only when assuming that all words are independent (assumption C), while the chi-squared test is noticeably conservative even in that case (41.9% vs. 0.3% rejected on the two- vs. one-tailed test), probably due to Yates' correction. That the Weibull IAT test does not perform as expected under the assumption of independent inter-arrival times suggests that the Weibull distribution is not a good model for describing the inter-arrival times.

Welch's t-test, the Wilcoxon rank-sum test and the bootstrap test produce reasonably uniform p-values under all three assumptions ($\leq$ 4.2 % rejected). The empirical IAT test appears to be appropriate only under the assumption of independent inter-arrival times, otherwise being conservative, anti-conservative or both. The permutation test, which is also based on the empirical p-value, is also conservative. The $+1$ correction in the empirical p-value (Equation 4.2) is the probable cause, using more randomisations (here 200) may reduce this problem.

Based on this experiment, it appears that under assumption A, Welch's t-test, the Wilcoxon rank-sum test and the bootstrap test are the prefer-

Table 4.4. Percentage of randomisations where the p-values are significantly non-uniform (two-tailed Kolmogorov-Smirnov test), for all tests, for all three assumptions (A, B, C), and for two frequency thresholds. Results for the one-tailed test in the anti-conservative direction are given between parentheses.

Test	A, count $\geq$ 100	A, count $\geq$ 20	B, count $\geq$ 100	B, count $\geq$ 20	C, count $\geq$ 100	C, count $\geq$ 20
Chi-squared	66.4 %	91.3 %	38.4 %	75.1 %	41.9 %	86.4 %
test	(58.2 %)	(62.2 %)	(37.1 %)	(35.7 %)	(0.3 %)	(26.3 %)
Log-likelihood	77.7 %	90.9 %	52.0 %	64.5 %	1.1 %	58.7 %
ratio test	(78.8 %)	(90.3 %)	(53.9 %)	(60.8 %)	(0.8 %)	(52.5 %)
Welch's t-test	1.8 %	48.8 %	2.0 %	24.7 %	0.5 %	49.5 %
	(2.0 %)	(45.1 %)	(2.4 %)	(18.8 %)	(0.4 %)	(35.9 %)
Wilcoxon	2.0 %	48.9 %	4.2 %	30.0 %	0.1 %	5.2 %
rank-sum test	(2.0 %)	(44.5 %)	(4.6 %)	(21.4 %)	(0.1 %)	(3.8 %)
Permutation	6.8 %	63.7 %	3.3 %	61.5 %	23.7 %	80.2 %
test	(0.0 %)	(0.0 %)	(0.8 %)	(4.3 %)	(0.0 %)	(0.0 %)
Bootstrap test	1.8 %	49.1 %	2.0 %	24.9 %	0.5 %	49.4 %
	(2.0 %)	(45.4 %)	(2.5 %)	(19.1 %)	(0.4 %)	(36.2 %)
Empirical IAT	16.8 %	69.6 %	2.2 %	58.3 %	23.4 %	80.0 %
test	(10.8 %)	(15.6 %)	(0.3 %)	(6.1 %)	(0.0 %)	(0.0 %)
Weibull IAT	17.2 %	60.6 %	25.1 %	59.1 %	0.5 %	53.2 %
test	(8.7 %)	(7.7 %)	(1.1 %)	(3.5 %)	(0.0 %)	(0.0 %)

able choices. For assumption B, the empirical IAT test shows best performance. Welch's t-test and the bootstrap test score better on the two-tailed test, but are more prone to being anti-conservative, which is more problematic in this application than being conservative. Under assumption C, there are many good choices, for example the log-likelihood ratio test.

4.5.2 Fictional prose by male and female authors

The purpose of the first case study was to test if there are linguistic differences between male- and female-authored fiction prose. We have used the same data as in the previous section: the texts in the British National Corpus categorised as fiction prose, for which the gender of the author is known. These texts are parts of novels and short stories. The subcorpus contains 203 texts written by men, and 206 texts written by women, approximately 15.6 million words in total. We preprocessed the texts by lowercasing all words and ignored punctuation and other markup.

We conducted the experiment as follows. For all types (unique words), we computed the significance using the chi-squared test, log-likelihood ratio test, Welch's t-test, Wilcoxon rank-sum test, empirical inter-arrival time test and the bootstrap test. After obtaining the initial p-values for all types, we applied the method by Benjamini and Hochberg [1995] to control the false discovery rate (Equation 2.19) at α = 0.05. This ensures that the expected rate of false positives over all positives is at most 5 %. We also computed the normalised dispersion (Def. 2.10) for all words.

The bootstrap and inter-arrival time tests were implemented slightly differently than presented in Section 4.4: the two subcorpora were resampled separately, instead of grouping the data, resampling, and then splitting the data, and the empirical p-value (Equation 4.2) was used. This implementation has the disadvantage that for very skewed distributions, the p-value may be unreasonably low, and the analytical solution for the bootstrap test proposed here is also computationally more attractive, as is does not require resampling. As the size of the data grows, the implementations converge, and we expect the p-values to be very similar for the given data size, about 1500 texts in each period.

The most frequent ($\sigma \geq$ 5,000) words that are significantly more frequent in either subcorpus according to the bootstrap test are given in Tables 4.5 and 4.6. We find that the results are consistent with earlier research. Overall, male-authored fiction is dominated by frequent use of noun-related forms, while female-authored fiction is more verb-oriented. Male authors use articles (*a*, *the*) and prepositions (*by*, *from*, *in*, *of*, *on*, *through*) more frequently, both of which are associated with nouns. Similarly, male-authored fiction contains more function words that are typically associated with noun phrases and nominal functions, such as *another*, *first*, *one*, *some*, *two*, and *other*. The list of significant items for male authors is shorter than that for female authors, which suggests that male authors write slightly more repetitively.

The personal pronouns that are overrepresented in male-authored fiction are the first-person plural forms *us* and *we* and the third-person pronouns *its*, *their*, and *they*, while women's fiction overuses the second-person forms *you* and *your*, which can have singular and plural referents. Stereotypically, men tend to write about *man* and *he*, and women about *her* and *she*. These pronoun findings are consistent with those of Argamon et al. [2003], who use the same data, but differ in that women do not significantly favour the first-person pronoun *I*.

Table 4.5. High-frequency words that are significantly overrepresented in male-authored prose fiction in the BNC according to the bootstrap test. Observed frequencies are given per million words.

Word	Freq M	Freq F	DP	p	Word	Freq M	Freq F	DP	p
a	22,824	21,442	.06	.0001	they	5,233	4,270	.17	.0001
another	735	632	.14	.0001	through	1,267	992	.16	.0001
by	2,914	2,473	.13	.0001	two	1,333	1,004	.17	.0001
first	1,002	854	.13	.0001	us	937	605	.26	.0001
from	4,058	3,500	.10	.0001	we	3,651	2,663	.21	.0001
in	14,371	13,563	.06	.0001	were	3,738	3,238	.12	.0001
its	977	701	.26	.0001	is	4,521	3,588	.21	.0003
man	1,603	1,270	.21	.0001	left	806	717	.14	.0005
of	22,483	19,747	.09	.0001	other	1,229	1,096	.12	.0005
on	7,520	6,942	.07	.0001	there	4,111	3,650	.13	.0005
one	3,146	2,801	.09	.0001	are	2,206	1,858	.18	.0007
some	1,651	1,415	.14	.0001	where	1,297	1,147	.15	.0013
the	58,013	45,333	.09	.0001	he	17,295	15,587	.14	.0045
their	2,090	1,663	.20	.0001					

Table 4.6 shows that female-authored fiction is marked by frequent verb use: there are more than twenty verb forms among the items overused by women (forms of *be*, *do*, and *have*; modals, such as *could*, *should*, *must*, and *would*; and activity and mental verbs, including *come*, *go*, *make*, *knew*, and *thought*). Only three such verb forms are overused in male-authored fiction (*were*, *is*, and *are*). Particularly outstanding features in women's fiction are contracted forms (*'ll*, *'m*, *'ve*, *n't*, *'re*), negative particles (*n't*, *never*, *not*), and intensifiers (*much*, *so*, *too*, *very*). These are all indicators that female-authored fiction employs a more involved, colloquial style than male-authored fiction, which, in contrast, is marked by features associated with an informational, noun-oriented style. These results are similar to Biber [1995].

These results do not necessarily reflect gender differences, as author gender and target audience are correlated in the fiction prose subcorpus. The stories are mostly targeted at adults and a small portion at children, for both male and female authors. However, male authors wrote relatively more texts for a mixed gender audience, while female authors wrote mainly texts targeted at a female audience. Previous research indicates that audience design is relevant, for example, in spoken interaction, and style shifting is typically a response to the speaker's audience [Bell, 1984].

The above analysis is based on words that are ranked as significant by

Table 4.6. High-frequency words that are significantly overrepresented in female-authored prose fiction in the BNC according to the bootstrap test. Observed frequencies are given per million words.

Word	Freq M	Freq F	DP	p	Word	Freq M	Freq F	DP	p
'll	1,298	1,784	.24	.0001	should	753	952	.16	.0001
'm	1,287	1,733	.24	.0001	so	2,843	3,469	.12	.0001
've	1,124	1,465	.23	.0001	thought	1,216	1,647	.19	.0001
be	4,513	5,186	.10	.0001	to	24,755	26,756	.05	.0001
come	1,076	1,283	.15	.0001	too	1,160	1,368	.14	.0001
could	2,859	3,314	.12	.0001	want	841	1,071	.20	.0001
did	2,728	3,218	.14	.0001	when	2,455	2,853	.13	.0001
eyes	966	1,525	.26	.0001	with	6,755	7,494	.07	.0001
face	1,001	1,246	.21	.0001	would	3,207	3,876	.14	.0001
for	6,484	7,076	.07	.0001	you	11,017	14,261	.16	.0001
go	1,265	1,522	.16	.0001	your	1,703	2,234	.18	.0001
her	6,915	17,533	.29	.0001	had	8,837	10,176	.15	.0003
how	1,350	1,582	.13	.0001	look	900	1,081	.16	.0003
if	2,898	3,266	.11	.0001	take	760	858	.13	.0003
knew	792	988	.18	.0001	very	1,191	1,445	.22	.0003
made	986	1,168	.13	.0001	do	3,983	4,588	.15	.0005
make	742	882	.13	.0001	because	778	963	.23	.0007
much	919	1,099	.15	.0001	put	752	860	.18	.0023
must	841	995	.18	.0001	that	10,624	11,455	.10	.0029
n't	6,262	7,990	.20	.0001	little	1,064	1,238	.19	.0047
never	968	1,294	.17	.0001	're	1,193	1,412	.24	.0049
not	4,604	5,449	.16	.0001	have	4,271	4,626	.11	.0053
own	751	966	.17	.0001	well	1,322	1,499	.18	.0057
she	7,948	19,609	.28	.0001					

the bootstrap test and most of these words are also significant according to the other tests. However, it is also interesting to know how many words are marked as significant by the tests using the bag-of-words model, such as the chi-squared test, which are not significant according to the bootstrap test. Tables 4.7 and 4.8 list high-frequency words (5,000 or more occurrences in both subcorpora) for which the difference between the p-values from the chi-squared test and bootstrap test is at least tenfold. Using false-discovery rate control at $\alpha = 0.05$, all of the chi-squared test p-values are significant, but the bootstrap test p-values are not significant. Although not shown in the tables, all these words are also significant according to the log-likelihood ratio test.

Some of the words in Tables 4.7 and 4.8 appear to support the above

Table 4.7. High-frequency words that are significantly overrepresented in male-authored prose fiction in the BNC according to the chi-squared test, but not according to the bootstrap test. Observed frequencies are given per million words.

Word	Freq M	Freq F	p_{chi}	p_{boot}	Word	Freq M	Freq F	p_{chi}	p_{boot}
an	2,572	2,441	.000	.103	people	881	746	.000	.014
back	2,384	2,255	.000	.095	them	2,583	2,388	.000	.051
down	2,002	1,851	.000	.021	this	3,367	3,192	.000	.154
has	916	783	.000	.052	up	3,476	3,318	.000	.153
his	10,099	9,093	.000	.013	which	1,811	1,531	.000	.019
I	17,482	16,864	.000	.523	who	2,026	1,867	.000	.033
into	2,566	2,451	.000	.148	then	2,723	2,618	.000	.389
my	3,494	2,975	.000	.059	looked	1,376	1,314	.001	.429
off	1,232	1,121	.000	.021	something	1,036	979	.000	.191
old	897	824	.000	.193	just	1,912	1,840	.001	.447
or	2,397	2,085	.000	.014	turned	797	754	.003	.292
out	3,400	3,225	.000	.075					

analysis: the writing style of women is more verb-oriented, whereas men overuse masculine and collective personal pronouns, such as *his* and *them*. However, the list of words for female-authored fiction also includes a male personal pronoun, *him*, and men appear to significantly overuse the first-person singular pronouns *I* and *my*, which is surprising in view of earlier research on gendered styles [Argamon et al., 2003, Newman et al., 2008].

Furthermore, men appear to overuse directional adverbs, such as *back*, *down*, *out*, and *up*. If words of all frequencies are considered, then the most prominent category of words that are significant according to the chi-squared test but not the bootstrap test is proper nouns, as in the *Matilda* example (Section 4.1). Many of these words are easily misinterpreted as genuine differences between subcorpora.

Figure 4.5 summarises the number of significant words that were returned by each test for varying significance testing thresholds. The t-test yields the least number of significant words, followed by the Wilcoxon rank-sum and bootstrap tests in both figures. Only the curve for the inter-arrival time test differs substantially between the left and right figures. The inter-arrival time test appears to have difficulty with comparing zero with non-zero frequencies and always deems such cases significant. As noted at the beginning of this section, the implementation proposed in Section 4.4 does not suffer from this problem[3]. We also observe that the

[3]The p-value for the empirical inter-arrival time test proposed in Section 4.4 is always $p = 1$ for 0 vs. 1 comparisons, as the randomised data always has one

Table 4.8. High-frequency words that are significantly overrepresented in female-authored prose fiction in the BNC according to the chi-squared test, but not according to the bootstrap test. Observed frequencies are given per million words.

Word	Freq M	Freq F	p_{chi}	p_{boot}	Word	Freq M	Freq F	p_{chi}	p_{boot}
all	3,587	3,744	.000	.177	only	1,482	1,592	.000	.024
and	25,613	26,640	.000	.087	said	4,892	5,611	.000	.068
any	1,095	1,176	.000	.103	seemed	700	779	.000	.079
as	6,298	6,738	.000	.006	think	1,307	1,462	.000	.015
away	1,133	1,211	.000	.062	time	1,816	1,926	.000	.022
been	2,868	3,019	.000	.132	told	765	891	.000	.007
but	5,891	6,070	.000	.291	was	15,461	15,863	.000	.340
'd	1,715	2,063	.000	.057	why	977	1,070	.000	.043
day	746	811	.000	.090	room	793	850	.000	.222
going	1,048	1,151	.000	.075	know	1,971	2,055	.000	.299
him	4,752	5,087	.000	.088	about	2,604	2,698	.000	.336
last	711	791	.000	.008	even	1,133	1,189	.001	.263
might	828	912	.000	.066	after	1,187	1,240	.003	.155
no	2,942	3,150	.000	.009	long	879	925	.003	.111
now	2,024	2,206	.000	.014	tell	772	812	.006	.235

chi-squared and log-likelihood ratio tests mark several orders of magnitude more words as significant than the t-test, the Wilcoxon rank-sum test and bootstrap test.

4.5.3 Personal letters before and after the English Civil War

The purpose of the second case study is to test the diachronic continuity in terms of word frequencies of a corpus of personal letters that was designed to be as homogeneous over time as possible. This research question has a direct connection with potential bias when using a certain statistical test, as a conservative test may lead to the conclusion that a corpus is indeed stable, while it is not, and an anti-conservative test may lead to the conclusion that some aspects of the language change over time, while they do not. Full results and extensive discussion of the study have been presented in Lijffijt et al. [2012], while in this section only the most relevant results are presented.

We used the Standardised-spelling Corpus of Early English Correspondence [2012], see Section 3.2 for a general introduction. We used only the 17th-century part (1600–1681), because that is fairly homogeneous with

occurrence and thus the same difference in counts of 1.

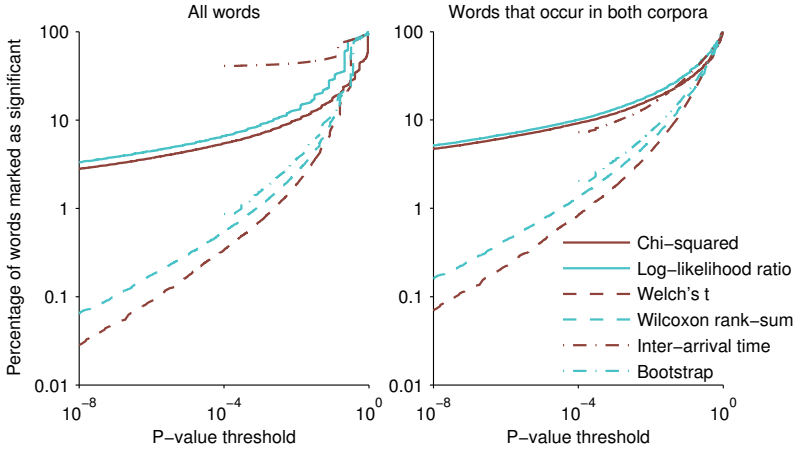

Figure 4.5. Comparison of the number of words marked as having significantly different frequency between the genders, for all six methods. For each method, a curve demonstrates how the number of significant words (y-axis) increases as the significance threshold (x-axis) increases in the male vs. female author comparison, here without correcting for multiple hypotheses. The figure on the left is based on all words, and the figure on the right includes only those words with frequencies greater than zero for both genders.

respect to author gender and social ranks. However, the English Civil War (1642–1651) is expected to have an impact on the language use, for example war-related words may be more frequent. We divided the data into two periods of roughly 40 years: 1600–1639 and 1640–1681, and compared these periods to each other. We analysed the data using the log-likelihood ratio test and the bootstrap test (using the same implementation as in the previous section) and also compared the results of the two tests.

The number of significant differences for both tests for various choices of α and minimum frequency thresholds is given in Table 4.9. We observe several trends: the number of words that have a significantly different frequency in the two periods decreases both as (1) the significance threshold α decreases, and (2) when the minimum frequency threshold increases. Besides, most of the words reported by the bootstrap test are also significant under the log-likelihood ratio test. For example, when employing a frequency threshold of 10 per 100,000, all words that are reported by the bootstrap test are also significant using the log-likelihood ratio test.

The log-likelihood ratio test marks 15–23 % more types as significant as the bootstrap test. This difference is considerably smaller than the differences observed in our other case study (Section 4.5.2) and in previous research [Paquot and Bestgen, 2009]. A possible explanation is the relatively small size of the corpus. For example, Paquot and Bestgen [2009]

Table 4.9. The number of words reported as significantly different between the two time periods by the log-likelihood ratio and bootstrap tests, for various frequency and significance thresholds. The numbers below 'both tests' show the overlap between the tests.

Min. frequency (per 100,000)	α	Log-likelihood ratio test	Bootstrap test	Both tests
	0.01	2,685	2,365	2,199
0	0.001	1,400	1,209	1,108
	0.0001	937	759	722
	0.01	2,365	2,034	2,013
1	0.001	1,400	1,209	1,108
	0.0001	937	759	722
	0.01	603	530	530
10	0.001	498	421	421
	0.0001	432	354	354

compares two sub-corpora that are almost 25 times as large.

Whether the corpus is diachronically comparable is not straightforward to answer. There are 46,440 different words in the data and most are indeed not significantly different. However, based on Table 4.9, we find that all words significant at the level $\alpha = 0.001$ have a frequency of at least 1 in 100,000. Thus, there is an implicit frequency requirement for a word to be marked as significant. Only 6,448 words satisfy the 1 in 100,000 constraint, indicating that 12–22 % of the words has significantly changed in frequency between the two periods.

Linguistic interpretation and analysis of the significant differences are presented in Lijffijt et al. [2012]. In particular, all significant differences with a frequency of 100 per 100,000 and all significant differences that appear in the *Society* > *Armed Hostility* section of the *Historical Thesaurus of the Oxford English Dictionary* are reviewed. These analyses are not included here because they were mostly carried out by the co-authors of the paper and for brevity.

4.6 Conclusion

In this chapter we studied the problem how to compare word frequencies across corpora. By modelling texts as event sequences and a text corpus as a database, we mapped the problem to the question *how to compare event frequencies across (databases) of event sequences*. This problem is relevant, for example, when a linguist wants to test a hypothesis such as

"word X is more frequent in male than in female speech". We have introduced two methods based on resampling, and we have compared and evaluated these methods, along with several existing methods, with respect to their suitability to this task.

We found that the choice of the test, or more specifically, the representation of the data that is used in the test, matters, both in theory and in practice, as evidenced by experiments and case studies on two text corpora. We concluded that assuming that all words are independent samples may lead to overestimating the significance of frequency differences. We demonstrated that the overestimation is related to the burstiness of words and that there exist bursty and non-bursty words at any frequency level, thus the overestimation occurs at all frequency levels.

5. Mining subsequences with surprising event counts

In this chapter, we consider the problem of mining subsequences with surprising event counts, and more specifically *how to take into account the multiple testing problem when looking for local frequency deviations in event sequences*. We introduce a method to find all subsequences of a long data sequence where the count of an event is significantly different from what is expected. In estimating what is expected, we have to take into account that we consider many subsequences concurrently. Existing methods for taking this into account are either computationally very demanding, or they do not account for any dependency structure.

The proposed method accounts for the dependency structure directly, by analysing the joint distribution of the patterns, while avoiding the use of computationally more demanding randomisation. We assert that computing the p-values for the method exactly is also computationally costly and introduce a simple and efficiently computable upper-bound that can be used instead. We provide empirical evidence that the upper-bound is more powerful than existing alternatives, and we demonstrate the utility of the method in experiments on two types of data, text and DNA.

5.1 Introduction

The amount of collected data is growing rapidly. As a result, the focus in data mining research is more than ever on faster and simpler methods, where fast currently means linear or sublinear in the size of the data. However, *big data* presents more challenges. For example, when mining *patterns*—local structure, as opposed to global structure [Mannila, 2002]—the number of patterns potentially present in the data is often exponential in the number of variables or features. Testing more patterns is nice, because it increases the likelihood of finding interesting results.

However, testing more patterns is also dangerous, as it increases the likelihood of finding *spurious* results, i.e., patterns caused by randomness.

The problem studied in this chapter is how to find interesting subsequences in a long event sequence. We restrict the set of subsequences by considering only subsequences of a given length, but even then the number of subsequences can be very large. Our aim is to find all subsequences where a given event is surprisingly frequent or infrequent. We define surprising as improbable under the assumption that the event sequence contains no structure.

Computing the statistical significance of an event count in a single subsequences is straightforward (see Section 5.4), but in this case we assess the statistical significance of many event counts concurrently. This makes it difficult to estimate what we should expect, even if there is no structure in the data. Several forms of *control* have proposed in the statistics literature to make p-values (probabilities) easier to interpret when testing multiple hypotheses. An overview of these studies is given in Section 5.2. Unfortunately, these methods rely either on randomisation of data, Bonferroni-style post-hoc correction, or both.

Statistical testing using randomisation is computationally expensive; a single randomisation has a computational cost linear in the size of the data or higher, and thousands or millions of randomisation may be required for sufficient resolution. Bonferroni-style post-hoc correction is also problematic, because the studied patterns (which each correspond to a hypothesis test) are typically dependent. In that case the p-values become *conservative*, i.e., too high, and true findings may go unnoticed. The larger the data, the worse the problem, because the number of patterns usually grows with the data, and the conservativeness grows with the number of patterns.

We propose a method for mining subsequences with surprising event counts based on a statistical test that includes a correction for testing multiple hypotheses. As such, the p-value, which is used as a measure for the surprisingness of an observation, depends on the observation, as well as on the size of the data, i.e., the event count of a subsequence and the length of the full sequence, respectively. The p-values are computed analytically and the use of a post-hoc correction and its possibly conservative effect on the p-values is avoided. Although the method is not directly applicable to other data or pattern types, it may act as a model for methods on other data.

The proposed method provides strong control over the *family-wise error rate*, which is the probability that any of the significant results is a false positive [Shaffer, 1995]. In other words, it answers the question "what is the probability that *any of the considered patterns* would have a statistic equal to or higher than the observed test statistic?", where the test statistic can be any interestingness measure, e.g., an event count (= support), or more complex measures such as lift, or weighted-relative accuracy, see Geng and Hamilton [2006] for an overview. The following example illustrates family-wise error rate control further.

Assume that the interestingness measure, and thus the test statistic, is the support of a pattern, and that the data is a transaction database in tabular form. For simplicity assume that all items have equal support. The probability that the statistic of a specific pattern P is significantly high can be assessed by, for example, using swap randomisation [Gionis et al., 2007] to generate randomised samples[1] and then computing how often a similar or higher statistic for pattern P occurs in the randomised samples. The obtained p-value answers the question "what is the probability that *this specific pattern* has a test statistic equal to or higher than the observed statistic?".

Now assume that this procedure is repeated for all itemsets of some fixed size. Because many hypotheses are tested, it is likely that many observations have small p-values. To prevent this, the observed statistics should be compared with the maximum observed statistic over all itemsets of that size, per randomisation. In that case, the p-values answer the question "what is the probability that *any of the considered patterns* would have a statistic equal to or higher than the observed statistic?", which is the same as family-wise error rate control. Significance testing with family-wise error rate control using randomisation for mining frequent itemsets is described in, e.g., Hanhijärvi [2011]. More related work is discussed in Section 5.2.

Our aim is to find subsequences of a fixed length where a certain event is significantly frequent or infrequent. This is essentially a *subgroup discovery* problem: the target is a specific event, the descriptions or patterns are subsequences, and the aim is to find all descriptions where the target is exceptionally frequent or infrequent. This problem setting has many applications. For example, biologists are interested in detecting *isochores*

[1]Which randomisation method to use depends on the assumptions that one wants to make.

and *CpG sites* in DNA sequences, which are regions that are especially rich or poor in GC content and rich in the dinucleotide CpG respectively [Bernardi, 2000], and another example is that in text analysis it is useful to identify text fragments where a certain word is under or overused.

Summary of contributions. We introduce a new method to test the significance of event frequencies in subsequences that provides p-values under control of the family-wise error rate. That is, the p-value corresponds to the probability of observing the observed statistic or higher in *any* of the subsequences of a given length in a single long sequence. We assert that computing the p-values exactly is computationally costly and derive a simple and efficiently computable upper bound. We investigate the tightness of the upper bound is and compare the power of the test to a generic post-hoc correction. We illustrate the utility of the method in experiments on two types of data, text and DNA. We find that the upper bound is sufficiently tight and that meaningful results can be obtained in practice.

Outline. Related work is discussed in Section 5.2. The problem statement and the method are introduced in Sections 5.3 and 5.4. Results from the experiments on the tightness of the upper bound, comparison with the generic post-hoc correction, and the experiments on the two data sets are presented in Section 5.5. Conclusions are given in Section 5.6.

5.2 Related work

The popularity of significance testing methods in data mining has increased considerably over the past decade. Gionis et al. [2007] introduced swap randomisation for mining significant patterns while maintaining row and column margins, and De Bie [2011] proposed a maximum-entropy approach that can also take into account other types of constraints. Webb [2008] and Hanhijärvi [2011] studied the problem of multiple testing for mining patterns. These studies are all restricted to mining itemsets or tiles. A generic approach to mining structure in data using statistical testing has been presented by Lijffijt et al. [in press].

There are only a few studies on statistical testing approaches for mining sequential data. Most related is the statistical test proposed by Kifer et al. [2004] for detecting change points in streams. However, they rule out the possibility of controlling the family-wise error rate, as they consider only streams of infinite length. Another drawback of that method is that the critical points cannot be computed analytically, but require ran-

domisation. Haiminen et al. [2008] proposed a method for determining the significance of co-occurrences of events in event sequences. However, that method is computationally intensive as it relies on randomisation.

An alternative method for modelling frequency variation is sequence segmentation, although the focus is then on global modelling, while the aim here is to find local structure. Mannila and Salmenkivi [2001] study efficient methods for sequence segmentation, while the method by Lijffijt et al. [in press] can be used to assess the significance of such a segmentation. Complementary to this work are the methods for comparing event counts between databases of sequences put forward by Lijffijt et al. [2011].

5.3 Problem statement

Our aim is to find parts of a sequence where a particular event is over or underrepresented. We consider an approach where we compute the frequency of an event in all subsequences of a given length, and then compute the statistical significance of the observed frequencies. The null hypothesis of interest is that the data has no structure, i.e., all events in the sequence are i.i.d. samples, and that the event occurs at each position with the same probability p.

We assume that the parameter p, which is used to define the null hypothesis, is fixed. The choice of p determines the perspective for the significance test. For example, the parameter p may be estimated from the sequence S, in which case the method finds regions in the sequence where the event frequency is significantly high (or low) with respect to the sequence as a whole. Alternatively, p can be based on background knowledge, for example an estimate derived from a large database of sequences.

The focus here is on computing p-values while controlling the *family-wise error rate* (Definition 2.18), i.e., the probability that at least one true null hypothesis is declared significant. The family-wise error rate depends on the arrangement and number of the tested subsequences. Three scenarios are considered here: (1) testing a single subsequence, (2) testing all subsequences of a given length, and (3) testing subsequences using a sliding window with a given step size.

The basic definitions for significance testing that are used in this chapter are presented in Sections 2.4 and 2.5. The basic notation for event sequences is presented in Section 2.1. We assume that the user chooses a priori a significance threshold α, and the goal is to control the family-wise

error rate at level α, thus the only definition that is required to complete the problem setting is the cumulative probability mass function of the event frequency, i.e., the null hypothesis, $\Pr(\{T \geq k\})$. We consider three scenarios, each with a different null hypothesis:

1. Testing a single subsequence at a random location: $\Pr(\{T \geq k\})$ is the probability that in a subsequence of length m there are k or more ones. Under the null hypothesis, each event is assumed to be independently generated by a Bernoulli distribution with parameter p. Thus, the probability distribution for the number of ones in a single subsequence at a random location is a Binomial distribution, $\Pr(\{T = k\}) = Bin(k; m, p)$, and the cumulative probability mass function is:

$$\Pr(\{T \geq k\}) = \sum_{i=k}^{m} Bin(i; m, p) = \sum_{i=k}^{m} \binom{m}{i} p^i (1-p)^{m-i}.$$

   A stepwise derivation is given in Section 5.4.

2. Testing all subsequences of length m: as stated previously, when testing many observations with equal null hypotheses, family-wise error rate control corresponds to asking the question "what is the probability that *any* of the null hypotheses is equal to or higher than the observed statistic?". More formally: for an event sequence of length n, there are $n - m + 1$ subsequences of length m. Let $Z_1, \ldots, Z_{n-m+1}$ denote random variables corresponding to the test statistics of the null hypotheses, one for each subsequence. To provide control of the family-wise error rate, the p-values should be smaller than or equal to

$$\Pr(\{T \geq k\}) = \Pr\left(\bigcup_{j=1}^{n-m+1} \{Z_j \geq k\}\right).$$

   Adjusted p-values that are controlled for the family-wise error rate can also be obtained by computing the significance of each of the subsequences (Z_j's) separately, and applying Bonferroni correction or an improved variant. The adjusted p-values give an upper bound on the true p-values, as specified in the equation above. However, as the Z_j's are strongly correlated, we expect these upper bounds to be very weak, thus leading to loss of statistical power. In Section 5.4, we argue that it is computationally costly to evaluate this probability distribution exactly and we derive a tighter upper bound that can be computed efficiently.

3. Testing subsequences using a sliding window with a given step size:

this scenario is comparable to the previous, but only a subset of the subsequences are tested. Given a step size r, the subsequences that are tested have indices $1, 1+r, 1+2r, \ldots, 1+\left\lfloor \frac{n-m}{r} \right\rfloor r$. To provide control of the family-wise error rate in this scenario, the p-values should be smaller than or equal to

$$\Pr\left(\{T \geq k\}\right) = \Pr\left(\bigcup_{j=0}^{\left\lfloor \frac{n-m}{r} \right\rfloor} \{Z_{1+j\cdot r} \geq k\}\right).$$

This probability distribution is also studied further in Section 5.4.

5.4 Methods

5.4.1 Testing one subsequence

Given a sequence of independent random variables $X_1, \ldots, X_n$, each following a Bernoulli distribution with parameter p, define $Z_{i,m}$ as

$$Z_{i,m} = \sum_{j=i}^{i+m-1} X_j.$$

Thus, $Z_{i,m}$ is the sum over a subsequence of the random variables of length m starting at position i.

Because $Z_{i,m}$ is the sum of m i.i.d. Bernoulli variables, the probability distribution for $Z_{i,m}$ is a binomial distribution:

$$\Pr\left(\{Z_{i,m} = k\}\right) = Bin\left(k; m, p\right) = \binom{m}{k} p^k (1-p)^{m-k}.$$

This distribution is independent of the position i.

Using this definition, it is straightforward to define the one-tailed p-value under the null hypothesis for a single subsequence at a random location (scenario 1). For the high frequency direction, the one-tailed p-value is given by

$$\begin{aligned} p_H &= \Pr\left(\{\sigma_a(S_{i,m}) \geq k\}\right) \\ &= \Pr\left(\{Z_{i,m} \geq k\}\right) \\ &= \sum_{j=k}^{m} \binom{m}{j} p^j (1-p)^{m-j} \\ &= 1 - \sum_{j=0}^{k-1} \binom{m}{j} p^j (1-p)^{m-j}, \end{aligned} \tag{5.1}$$

while the one-tailed p-value in the low frequency direction is given by

$$\begin{aligned} p_L &= \Pr\left(\{\sigma_a(S_{i,m}) \leq k\}\right) \\ &= \Pr\left(\{Z_{i,m} \leq k\}\right) \\ &= \sum_{j=0}^{k} \binom{m}{j} p^j (1-p)^{m-j}. \end{aligned} \tag{5.2}$$

As can be seen, the p-values correspond to the cumulative distribution function of the binomial distribution. These tests are also known as the *binomial test*. Many statistical software packages contain a function for computing its value.

5.4.2 Testing all subsequences of length m

When testing a single subsequence at a random location, the probability of rejecting the null hypothesis while it is actually true—a *false positive* (Def 2.16) or *type I error*—is exactly α, and thus the result is easy to interpret. However, when testing the significance of the frequency of multiple subsequences, or a subsequence at an optimised location, the probability of false positives increases.

Assume that the observed frequencies for all subsequences of a given length are tested. For example, this is the case when studying data using a sliding window with step size one. In this case, the probability under the null hypothesis of observing a certain frequency k or higher in at least one subsequence of length m is

$$\Pr\left(\bigcup_{i=1}^{n-m+1} \{Z_{i,m} \geq k\}\right). \tag{5.3}$$

When testing the frequency of an event in all subsequences, it seems reasonable to use this probability as a p-value. This is also theoretically justified: the probability expressed in Equation 5.3 is equal to the probability of obtaining at least one false positive, thus, using this as the p-value corresponds to strong control of the family-wise error rate [Shaffer, 1995].

Thus, in this scenario, the p-value in the high frequency direction is

$$p_H = \Pr\left(\bigcup_{i=1}^{n-m+1} \{Z_{i,m} \geq k\}\right).$$

The p-value can be decomposed as

$$\begin{aligned} p_H = \; & \Pr\left(\{Z_{1,m} \geq k\}\right) + \\ & \Pr\left(\{Z_{2,m} \geq k\} \cap \{Z_{1,m} < k\}\right) + \\ & \Pr\left(\{Z_{3,m} \geq k\} \cap \{Z_{1,m} < k\} \cap \{Z_{2,m} < k\}\right) + \ldots + \\ & \Pr\left(\{Z_{n-m+1,m} \geq k\} \cap \bigcap_{i=1}^{n-m} \{Z_{i,m} < k\}\right), \end{aligned} \tag{5.4}$$

which highlights that the p-value for this scenario is equal to the p-value in the previous scenario ($\Pr\left(\{Z_{1,m} \geq k\}\right)$, see Equation 5.1) plus a set of terms that can be interpreted as the correction terms for multiple testing.

These correction terms are in general difficult to compute exactly. A straightforward approach is the following: define a column vector v with a probability for each possible subsequence, and a transition matrix W that specifies the transition probabilities between the subsequences. All subsequences with $\geq k$ ones can be collapsed into a single state with one outgoing link to itself. The exact p-value is given by computing $W^{n-m} \cdot v$. However, the matrix W has $O(2^{2m})$ entries, so this approach is feasible only when m is very small.

The main result of this chapter is that an upper bound can be derived that is very easy to compute. Consider the following approximation:

$$\tilde{p}_H = \Pr\left(\{Z_{1,m} \geq k\}\right) + (n-m) \cdot \Pr\left(\{Z_{2,m} \geq k\} \cap \{Z_{1,m} < k\}\right).$$

Theorem 5.1. *$\tilde{p}_H$ is an upper bound on the exact p-value p_H, i.e., $\tilde{p}_H \geq p_H$.*

Proof. For the correction terms of p_H (Equation 5.4) it holds that

$$\begin{aligned} & \Pr\left(\{Z_{2,m} \geq k\} \cap \{Z_{1,m} < k\}\right) \\ & \geq \Pr\left(\{Z_{3,m} \geq k\} \cap \{Z_{2,m} < k\} \cap \{Z_{1,m} < k\}\right) \\ & \geq \Pr\left(\{Z_{4,m} \geq k\} \cap \{Z_{3,m} < k\} \cap \{Z_{2,m} < k\} \cap \{Z_{1,m} < k\}\right) \\ & \geq \ldots \\ & \geq \Pr\left(\{Z_{n-m+1,m} \geq k\} \cap \bigcap_{i=1}^{n-m} \{Z_{i,m} < k\}\right). \end{aligned} \tag{5.5}$$

Combining Equations 5.4 and 5.5 gives

$$\begin{aligned} p_H = \; & \Pr\left(\{Z_{1,m} \geq k\}\right) + \\ & \Pr\left(\{Z_{2,m} \geq k\} \cap \{Z_{1,m} < k\}\right) + \\ & \Pr\left(\{Z_{3,m} \geq k\} \cap \{Z_{1,m} < k\} \cap \{Z_{2,m} < k\}\right) + \ldots + \\ & \Pr\left(\{Z_{n-m+1,m} \geq k\} \cap \bigcap_{i=1}^{n-m} \{Z_{i,m} < k\}\right) \\ & \leq \Pr\left(\{Z_{1,m} \geq k\}\right) + (n-m) \cdot \Pr\left(\{Z_{2,m} \geq k\} \cap \{Z_{1,m} < k\}\right). \end{aligned}$$

Thus, $\tilde{p}_H$ is an upper bound on the exact p-value p_H. □

The upper bound can be computed as follows. The first term of $\tilde{p}_H$ can be evaluated using Equation 5.1, while the second term can be rewritten as:

$$\begin{aligned}
&\Pr\left(\{Z_{2,m} \geq k\} \cap \{Z_{1,m} < k\}\right) \\
&= \Pr\left(\{Z_{1,1} = 0\} \cap \{Z_{2,m-1} = k-1\} \cap \{Z_{m+1,1} = 1\}\right) \\
&= \Pr\left(\{Z_{1,1} = 0\}\right) \cdot \Pr\left(\{Z_{2,m-1} = k-1\}\right) \cdot \Pr\left(\{Z_{m+1,1} = 1\}\right) \\
&= (1-p) \cdot Bin\left(k-1; m-1, p\right) \cdot p.
\end{aligned}$$

Both the binomial and cumulative binomial distributions can be approximated algorithmically in $O(1)$ time, see, e.g., [Loader, 2000]. Thus, the upper bound can be computed in constant time.

The proposition here is to use the upper bound $\tilde{p}_H$ as a statistical test. Because $\hat{p}_H$ is an upper bound, a test based on $\hat{p}_H$ may be conservative under the null hypothesis, but not anti-conservative. In other words, the results may be statistically more significant under the null hypothesis, but not less significant. This is important, because the test based on $\hat{p}_H$ then also provides strong control of the family-wise error rate.

As the exact p-value p_H is difficult to compute, it is not possible to analyse directly how tight the upper bound is. In Section 5.5.1, empirical results on the tightness of the approximation are presented, and in Section 5.5.2 the power of this test is compared with the alternative of combining the binomial test with a general post-hoc correction.

To complete the method, an upper bound to the one-tailed p-value in the low direction is derived, analogously to the previous case. For brevity, only the result is given. Define

$$\tilde{p}_L = \Pr\left(\{Z_{1,m} \leq k\}\right) + (n-m) \cdot \Pr\left(\{Z_{2,m} \leq k\} \cap \{Z_{1,m} > k\}\right).$$

Theorem 5.2. *$\tilde{p}_L$ is an upper bound on the exact p-value p_L, i.e., $\tilde{p}_L \geq p_L$.*

Proof. Analogous to Theorem 5.1. □

The correction term can be computed using

$$\Pr\left(\{Z_{2,m} \leq k\} \cap \{Z_{1,m} > k\}\right) = p \cdot Bin\left(k; m-1, p\right) \cdot (1-p).$$

5.4.3 A generalisation for step sizes larger than one

When using a sliding window with step size larger than one, fewer hypotheses are tested and the dependency between the consecutive subsequences is different. The upper bound from Section 5.4.2 is also an upper

bound when using a step size larger than one, but a tighter bound can be obtained relatively easily using the same techniques as in Section 5.4.2, but with some additional rewriting.

Let r be the user-defined step size. In this scenario, the subsequences with indices $1, 1+r, 1+2r, \ldots, 1+\left\lfloor \frac{n-m}{r} \right\rfloor r$ are tested for an event being significantly frequent in that subsequence. The corresponding p-value that provides strong control over the family-wise error rate is

$$p_H = \Pr\left(\bigcup_{i=0}^{\left\lfloor \frac{n-m}{r} \right\rfloor} \{Z_{1+i\cdot r,m} \geq k\} \right).$$

There are $1 + \left\lfloor \frac{n-m}{r} \right\rfloor$ null hypotheses. The approximation $\tilde{p}_H$ for this scenario is defined as

$$\tilde{p}_H = \Pr\left(\{Z_{1,m} \geq k\}\right) + \left\lfloor \frac{n-m}{r} \right\rfloor \cdot \Pr\left(\{Z_{1+r,m} \geq k\} \cap \{Z_{1,m} < k\}\right).$$

Theorem 5.3. *$\tilde{p}_H$ is an upper bound on the exact p-value p_H, i.e., $\tilde{p}_H \geq p_H$.*

Proof. p_H can be decomposed as

$$\begin{aligned} p_H = {} & \Pr\left(\{Z_{1,m} \geq k\}\right) + \Pr\left(\{Z_{1+r,m} \geq k\} \cap \{Z_{1,m} < k\}\right) + \ldots \\ & + \Pr\left(\left\{ Z_{1+\left\lfloor \frac{n-m}{r} \right\rfloor r,m} \geq k \right\} \cap \bigcap_{i=0}^{\left\lfloor \frac{n-m}{r} \right\rfloor - 1} \{Z_{1+i\cdot r,m} < k\} \right). \end{aligned} \tag{5.6}$$

Also, it holds that

$$\begin{aligned} & \Pr\left(\{Z_{1+r,m} \geq k\} \cap \{Z_{1,m} < k\}\right) \\ & \geq \Pr\left(\{Z_{1+2r,m} \geq k\} \cap \{Z_{1+r,m} < k\} \cap \{Z_{1,m} < k\}\right) \\ & \geq \Pr\left(\{Z_{1+3r,m} \geq k\} \cap \{Z_{1+2r,m} < k\} \cap \{Z_{1+r,m} < k\} \cap \{Z_{1,m} < k\}\right) \\ & \geq \ldots \end{aligned} \tag{5.7}$$

Combining Equations 5.6 and 5.7 gives

$$p_H \leq \Pr\left(\{Z_{1,m} \geq k\}\right) + \left\lfloor \frac{n-m}{r} \right\rfloor \cdot \Pr\left(\{Z_{i+r,m} \geq k\} \cap \{Z_{1,m} < k\}\right).$$

Thus, $\tilde{p}_H$ is an upper bound on the exact p-value p_H. □

To compute the correction term, it can be rewritten as follows. First, we divide the term into three parts: the overlap between the two subsequences, $Z_{1+r,m-r}$, and the two non-overlapping parts, $Z_{1,r}$ and $Z_{1+m,r}$. Secondly, it holds that

$$\{Z_{1+r,m} \geq k\} \Rightarrow \{Z_{1+r,m-r} + Z_{1+m,r} \geq k\}, \text{ and}$$

$$\{Z_{1,m} < k\} \Rightarrow \{Z_{1,r} + Z_{1+r,m-r} < k\}.$$

Both right hand sides are satisfied simultaneously if and only if

$$\begin{aligned} \{Z_{1+m,r} \geq k - Z_{1+r,m-r}\},\ \{Z_{1+r,m-r} \geq k - Z_{1+m,r}\}, \\ \{Z_{1,r} < k - Z_{1+r,m-r}\}, \text{ and } \{Z_{1+r,m-r} < k - Z_{1,r}\}. \end{aligned} \tag{5.8}$$

Since $Z_{1+m,r}$ and $Z_{1,r}$ are both by definition between 0 and r, it holds that

$$\{k - r \leq Z_{1+r,m-r} < k\}. \tag{5.9}$$

Finally, the correction term can be rewritten to an explicit sum using Equations 5.8 and 5.9. Let $b = \max(0, k - r)$, we find that

$$\begin{aligned} &\Pr(\{Z_{1+r,m} \geq k\} \cap \{Z_{1,m} < k\}) \\ &= \sum_{j=b}^{k-1} \Pr(\{Z_{1+r,m-r} = j\} \cap \{Z_{1+m,r} \geq k - j\} \cap \{Z_{1,r} < k - j\}) \\ &= \sum_{j=b}^{k-1} \Pr(\{Z_{1+r,m-r} = j\}) \cdot \Pr(\{Z_{1+m,r} \geq k - j\}) \cdot \Pr(\{Z_{1,r} < k - j\}) \\ &= \sum_{j=b}^{k-1} \left(Bin(j; m - r, p) \cdot \sum_{l=k-j}^{r} Bin(l; r, p) \cdot \sum_{l=0}^{k-j-1} Bin(l; r, p) \right). \end{aligned}$$

One may verify that the result for $r = 1$ is the same as in Section 5.4.2. The binomials can be computed in constant time, thus the computational complexity of the correction term is $O(\min(k, r) \cdot \max(k, r)) = O(kr)$ and independent of the size of the full sequence. An upper bound $\tilde{p}_L$ can be derived analogously.

5.5 Experiments

The results of the experiments on the power of the test are discussed in Sections 5.5.1 and 5.5.2. In 5.5.1, we review how far the p-values from the test are from the ideal distribution for synthetic data generated under the null hypothesis, and in 5.5.2, we compare the proposed method with the alternative of post-hoc correction. We investigated the practical utility of the test using two types of data: an English novel and a part of the human reference genome. The findings are presented in Sections 5.5.3 and 5.5.4.

5.5.1 Tightness of the upper bound

Since the proposed test provides strong control for the family-wise error rate, the probability of observing one or more false positives is at most α. However, this provides no information on the *power* of the test, i.e., the probability of false negatives (Def 2.17). The probability or rate of false

negatives cannot be specified directly, because it depends on the alternative hypothesis; there is no general false negative rate. Instead, we use the fact that there is a trade-off between the probability false positives and the probability of false negatives to study the power of the test.

By definition, the probability of false negatives is minimised when the probability of false positives is maximised. The probability of false positives is limited from above to α due to control for the family-wise error rate. Thus, to minimise the probability of false positives, the probability of observing one or more false positives should be as close to α as possible. The results from the following experiment show how close the probability of encountering one or more false positives is in practice.

We conducted the experiment as follows. The tightness of the upper bound may depend both on the length of sliding window, as well as on the event probability. Thus, we generated 1,000 sequences of length $n = 9,999 + m$ (such that there are 10,000 p-values per sequence) for various combinations of window lengths ($m \in \{100, 1000, 10000\}$) and event probabilities ($p \in \{0.001, 0.01, 0.1\}$). Then, we computed the p-values $\tilde{p}_H$ for all subsequences using a sliding window with step size 1.

The quantity of interest in the experiments is the minimal p-value per sequence. Ideally, the distribution of minimal p-values over the sequences is uniform, which means that for any value α, the probability of a false positive is exactly α itself. This ensures that the probability of false positives is maximal (while providing family-wise error rate control), and that the probability of false negatives is minimal, for any α. Note that this holds by definition for the exact p-values under the null hypothesis, but the upper bound may have a higher probability of false negatives.

The results from the experiment are presented in Figure 5.1. We find that the p-values are reasonably close to the optimal distribution and that they are further from the optimal distribution when the expected number of events ($= m \cdot p$) is larger. The largest observed effect is approximately 1 order of magnitude ($m =$ 10,000, $p = 0.1$), indicating that the p-values are 1 order of magnitude too high in that case. The results for very low event probabilities (e.g., $m = 100$, $p = 0.001$) may appear more conservative, but they are skewed mostly because there are very few distinct p-values: the highest number of events observed in any subsequence is 3 ($\tilde{p}_H = 0.0437$), and for event counts 0 or 1, $\tilde{p}_H = 1$.

Estimates for p-values that are conservative by one order of magnitude are not a problem in most practical settings; much larger differences in

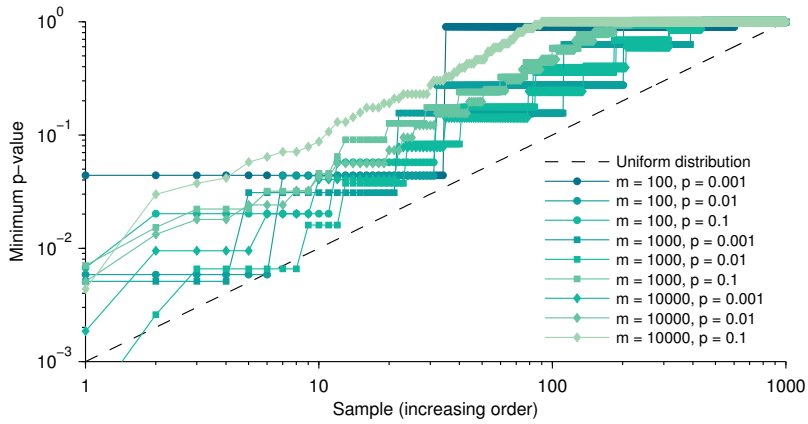

Figure 5.1. The distribution of minimal p-values over 1,000 synthetic sequences for the proposed method, using various window lengths m and event probabilities p, compared to the uniform distribution.

the choice of α can be observed in the literature: from $\alpha = 0.1$ to $\alpha = 0.00001$. Also, because the p-values are controlled for family-wise error rate, use of a 'large' α, such as 0.05, still guarantees that there is a low probability of obtaining any false-positive results.

5.5.2 Comparison to Hochberg's step-up procedure

An alternative approach to obtaining p-values with strong control of the family-wise error rate, for the same null hypothesis, is to use the binomial test (Equations 5.1 and 5.2) with post-hoc correction. The correction with largest power that provides strong control for the family-wise error rate, which is applicable in this setting, and that does not require specifying the dependency structure of the p-values, is Hochberg's step-up procedure [Hochberg, 1988]. Hochberg's procedure is valid for independent and positively dependent p-values [Sarkar and Chang, 1997]. The p-values in the setting considered in this chapter are positively dependent, because the event frequencies for overlapping subsequences have positive correlation.

To compare the power of the proposed method with the alternative using Hochberg's procedure, we conducted the following experiment. For each sequence generated in the previous experiment (Section 5.5.1), we computed the p-values for all subsequences of the same lengths using the binomial test, and then adjusted the p-values using Hochberg's procedure (per sequence). This ensures that the p-values are directly comparable to those in the previous experiment. Then, we compared the minimal p-values per sequence with those from the upper-bound method.

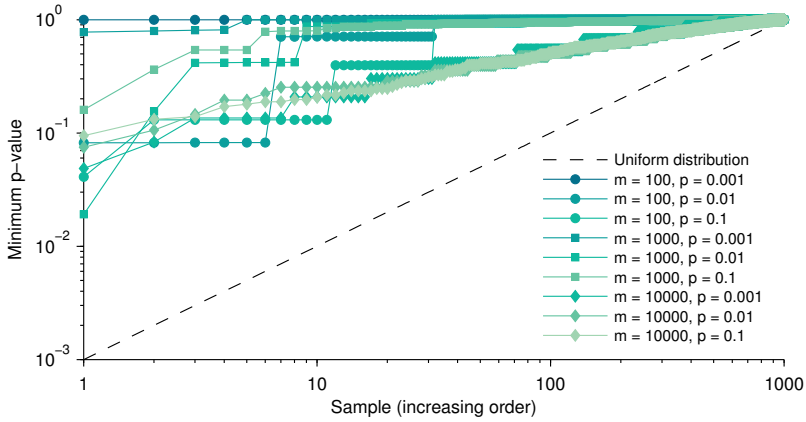

Figure 5.2. The distribution of minimal p-values for the binomial test with Hochberg's post-hoc correction, on the same data as in Figure 5.1.

The distribution of minimal p-values is shown in Figure 5.2. We observe that the p-values from the method with post-hoc correction are far from uniform, for any combination of parameters, while the distribution becomes more uniform as the expected number of events per subsequence increases. The proposed method outperforms the post-hoc approach for any combination of parameters, although it cannot be guaranteed that this holds for much larger expected event counts.

Figure 5.3 gives a direct comparison of the minimal p-values for both methods, per synthetic sequence. We observe even more clearly that the proposed method has superior performance for p-values ≤ 0.1 for any choice of parameters. For example, we find that the smallest p-value output by Hochberg's procedure is 0.02, while the upper-bound method yielded a p-value of 0.02 or lower in 35 sequences. The figure also supports the hypothesis that the difference between the methods is smaller when the expected number of events in a subsequence is larger.

5.5.3 Bursty and non-bursty words in an English novel

The prime motivation for the method proposed in this chapter comes from the domain of text analysis. Church and Gale [1995] and Katz [1996] both studied *burstiness* of words in the context of probabilistic modeling of word counts, and the concept is related to relevance measures in information retrieval, such as inverse document frequency [Spärck Jones, 1972]. More recently, using a quantification of burstiness based on the inter-arrival time distributions of words, burstiness of words has been related to semantic categories [Altmann et al., 2009], statistical tests for

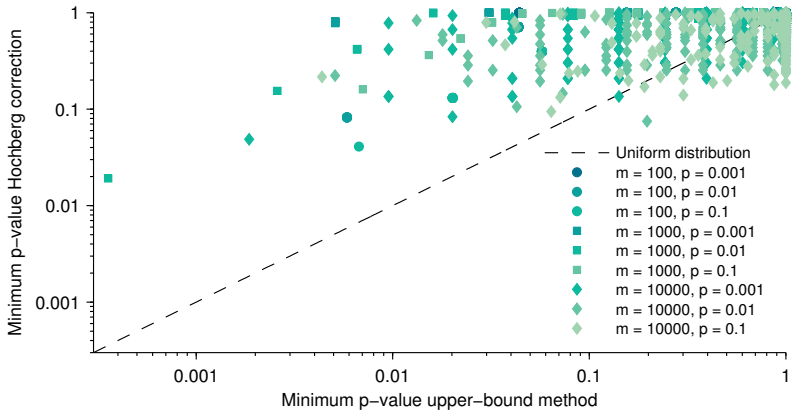

Figure 5.3. Minimal p-values per sequence between the proposed upper-bound method (x-axis) vs. the binomial test combined with Hochberg's procedure (y-axis). Each dot corresponds to a synthetic sequence.

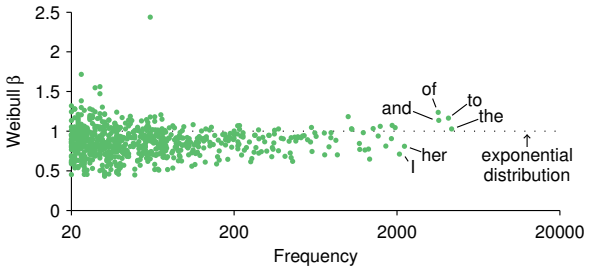

Figure 5.4. The relationship between *burstiness*, measured using the Weibull distribution, and *frequency* of words. Each dot represents a word in the novel *Pride and Prejudice*.

comparing corpora that take into account burstiness have been proposed (see Chapter 4), and the impact of burstiness on choosing optimal window lengths for sequence analysis has been studied (see Chapter 6).

For the purpose of text analysis, it is useful to know if there are fragments in a text where a certain word is over or underused and to locate such fragments. We investigated the suitability of the proposed method to this task, using the following experiment. We computed the frequency and burstiness of all words in the novel *Pride & Prejudice* by Jane Austen (see Section 3.3), using the definitions provided in Sections 2.1, 2.2, and 2.3. Then, we selected the five most and least bursty words in two frequency bins, see Table 5.1. An overview of the relation between frequency and burstiness is given in Figure 5.4.

For each of the selected words, we computed the frequency throughout the book using a sliding window of length 5,000 and step size 1. The book contains $n = 121{,}892$ words and thus there are 116,893 subsequences.

	Low frequency [40–50]	High frequency [300-600]
Non-bursty	hardly, help, perfectly, point, scarcely	an, elizabeth, more, there, when
Bursty	marry, pride, read, rosings, william	are, me, their, will, your

Table 5.1. The five least and most bursty words in two frequency bins in the novel *Pride & Prejudice*. We investigated the local behaviour of these words to study the suitability of the proposed method to locate over and underuse of words in text.

We used a window length of 5,000 to ensure that low event counts could also be significant; for example, for a window length of 2,000 and event probability $p = 1/300$, the p-value for $k = 0$ is $\tilde{p}_L = 0.4833$. Thus, an event count of zero would not be significant, even for fairly frequent words. With a window length of 5,000, event counts of 3 and less are significant at $\alpha = 0.05$ ($\tilde{p}_L \leq 0.0164$).

We computed the significance of the observed frequencies for both the high and low direction. Because the results are for illustrative purposes, we did not apply any additional correction for testing multiple sets of hypotheses. Figure 5.5 shows the results for four words. The word *an* is frequent and non-bursty, and no parts of the book show significant under or overuse of the word. For the pronoun *me*, which is frequent and bursty, there are two areas with overuse, and four areas with underuse, compared to the average frequency. The name of the main character, *Elizabeth*, is non-bursty ($\beta = 1.05$) and frequent throughout the novel, except for two parts. Finally, the family name *Rosings*, which is infrequent and bursty, is used a lot in two text fragments and occurs only rarely in other parts of the book.

An overview of results is given in Table 5.2. As expected, each of the bursty words is significantly over or underrepresented in at least one fragment of the book. Surprising is that some frequent words that are non-bursty according to the Weibull distribution estimate are still under or overused in one or more fragments. This indicates that there is local structure that is not captured by the Weibull measure of word burstiness. We find that the results from the proposed method are supported by visual inspection of the data.

5.5.4 Variation in GC and TA content in DNA

Variation of GC content in DNA sequences is used to define *isochores*, regions in DNA sequences where the GC content is approximately the same,

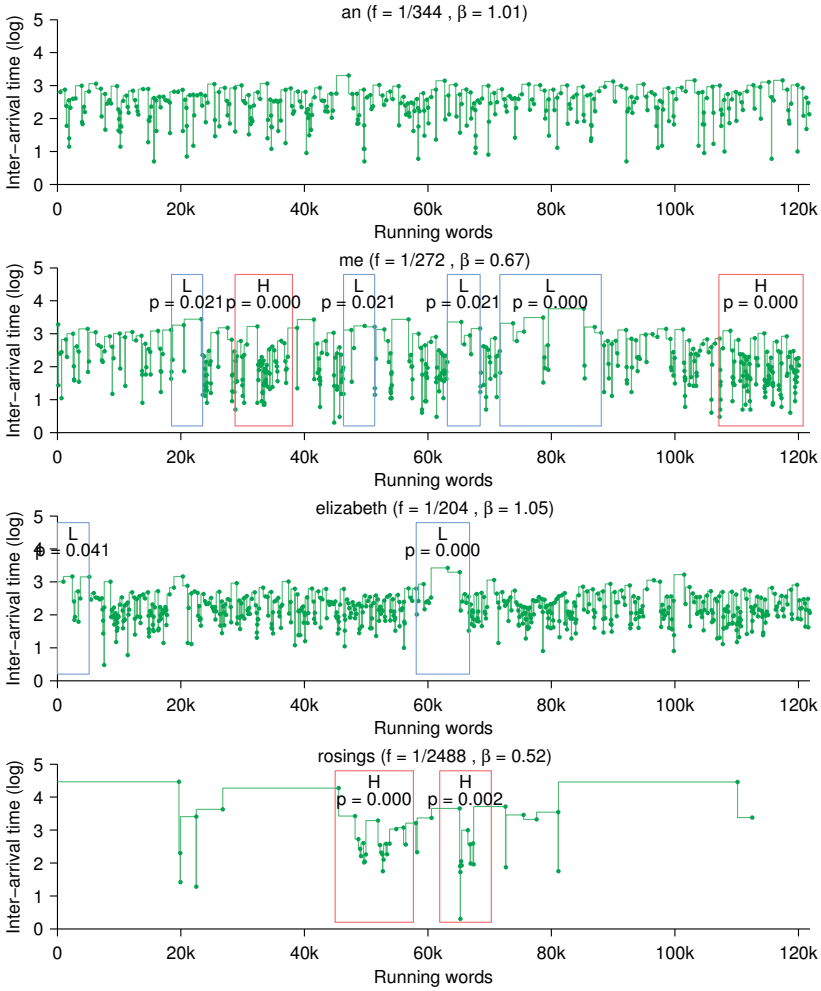

Figure 5.5. Regions of significant over (H) and underuse (L) for four words in the novel *Pride and Prejudice*, compared to the average frequency in the book. Each dot corresponds to an occurrence of the word in the text, lines are added to highlight the distances between consecutive occurrences. For the visualisation, we merged overlapping significant subsequences into longer subsequences.

Non-bursty						Bursty					
Frequent			Infrequent			Frequent			Infrequent		
Word	L	H	Word	L	H	Word	L	H	Word	L	H
an	0	0	hardly	0	0	are	1	0	marry	0	1
elizabeth	2	0	help	0	0	me	4	2	pride	0	1
more	0	0	perfectly	0	0	their	1	0	read	0	2
there	0	1	point	0	0	will	2	3	rosings	0	2
when	0	0	scarcely	0	0	your	2	3	william	0	1

Table 5.2. Number of areas with significant underuse (L) or overuse (H) for each of the twenty words.

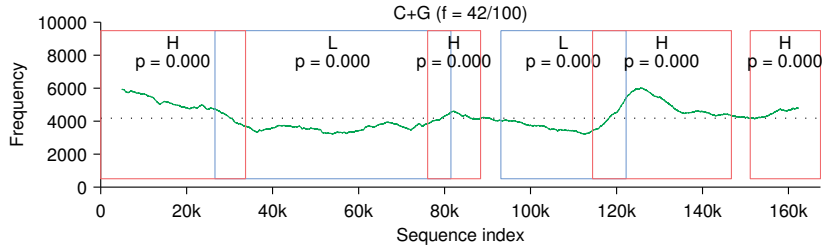

Figure 5.6. Analysis of the GC content at the start of Chromosome 1 of the Homo Sapiens reference genome, using a sliding window of length 10,000. Overlapping significant parts have been merged for the purpose of visualisation.

which in turn are used to identify gene structure [Bernardi, 2000]. The method introduced in this chapter seems particularly suitable to finding significant variation of nucleotide frequencies in DNA sequences. To test this, we conducted the following experiment. We computed the frequency of C+G for all subsequences of length 10,000 in chromosome 1 from the Homo Sapiens reference genome (build 37, patch 9, see Section 3.4). The reference build contains 225,280,621 fixed nucleotides, thus the number of hypotheses tested concurrently is very large in this case.

The first consecutive fixed part is illustrated in Figure 5.6. We observe that the test is sufficiently powerful, because several parts of the sequence are identified as having significantly high or low GC content. We find also that the GC content is quite volatile: the parts where the content is significantly low and high overlap each other, which indicates that the frequency goes up and down rapidly. The regions where the GC content is significantly high and significantly low can overlap because we consider fairly long subsequences (length 10,000). Although the figure shows only part of the whole sequence, the findings contradict previous research, as the GC content is assumed to substantially change only at intervals of 300,000 bases and more [Bernardi, 2000].

5.6 Conclusion

In this chapter, we have introduced a novel statistical test for assessing the significance of event frequencies in subsequences when using a sliding window. The test provides strong control of the family-wise error rate and takes into account the dependency structure of overlapping subsequences. It has been argued that exact p-values under the null hypothesis are difficult to compute, and we have introduced an easy-to-compute upper bound that can be used instead. We have provided empirical proof that the upper bound is sufficiently tight and that the test offers increased power compared to combining the binomial test with a generic post-hoc correction.

We have investigated the utility of the test on linguistic and biological sequences and found several novel and interesting patterns. We have illustrated that meaningful results can be obtained, and that the method remains sufficiently powerful even when testing a very large number of hypotheses. We conclude that the proposed method is simple, fast and powerful and that the method can produce meaningful results on various types of data.

6. Selecting the most informative set of window lengths

In this chapter, we consider the problem of *which granularities to use when looking for local patterns in an event sequence*. Event sequences often contain continuous variability at different levels. In other words, their properties and characteristics change at different rates, concurrently. For example, the sales of a product may slowly become more frequent over a period of several weeks, but there may be interesting variation within a week at the same time. To provide an accurate and robust *view* of such multi-level structural behaviour, one needs to determine the appropriate levels of granularity for analysing the underlying sequence.

We introduce the problem of finding the best set of window lengths for analysing discrete and continuous event sequences and we present suitable criteria for choosing a set of window lengths. We show that the corresponding optimisation problem is NP-hard in general, but that it can be approximated efficiently. We show that for certain criteria and data distributions the problem can also be solved exactly. We give examples of tasks to demonstrate the applicability of the problem and present experiments on both synthetic data and real data from several domains. We find that the method works well in practice and that the optimal sets of window lengths themselves can provide new insight into the data.

6.1 Introduction

Sequential data often contains slowly changing properties, mixed with faster changing properties. For example, the sales of a product may slowly become more frequent over a period of several weeks, but there may be interesting variation throughout a week at the same time. To provide an accurate and robust *view* of such multi-level structural behaviour, one needs to determine the appropriate levels of granularity for analysing the

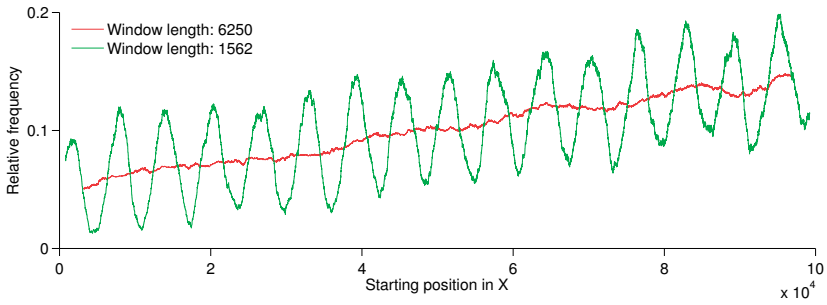

Figure 6.1. The frequency of an event over time, computed using two sliding windows of different lengths. The generative process for this sequence is described in Section 6.5.

underlying sequence.

Sliding windows are frequently employed in several sequence analysis tasks, such as mining frequent episodes [Mannila et al., 1997], finding biological or time series motifs [Chiu et al., 2003, Das and Dai, 2007], discovering poly-regions in biological sequences [Papapetrou et al., 2012], analysis of electroencephalogram (EEG) sequences [Sörnmo and Laguna, 2005], or in linguistic analysis of documents [Biber, 1988]. However, such methods are often parametrised by a user-defined window length and it can be unclear how to choose the most appropriate window length.

This problem can be avoided by either defining an appropriate objective function and using an optimisation algorithm to select the best window length, or by using all possible window lengths at the same time. The first approach has the limitation that a single window length may leave out important information. The second approach does not suffer from that problem, but provides a large amount of information, which may be too time consuming to analyse. We propose to use a small set of window lengths that together provide as much information as possible about the underlying data, with respect to a quantity of interest. We demonstrate that a good balance between informativeness and amount of information can be found.

Example. The frequency of an event in a sequence may show variation at different levels. Figure 6.1 shows an example of the relative frequency of an event over time, which is computed using two incremental sliding windows of lengths 1562 and 6250. The generative process for this sequence is described in Section 6.5. We observe that each window length tells us a different story about the event frequency. In other words, each window length entails a different view of the data. The longer window

suggests a smoothly increasing frequency throughout the sequence, while the shorter window captures a periodic behaviour of the event frequency.

Summary of contributions. We introduce the problem of finding the most informative set of window lengths for analysing event sequences, with respect to a statistic of interest. The statistic can be any quantity that is computed over a subsequence. We define suitable criteria and an efficient method for choosing a set of window lengths. We give examples that demonstrate the applicability of the problem to different domains.

We study optimal window lengths for synthetic data and show that analytical solutions can be obtained for certain statistics and data distributions, against which empirical results can be compared. We present experiments on data from several application domains: natural language texts, DNA sequences, and time series. We find that the scales of the occurrence patterns of various events (e.g., word types or DNA segments) vary significantly, and that the optimal scales can provide useful new insight into the data.

Outline. Related work is discussed in Section 6.2. The problem statement and the method are introduced in Sections 6.3 and 6.4. Experiments are presented in Section 6.5 and Section 6.6 contains the conclusions.

6.2 Related work

String and Text Mining. Sliding windows have been used extensively in string mining. Indexing methods for string matching based on *n-grams* [Li et al., 2007a], i.e., subsequences of length n, employ sliding windows of fixed or variable length to create dictionaries and speed-up approximate string search in large collections of texts. Determining the appropriate window length is always a challenge, as small window lengths result in higher recall but large index structures.

Looking at different linguistic dimensions of text results in extracting different views of the underlying text structure [Biber, 1988]. One way to quantify these views is by using sliding windows. Recently, an interactive text analysis tool[1] has been developed for exploring the effect of window length on three commonly studied linguistic measures: type-token ratio, proportion of hapax legomena, and average word length. However, the window length is user-defined.

Bioinformatics. Several sliding window approaches have been proposed

[1] http://www.uta.fi/sis/tauchi/virg/projects/dammoc/tve.html

for analysing large genomes and genetic associations. Existing methods can be categorised into two groups: fixed-length vs. variable-length sliding windows [Bourgain et al., 2000, Li et al., 2007b, Mathias et al., 2006, Papapetrou et al., 2012, Toivonen et al., 2000]. For the case of a fixed window length, it is difficult to determine the optimal window length per task while variable window length provide higher flexibility.

A variable window length framework for genetic association analysis employs principal component analysis to find the optimum window length [Tang et al., 2009]. Sliding windows have also been used for searching large biological sequences for poly-regions [Papapetrou et al., 2012], motifs [Das and Dai, 2007], and tandem repeats [Benson, 1999]. In each of these cases it is assumed that there exists only one optimum length and the solution is limited to the task of genetic association analysis.

Stream Mining. A typical task in stream mining is to detect and monitor frequent items or itemsets in an evolving stream, counted over sliding windows. We present a brief survey of the use of sliding windows in stream mining, although the overall setting is different from the problem studied in this chapter, and online learning is not considered here.

In the case of a fixed window length, the length of the window is set at the beginning and the data mining task is to discover recent trends in the data contained in the window [Demaine et al., 2002, Golab et al., 2003, Karp et al., 2003]. In the time-fading model [Lin et al., 2005] the full stream is taken into account in order to compute itemset frequencies but the frequencies are weighted by recency, i.e., recent transactions have a high weight compared to older transactions. The tilted-time window [Giannella et al., 2003] can be seen as a combination of different scales reflecting the alteration of the time scales of the windows over time.

In the landmark model, particular time periods are fixed while the landmark designates the start of the system until the current time [Jin and Agrawal, 2005, Karp et al., 2003]. Calders et al. [2008] introduced a frequency measure based on a variable window length, where the frequency of an item is defined as the maximal frequency over all windows until the most recent event in the stream. Several variants of these methods have been proposed for specific objectives.

Time Series. Enumerating frequently occurring patterns is a common data mining problem in time series. Such patterns are called *motifs* due to their close analogy to their discrete counterparts in computational biology. Efficient motif discovery algorithms have been proposed, based on sliding

windows, for summarising and visualising massive time series databases [Chiu et al., 2003, Mueen et al., 2009]. Related, but different in nature is the problem of scale-space decomposition of time series [Vespier et al., 2012], which aims at defining several frequency bands that correspond to the components of a signal.

Papadimitriou and Yu [2006] proposed a method for discovering locally optimal patterns in time series at multiple scales along with a criterion for choosing the best window lengths. However, this is a local heuristic and applies only to continuous data. As such, the overall setting and objectives in this chapter are substantially different.

Summary. Sliding windows have been widely used in many application domains that involve discrete or continuous sequences. However, window lengths are chosen either empirically or they are optimised for the task at hand. To the best of our knowledge, no earlier work has proposed a principled method for choosing the set of appropriate window lengths that optimally summarise the data for a given statistic and data mining task.

6.3 Problem statement

The basic notation for event sequences is presented in Section 2.1. We assume that a user is interested in analysing an event sequence or a database of event sequences using a *sliding window*, i.e., fixed-length subsequences are considered by sliding a *window* across the sequence. The *step size* determines by how many indices the sequence is moved at each step. For example, if the length of the window is 10 and the step size is 5, then we consider subsequences $S_{1,10}, S_{6,10}, S_{11,10}$, etc.

Furthermore, we assume that the relevant "information" contained in a subsequence $S_{i,m}$ is quantified by a *statistic* $f(S_{i,m}) : L^m \to \mathbb{R}$. Examples of possible statistics f are given below, but in principle f can be any function with a real-valued output. The problem considered here is the following: a user may be interested in analysing an event sequence with a large set of different window lengths, but due to cognitive limitations that would require too much time. The question that arises is "if I can use at most k window lengths, which window lengths should I choose to learn as much as possible about all the window lengths that I am interested in?".

In other words, the problem is how to select a set of k granularity levels that is most informative with respect to predicting the statistic f at all other granularity levels. Each level of granularity is described by a win-

dow length. Hence, our aim is to determine the set of k window lengths that simultaneously best describe the structure of an event sequence with respect to the statistic f at all window lengths.

The problem of capturing several different levels of structure with respect to f is translated into an optimisation problem as follows. Depending on the task at hand different objective functions may be considered. We propose an objective function that finds both a set of k window lengths and the parameters of a regression function, such that, at each position i in S, if we are given the value of f at those k window lengths, we can estimate f for all other window lengths at position i as accurately as possible.

Let $\Omega = \{\omega_1, \ldots, \omega_m\}$ be the set of m window lengths that a user is interested in, and let $\omega_{max} = \max_{i \in \{1,\ldots,m\}} \omega_i$ and $n^* = n - \omega_{max} + 1$. Denote $\Theta = (\theta_1, \ldots, \theta_k) \subseteq \Omega$ a vector of k window lengths and let $\bar{f}(S, \Theta, i) = (f(S_{i,\theta_1}), \ldots, f(S_{i,\theta_k}))$ be the vector of real numbers that corresponds to the values of f at position $i \in \{1, \ldots, n^*\}$ for the k window lengths in Θ.

Definition 6.1 (Reconstruction function)**.** *$g(\bar{f}(S, \Theta, i), \omega) : \mathbb{R}^k \times m \to \mathbb{R}$ is a function that, given the set of values $\bar{f}(S, \Theta, i)$ and a window length ω, estimates the value of f for window length ω; in other words, g is an estimator for $f(S_{i,\omega})$ and is referred to as the* reconstruction function*.*

An illustration of the mechanism of the reconstruction function is shown in Figure 6.2. The optimisation problem that corresponds to the window-length selection problem is the following:

Problem 6.1 (Select k-window lengths problem)**.** *Given an event sequence S, a statistic f, and a set of window lengths Ω, find a set of k window lengths $\Theta = \{\theta_1, \ldots, \theta_k\} \subseteq \Omega$ and a reconstruction function g that minimise*

$$\frac{1}{n^*} \sum_{i=1}^{n^*} \sum_{\omega \in \Omega} \big(f(S_{i,\omega}) - g(\bar{f}(S, \Theta, i), \omega)\big)^2.$$

The reconstruction function g can be any regression function. However, not specifying g would lead to a practically impossible optimisation task, as it is infeasible to explore the space of all possible regression models. Hence, we propose that, depending on the task at hand, the set of possible models should be restricted to obtain a tractable optimisation problem. For example, g can be restricted to the class of *nearest neighbour regressors* (see Section 6.4), in which case the optimisation problem is equivalent to the k-medoids clustering problem.

The idea here is that additional parameters used by the reconstruction function g that are learned during the optimisation process, i.e., those not

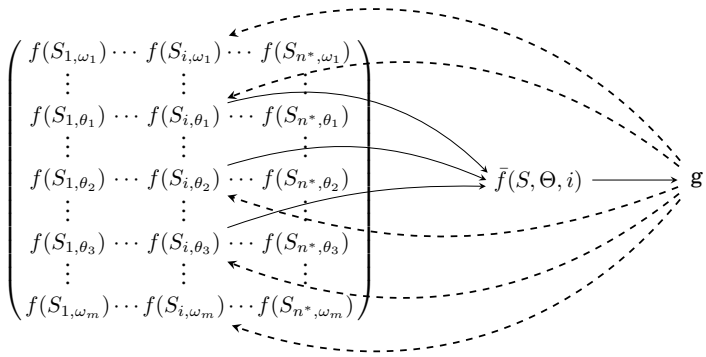

Figure 6.2. Illustration of the reconstruction function g. Each row corresponds to a window length in Ω and each column to a position in the event sequence S. Function g estimates the value of the statistic f for all window lengths at position i in S, based on the values of f for a small set window lengths $\{\theta_1, \ldots, \theta_k\}$, in this case $k = 3$.

in $\bar{f}(S, \Theta, i)$, are kept implicit and not shown to the user. Neither is g itself considered to be interesting. This means that g should be restricted to regression functions that are easy to comprehend by end-users. In this chapter, we consider only the nearest neighbour regressor, which we introduce in Section 6.4.

Examples of informative statistics. The following equations give examples of the statistic f which are used in the experiments. For example, f may be the relative frequency of a set of events $A \subseteq L$:

$$f(S_{i,\omega}) = \frac{\sigma_A\left(S_{i,\omega}\right)}{\omega} = \zeta_A\left(S_{i,\omega}\right). \tag{6.1}$$

Alternatively, f may be defined as the *hapax legomenon* ratio of a sequence, i.e.,

$$f(S_{i,\omega}) = \frac{\text{\#events occurring exactly once in } S_{i,\omega}}{\omega}. \tag{6.2}$$

For real valued data, the mean value of a subsequence can be used as a statistic, in which case f is defined as

$$f(S_{i,\omega}) = \frac{1}{\omega} \sum_{j=i}^{i+\omega-1} s_j. \tag{6.3}$$

The utility of these three definitions is illustrated in Section 6.5.

6.4 Methods

In this section, we introduce one possible approach to selecting the k most informative window lengths using the problem setting defined above. We

restrict the reconstruction function g to the class of k-partition nearest neighbour regressors. This restricted problem setting is defined below, in Section 6.4.1. In Section 6.4.2, we introduce an auxiliary data structure, called the *Window-Trace matrix*, and the optimisation algorithm is described in Section 6.4.3. In Section 6.4.4, we show that for certain data structures, the optimisation problem can be solved analytically.

6.4.1 Partition-based regression

The optimisation problem is made tractable by restricting the reconstruction function g to the following class of regression functions.

Definition 6.2 (k-partition nearest neighbour regressor)**.** *Let S be an event sequence, i an index in S, $\Theta = (\theta_1, \ldots, \theta_k)$ the vector of chosen window lengths, and ω any window length in Ω. A* k-partition nearest neighbour regressor *is a* reconstruction function $g(\bar{f}(S, \Theta, i), \omega)$ *that is based on a partitioning of the set of window lengths Ω into k non-overlapping clusters $\Psi_1, \ldots, \Psi_k$. Each of the window lengths in Θ is used as the representative for the cluster with the same index. This implicitly requires that $\theta_j \in \Psi_j$ for all $j \in \{1, \ldots, k\}$. Let Ψ_j be the cluster that contains ω. The reconstruction function g simply returns the value of the representative window length for cluster Ψ_j: $g(\bar{f}(S, \Theta, i), \omega) = f(S_{i,\theta_j})$.*

By restricting g to the class of k-partition nearest neighbour regressors, Problem 6.1 becomes equivalent to partitioning the set of all window lengths Ω into k clusters, and selecting for each cluster one representative window length, such that the expected squared error is minimised. This problem is equivalent to the k-medoids clustering problem.

Since the statistic f is unconstrained and the k-medoids clustering problem is NP-hard [Aloise et al., 2009], this optimisation problem is also NP-hard in general. Since the number of window lengths $|\Omega|$ may be very large, we do not consider any exponential-time exact algorithms. There exist several optimisation algorithms that give good approximations; we use a modified version of the clustering large applications (`Clara`) algorithm [Kaufman and Rousseeuw, 1990], see Section 6.4.3.

6.4.2 The window-trace matrix

To solve Problem 6.1, we use an auxiliary matrix called the *window-trace (W-T) matrix*. This matrix stores the values of statistic f for a set of indices I in the event sequence and for all window lengths in Ω. More specif-

ically, let S be the input sequence and f the statistic at hand. Then the W-T matrix $\mathcal{T}$ contains all values of $f(S_{i,\omega})$ for all window lengths $\omega \in \Omega$ and all indices $i \in I$. $\mathcal{T}$ is given by

$$\mathcal{T}_{j,i} = f(S_{i,\omega_j}). \tag{6.4}$$

The most comprehensive representation is obtained by choosing $I = \{1, \ldots, n^*\}$, i.e., all indices for which f can be computed. However, for computational efficiency, we restrict the set of indices I to $|I| = N$ indices, which are sampled uniformly at random with replacement from $\{1, \ldots, n^*\}$. Furthermore, we use $\mathcal{T}_{j,*}$ to denote the row of $\mathcal{T}$ corresponding to window length ω_j.

6.4.3 Optimisation algorithm

We solve the optimisation problem with a modified version of the clustering large applications (`Clara`) algorithm [Kaufman and Rousseeuw, 1990], a well-known algorithm to efficiently solve the k-medoids problem. We make the following modification to the algorithm to increase the quality of the solution.

Arthur and Vassilvitskii [2007] studied the effects of seeding—the process of choosing the initial representatives for each cluster—for the k-means algorithm, and present a method of "careful seeding" that leads to a provable approximation ratio on the solution. Their improved algorithm is known as k-means++. The effects of seeding for the `Clara` algorithm appears not to have been studied before.

We propose to change the initial seeding in the `Clara` algorithm and in the partitioning around medoids (`PAM`) subroutine to the method described in Arthur and Vassilvitskii [2007]. The k-means++ seeding method produces a k-medoids solution, so it can be applied directly. We expect that this careful seeding substantially increases the quality of the result. We refer to the improved variants as `Clara++` and `PAM++`. Pseudocode for the methods is given in Algorithms 6.1 and 6.2.

The parameters r and s are related to a trade-off between quality and computational complexity; they define the number of repetitions and number of samples included in the `PAM` subroutine. In the original `Clara` algorithm these are not considered to be parameters and have default values of $r = 5$ and $s = 40 + 2k$ [Kaufman and Rousseeuw, 1990]. However, it is not obvious that these defaults suffice to yield good results, and we review the effects of these parameters in Section 6.5.1.

Algorithm 6.1 Clara++($\mathcal{T}$, k, r, s)

$\Theta_1 = \text{uniform}([1, \ldots, n])$ {Pick a number between 1 and n uniformly at random, n is the number of rows in $\mathcal{T}$}
for i = 2 to k **do**
  $\Theta_i = \text{rand}([1, \ldots, n])$ {Pick a number between 1 and n at random with probability proportional to the distance to the closest medoid in Θ}
end for
$cost^* = \infty$
for i = 1 to r **do**
  $S = \Theta \cup \text{uniform}(\{1, \ldots, n\} \setminus \Theta, s - k)$ {Assign S a set that contains Θ, the best set of medoids currently known, and pick $s - k$ other row indices uniformly at random}
  $\mathcal{T}^S = \begin{bmatrix} \mathcal{T}_{S_1,*} \\ \vdots \\ \mathcal{T}_{S_s,*} \end{bmatrix}$ {Select the s rows of $\mathcal{T}$ whose index is in S}
  $\Theta = \text{PAM++}(\mathcal{T}^S, k)$ {Compute PAM++ solution on sample}
  $cost = \text{computeClusteringCost}(\mathcal{T}, \Theta)$ {Compute cost for full matrix}
  if $cost < cost^*$ **then**
    $\Theta^* = \Theta$
    $cost^* = cost$
  end if
end for
return Θ^*

Computational Complexity. Let N be the number of columns of $\mathcal{T}$, i.e., the number of samples, and let m be the number of rows of $\mathcal{T}$: $m = |\Omega|$. The memory required to store the Window-Trace matrix $\mathcal{T}$ is $\mathcal{O}(m \cdot N)$, and, assuming that the complexity of computing the statistic $f(S_{i,\omega})$ is constant, the computational complexity to produce $\mathcal{T}$ is also $\mathcal{O}(m \cdot N)$.

Clara++ consists of the initial selection of k medoids and then executing the PAM++ subroutine r times, each on a data sample of size s, plus computing the cost of the clustering on each iteration. The initialisation of the k medoids has a computational complexity of $\mathcal{O}(k \cdot m \cdot N)$, because for each medoid the distance to all other points has to be computed. Let t denote the number of iterations required for convergence of the PAM++ subroutine. Since computing the full distance matrix takes $\mathcal{O}(s^2 \cdot N)$ steps, the computational complexity of the PAM++ subroutine is $\mathcal{O}(s^2 \cdot N + t \cdot k \cdot s + t \cdot s^2)$. As $k < s$, this simplifies to $\mathcal{O}(s^2 \cdot (N + t))$.

Algorithm 6.2 PAM++($\mathcal{T}$, k)

$\Theta_1 = \text{uniform}([1, \ldots, n])$ {Pick a number between 1 and n uniformly at random, n is the number of rows in $\mathcal{T}$}
for i = 2 to k **do**
  $\Theta_i = \text{rand}([1, \ldots, n])$ {Pick a number between 1 and n at random with probability proportional to the distance to the closest medoid in Θ}
end for
Θ_{old} = {}
while $\Theta_{old} \neq \Theta$ **do**
  $\Theta_{old} = \Theta$
  for i = 1 to n **do**
    $L_i = \arg\min_{j \in [1,\ldots,k]}(\|\mathcal{T}_{i*} - \mathcal{T}_{\Theta_j *}\|^2)$ {Label each point with nearest medoid}
  end for
  for i = 1 to k **do**
    $C = \{x \mid L_x = i\}$ {Find the set of points in cluster i}
    $\Theta_i = argmin_{x \in C} \sum_{y \in C} \|T_{x*} - T_{y*}\|^2$ {Pick best medoid for cluster i}
  end for
end while
return Θ

Computing the cost of a clustering has complexity $\mathcal{O}(k \cdot m \cdot N)$, thus the total computational cost of `Clara++` is $\mathcal{O}(r \cdot s^2 \cdot (N + t) + r \cdot k \cdot m \cdot N)$. That is, the cost is linear in the number of window lengths m, in the number of data samples N, and in the number of repetitions r, but quadratic in s, the number of samples considered in an iteration of `PAM++`.

6.4.4 Analytical solutions

For certain statistics and data distributions, it is possible to derive the solution, or at least the function for the distance between two window lengths, exactly. In this section, we present an analytical solution for the case where the statistic is the frequency of an event and the event sequence comes from a Bernoulli process. From the result follows that for a Bernoulli process, the window lengths (i.e., the clustering) is independent of the frequency of an event.

Preliminaries. Let $(X_1, \ldots, X_n)$ be a sequence of Bernoulli random variables with common parameter p, i.e., $X_i \in \{0, 1\}, \Pr(\{X_i = 1\}) = p$, for all $i \in \{1, \ldots, n\}$. The random variables could, for example, denote the

occurrences of an event. Similar to the notation for event sequences, we use $X_{i,\omega}$ to denote the subsequence of length ω starting at position i, $(X_i, \ldots, X_{i+\omega-1})$. Let the statistic f be the relative frequency of ones:

$$f(X_{i,\omega}) = \frac{1}{\omega} \sum_{j=i}^{i+\omega-1} X_j. \tag{6.5}$$

The selection of an optimal set of window lengths is based on the squared error between predictions made using those window lengths (Problem 6.1). Under the constraint of using a k-partition nearest neighbour regressor, the predictions correspond to the value of the nearest window length (Section 6.4.1). Thus, to select the optimal window lengths, we have to compute the distance (squared error) between all pairs of window lengths. We find that the distance between window lengths is as follows.

Theorem 6.1. *For the statistic and generative process described above, the expected distance between two window lengths γ and ω, with $\gamma < \omega$, is*

$$\mathrm{E}\left[d(\omega, \gamma)\right] = \frac{\omega - \gamma}{\omega\gamma} p(1-p).$$

Proof. The expected distance between two window lengths γ and ω is

$$\mathrm{E}\left[d(\omega, \gamma)\right] = \mathrm{E}\left[\frac{1}{n-m+1} \sum_{i=1}^{n-m+1} \left(f(X_{i,\gamma}) - f(X_{i,\omega})\right)^2\right].$$

Since $X_1, \ldots, X_n$ are i.i.d. random variables, this simplifies to

$$\mathrm{E}\left[d(\omega, \gamma)\right] = \mathrm{E}\left[\left(f(X_{1,\gamma}) - f(X_{1,\omega})\right)^2\right].$$

Assuming without loss of generality that $\gamma < \omega$, we find that

$$\begin{aligned} f(X_{i,\omega}) &= \frac{1}{\omega} \sum_{j=1}^{i+\omega-1} X_j \\ &= \frac{1}{\omega} \sum_{j=1}^{i+\gamma-1} X_j + \frac{1}{\omega} \sum_{j=1+\gamma}^{i+\omega-1} X_j \\ &= \frac{\gamma}{\omega} f(X_{i,\gamma}) + \frac{\omega - \gamma}{\omega} f(X_{i+\gamma,\omega-\gamma}). \end{aligned}$$

Thus we can rewrite the expected distance as

$$\begin{aligned} &\mathrm{E}\left[d(\omega, \gamma)\right] \\ &= \mathrm{E}\left[\left(f(X_{1,\gamma}) - \frac{\gamma}{\omega} f(X_{1,\gamma}) - \frac{\omega - \gamma}{\omega} f(X_{1+\gamma,\omega-\gamma})\right)^2\right] \\ &= \mathrm{E}\left[\left(\frac{\omega - \gamma}{\omega}\right)^2 \left(f(X_{1,\gamma}) - f(X_{1+\gamma,\omega-\gamma})\right)^2\right] \\ &= \left(\frac{\omega - \gamma}{\omega}\right)^2 \mathrm{E}\left[\left(f(X_{1,\gamma}) - f(X_{1+\gamma,\omega-\gamma})\right)^2\right] \\ &= \left(\frac{\omega - \gamma}{\omega}\right)^2 \mathrm{E}\left[f(X_{1,\gamma})^2\right] + \mathrm{E}\left[f(X_{1+\gamma,\omega-\gamma})^2\right] - 2\,\mathrm{E}\left[f(X_{1,\gamma}) f(X_{1+\gamma,\omega-\gamma})\right]. \end{aligned}$$

These three expectations are

$$\begin{aligned}
\mathrm{E}\left[f(X_{1,\gamma})^2\right] &= \frac{p(1-p)}{\gamma} + p^2, \\
\mathrm{E}\left[f(X_{1+\gamma,\omega-\gamma})^2\right] &= \frac{p(1-p)}{\omega-\gamma} + p^2, \text{ and} \\
\mathrm{E}\left[f(X_{1,\gamma})f(X_{1+\gamma,\omega-\gamma})\right] &= p^2.
\end{aligned}$$

For brevity, we skip the derivation for these three expectations. They can be derived, for example, using the fact that the variance of a binomial distribution is $\mathrm{Var}\left[Bin(n,p)\right] = \mathrm{E}\left[Bin(n,p)^2\right] - \mathrm{E}\left[Bin(n,p)\right]^2 = np(1-p)$, and its expectation is $\mathrm{E}\left[Bin(n,p)\right] = np$.

By writing out the expected distance we find that

$$\begin{aligned}
\mathrm{E}\left[d(\omega,\gamma)\right] &= \left(\frac{\omega-\gamma}{\omega}\right)^2 \frac{p(1-p)}{\gamma} + p^2 + \frac{p(1-p)}{\omega-\gamma} + p^2 - 2p^2 \\
&= \left(\frac{\omega-\gamma}{\omega}\right)^2 \left(\frac{1}{\gamma} + \frac{1}{\omega-\gamma}\right) p(1-p) \\
&= \frac{(\omega-\gamma)^2}{\omega^2} \frac{\omega-\gamma+\gamma}{\gamma(\omega-\gamma)} p(1-p) \\
&= \frac{\omega-\gamma}{\omega\gamma} p(1-p).
\end{aligned}$$

□

There is no interaction between the window lengths γ, ω and the event probability p, which implies that all distances relative to each other are independent of p. Thus, for this specific statistic and data distribution, the optimal window lengths are unaffected by the event frequency, and depend only on the set of window lengths Ω.

6.5 Experiments

6.5.1 Evaluation on synthetic data

In any data mining task, it is important to be able to evaluate the statistical significance of a result. We studied what to expect regarding optimal sets of window lengths for one type of random process in Section 6.4.4, but we do not know to what extent there is variation in the solution given by the `Clara++` algorithm for more complex data, while that information is essential to determine the significance of a result. To provide a baseline for the experiments on real data (Section 6.5.2), we designed four experiments using randomly generated data. Randomly generated data is useful here because the precise properties of the data are then known.

Bernoulli process with fixed rate

We are interested in how much the set of window lengths given by Clara++ varies in the case when the data is generated by a fixed rate Bernoulli process. We used Algorithm 6.3 to generate such random sequences. The algorithm has two parameters, n and p, which are the length of the event sequence and the occurrence probability of the event.

Algorithm 6.3 Simulate a fixed-rate Bernoulli process SIM1(n, p)

for i = 1 to n **do**
  $X(i)$ = Bernoulli(p)
end for

Experiment 1. Since Clara++ is a non-deterministic algorithm, the output may vary, even with the same input sequence. To investigate this, we tested the stability of the solution in terms of the optimal window lengths for a single sequence generated by Algorithm 6.3 with parameters $n = 1{,}000$ and $p = 0.1$.

We varied the number of repetitions from 5 to 40 (doubling the value each time) and the number of samples from 40 to 320 (also by doubling the value each time). The number of clusters was varied from 1 to 4 and Ω contained all window lengths from 1 to 500. As the statistic we used the relative frequency of the event (Equation 6.1). We repeated the experiment 100 times for each combination of parameters. For comparison, we also computed the solution of the PAM++ algorithm.

The results are presented in Figure 6.3. We find that the variation with the default parameter settings is quite large. For example, the top left figure ($k = 4$) shows that the smallest window length is sometimes larger than the second largest window length in another run. The variation is greatly reduced when the number of repetitions increases, while increasing the number of samples has hardly any effect. The bottom right figure shows that the set of window lengths is quite stable when 40 repetitions are used. As the computational complexity is linear in the number of repetitions, it is no problem to use 40 repetitions instead of 5, while this greatly improves the probability of obtaining a close to optimal result.

Experiment 2. Several data sets, even if they are from the same generative process, may yield quite different results. We designed the second experiment to test the stability of the solutions given by Clara++ for different data sets that have the same properties. We generated one data set with parameters $n = 1{,}000$, $p = 0.1$, and then produced 100 varia-

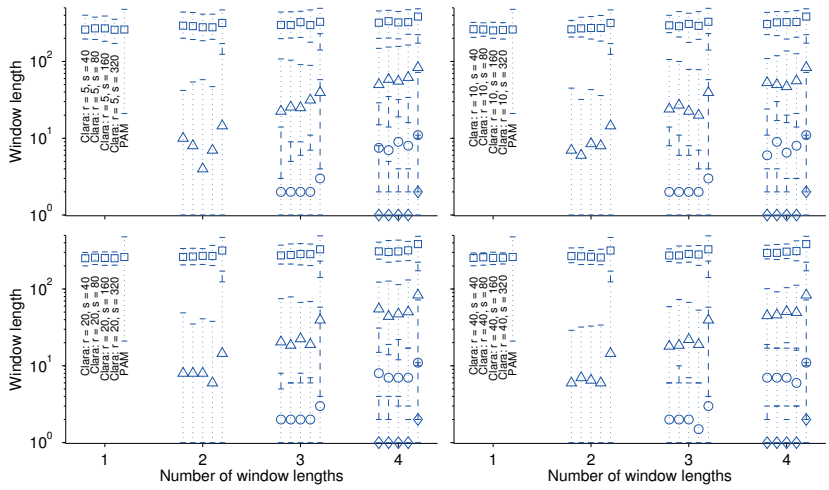

Figure 6.3. Optimal sets of window lengths from `Clara++` on a random event sequence, using varying number of repetitions ($r = 5$, 10, 20, 50, one value per figure) and number of samples ($s = 40$, 80, 160, 320, adjacent bars in each figure). Squares, triangles, circles and diamonds represent the medians for the first, second, third and fourth window lengths, the dotted lines represent 90 % confidence intervals and dashed lines denote that the confidence intervals for the window lengths are overlapping. For comparison, the variability for the `PAM++` algorithm is also shown for each number of window lengths.

tions by randomly permuting the indices of the sequence. We computed the optimal window lengths for $k = 1, \ldots, 4$ on each data set, using 40 repetitions and 40 samples as parameters for `Clara++`. The previous experiment showed that these parameter values are good choices, and the other parameters were kept the same as in the previous experiment.

The results are presented in Figure 6.4. There is much more variation than in the previous experiment, which can be explained by the fact that the input sequences are slightly different in each repetition. The observed variance in the figure can be used in future experiments to draw conclusions with respect to the significance of differences in sets of window lengths obtained for various events or data sets.

Bernoulli process with variable rate

In the previous experiments, we kept the frequency of the events constant over time, which leads to the sequence having structure only on a single scale. To test the ability of our method for finding the true underlying scales at which the data is structured, we designed an algorithm to simulate a Bernoulli process with variable rate.

The full process is described in Algorithm 6.4. The first component of the variable rate is based on a slow increase of the event frequency over time,

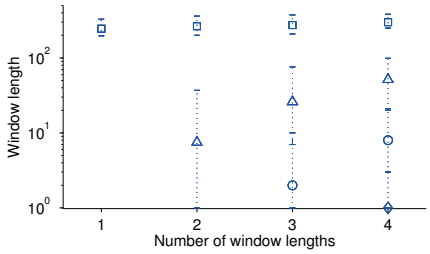

Figure 6.4. Optimal sets of window lengths from `Clara++` over 100 data sets with the same properties. Squares, triangles, circles and diamonds represent the medians for the first, second, third and fourth window lengths, the dotted lines represent 90 % confidence intervals and dashed lines denote that the confidence intervals for the window lengths are overlapping.

Algorithm 6.4 Simulate a variable-rate Bernoulli process SIM2(n, p, c)

for i = 1 to n **do**

$t_1 = 0.5 + (i-1)/(n-1)$; // Multiplier for scale 1: [0.5–1.5]

$t_2 = 0.5 \cdot \sin(c \cdot 2 \cdot \pi \cdot (i-1)/(n-1))$; // Multiplier for scale 2: [−0.5–0.5]

$X(i) = Bernoulli(p \cdot (t_1 + t_2))$

end for

which ranges from $0.5 \cdot p$ at the start to $1.5 \cdot p$ at the end of the sequence S. The second component consists of the event frequency going up and down rhythmically, based on a sine wave with peak amplitude 0.5 and mean 0.

Both components are added together to give a variable event frequency, which is multiplied by the parameter p. The extra parameter, c, decides the periodicity of the sine wave, and thus the second scale. We generated a sequence with parameters $n = 100{,}000$, $p = 0.1$ and $c = 16$. The sequence has 10,009 events and has also been used to produce Figure 6.1.

Experiment 3. As discussed in Section 6.4.2, we estimate the optimal set of window lengths for a sequence using a W-T matrix $\mathcal{T}$ based on samples from the data. We designed the third experiment to investigate empirically how many samples $\mathcal{T}$ should be based on to obtain a solution close to the solution that was obtained on the full matrix, i.e., the matrix $\mathcal{T}$ that covers the whole input sequence. We varied the number of samples from 1 to 16,384 using powers of 2 and computed the solution 100 times for each sample size to assess the variance. We chose Ω to include window lengths from 1 up to $\lfloor n/c \rfloor = 6{,}250$ (which is the scale of the second component in the data) and the number of outputs as $k = 3$.

Figure 6.5 illustrates the results. We find that the solutions are remarkably robust: the solutions using only 8 samples are already quite accurate approximations and from 64 samples and up, the solutions are practically

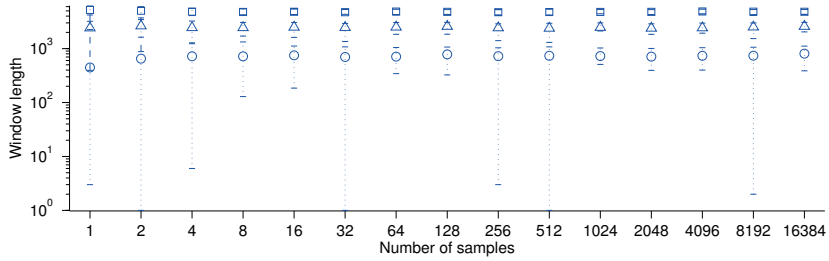

Figure 6.5. Optimal sets of window lengths from `Clara++` on a sequence obtained from simulating a Bernoulli process with rate that varies over time, using various numbers of samples to construct the Window-Trace matrix $\mathcal{T}$. Squares, triangles, and circles represent the medians for the first, second, and third window lengths, the dotted lines represent 90 % confidence intervals and dashed lines denote that the confidence intervals for the window lengths are overlapping.

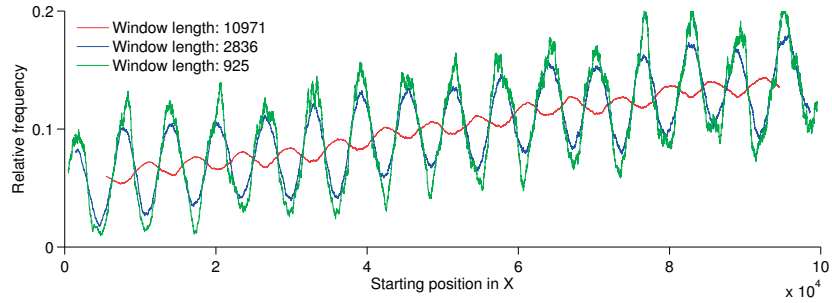

Figure 6.6. The frequency of an event over time, computed using three sliding windows of different lengths given by `Clara++`. The sequence was obtained by simulating a Bernoulli process with variable rate.

equivalent. Thus, for simple data sets like this, a Window-Trace matrix based on 64 positions in S is sufficient.

Experiment 4. Our fourth experiment aims to test if the two scales that are present in the synthetic sequence can indeed be retrieved. To prevent making it too easy for the algorithm, we chose Ω to contain all window lengths from 1 to 20,000 and based the Window-Trace matrix $\mathcal{T}$ on 1,000 indices in S. In a typical setting, we would not know how many scales a data set has. A higher k always provides more information, thus choosing k too high is better than too low. For exploratory purposes, we used $k = 3$. Figure 6.6 presents the results. We find that the variable trend in the data can be clearly identified and that the slow trend is reasonably visible.

Choosing proper parameter values

Based on the previous experiments, the following conclusions regarding the parameter choices can be drawn:

– The accuracy of the solution can be increased by using more repetitions in the Clara++ algorithm. At least 40 repetitions is recommended, instead of the default value of 5. More complex data and a larger set of window lengths possibly require more repetitions.

– Increasing the number of samples has only a minor effect, while that increases the computational complexity quadratically.

– The number of samples in the Window-Trace matrix can be small; 64 samples is sufficient for a Bernoulli sequence.

– The uncertainty present in the data is larger than the uncertainty in the solution, which means that the algorithm is prone to overlearning.

6.5.2 Evaluation on real data

We evaluated the practical utility of the method in four experiments on real data. First, we study the frequency over time of several words of varying type throughout the novel *Pride and Prejudice*. Secondly, we study what window lengths would be appropriate for analysing the hapax legomenon ratio in subsequences throughout texts from various genres. Thirdly, we examine the frequency over time of nucleotides and dinucleotides in two reference genomes from the NCBI repository, and lastly, we identify the appropriate window lengths for analysing a time series with multi-scale structure.

Optimal window lengths for several words

As discussed in Chapters 4 and 5, *burstiness* [Katz, 1996] and *dispersion* [Gries, 2008] of words in natural language corpora have become important concepts in research in linguistics, natural language processing and text mining. In Section 6.4.4, we showed that the frequency of an event does not affect the optimal set of window lengths, if there is no other structure in the sequence. Thus it would be interesting to know if the optimal set of window lengths does depend on the burstiness of an event in a sequence.

To test this, we conducted the following experiment. The data that we used is the popular novel *Pride and Prejudice* by Jane Austen, see Section 3.3. The length of the novel is approximately 120,000 words. We selected the 30 most and least bursty words that occur at least 100 times, using the

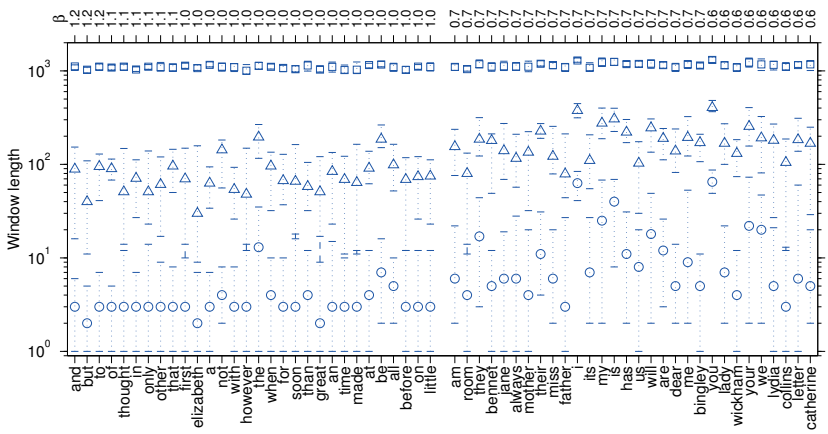

Figure 6.7. Optimal sets of window lengths for analysing the frequency over time of 60 words in the novel *Pride and Prejudice*, for $k = 3$. Squares, triangles, and circles represent the medians for the first, second, and third window lengths, the dotted lines represent 90 % confidence intervals and dashed lines denote that the confidence intervals for the window lengths are overlapping. The words are ordered by burstiness, i.e., the Weibull β parameter.

Weibull distribution to measure the burstiness of words, see Section 2.3. To study the effect of burstiness on the optimal sets of window lengths, we set the parameters to $k = 3$ and $\Omega = \{1, \ldots, 2000\}$.

The result is shown in Figure 6.7. The value for the Weibull β parameters are given at the top of the figure. We observe a clear trend: for the two smaller window lengths, the window lengths are significantly longer for bursty words than for non-bursty words, although there is considerable variation within the groups. The effect is strongest for the words *I* and *you*, which are the most frequent bursty words. Although not so obvious in the figure, the average and median window lengths for the longest windows are also higher for bursty words than for non-bursty words (mean/median non-bursty vs. bursty: 1101/1099 vs. 1164/1155, standard deviation non-bursty vs. bursty: 76 vs. 96).

That bursty words give longer window lengths may be due to the fact that bursty words exhibit a larger scale structure (bursts and intervals between bursts) than the more uniformly distributed words. The variation over individual words inside the groups is likely due to an interaction between the burstiness and the frequency of words and because the Weibull β does not capture exactly the same *burstiness* as the window length selection method. In Chapter 5, we showed that texts contain local structure that is not captured by the Weibull β measure.

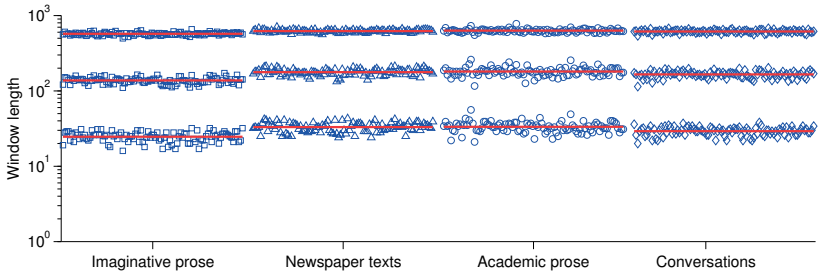

Figure 6.8. Optimal sets of window lengths for analysing the Hapax legomenon ratio over time for 400 texts from the British National Corpus, for various genres, using $k = 3$. Each point corresponds to a window length selected for that book, and red lines present the average per genre.

Hapax legomenon ratio in several genres

The genre of a text has an important effect on its structure, which can be measured in terms of several linguistic features, for example the hapax legomenon ratio of texts [Biber, 1988]. We designed the following experiment to test if the optimal set of window lengths differs significantly over texts from different genres. We sampled 100 texts from the British National Corpus [2007] for each of the main genres in the corpus: conversation, imaginative fiction, academic prose and newspaper texts. The statistic used in this experiment is the hapax legomenon ratio (Equation 6.2). We used $\Omega = \{1, \ldots, 1{,}000\}$ and $k = 3$.

The result is shown in Figure 6.8. Although the set of window lengths varies over the texts within each genre, there are significant differences: imaginative prose and conversations each seem to have a different structure than the texts from other three genres, while newspaper texts and academic prose are similar with respect to hapax legomenon ratios. This suggests that the scale structure of imaginative prose is more uniform than for other genres. Perhaps the difference is partly an artefact of the corpus structure, as the texts in the imaginative prose class are long coherent stories, while the 'texts' in the other classes are collections of articles, topics and conversations, rather than single documents.

Frequency of nucleotides throughout DNA sequences

Studies in biology and bioinformatics have shown that DNA chains consists of a number of important, known functional regions, at both large and small scales, which contain a high occurrence of one or more nucleotides [Bernardi, 2000]. Examples of such regions include: *isochores*, which correspond to long regions of genomic sequences that are specifically GC-rich or GC-poor and correlate with gene density, and *CpG is-*

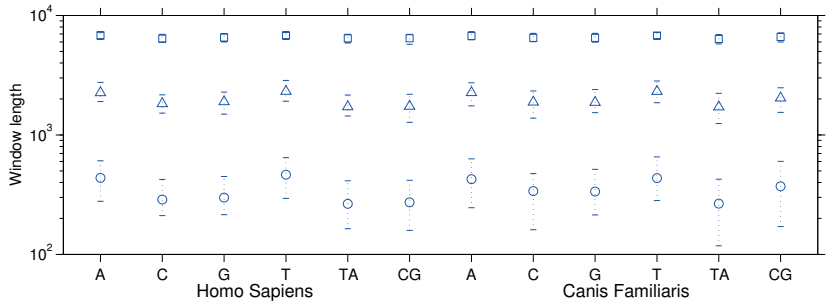

Figure 6.9. Optimal sets of window lengths for analysing the frequency over time of nucleotides and dinucleotides in Homo sapiens chromosome 1 and Canis lupus familiaris chromosome 1, for $k = 5$.

lands, which correspond to regions of several hundred nucleotides that are rich in the dinucleotide CpG, whose presence in the genome has been associated with the location of genes. We tested the proposed method for its utility towards selecting the appropriate scales at which to study (di)nucleotide frequencies in DNA, using the following experiment.

We studied Chromosome 1 of two organisms: *Homo sapiens* (human) and *Canis lupus familiaris* (dog), which have lengths 225 and 122 million nucleotides, respectively, see Section 3.4 for more background. We considered six event types: the four nucleotides A, C, G, and T, as well as dinucleotides TA and CG. We computed the solutions for $k = 3$ and window lengths up to 10,000. The statistic used in the experiment was the relative event frequency, and the W-T matrix contained 1,000 samples.

Figure 6.9 shows a comparison between the best window lengths found by the proposed algorithm for the two organisms. It can be observed that the four single nucleotides as well as the two dinucleotides exhibit similar behaviour for both organisms. This may be explained by the high genomic structural similarity between humans and dogs [Kirkness et al., 2003]. Nonetheless, the nucleotides C and G and both dinucleotides behave substantially different from the nucleotides A and T. From Section 6.4.4 we know that the frequency of an event does not affect the clustering, thus the differences are certainly structural, and not merely an effect of A and T being more frequent.

Figure 6.10 illustrates the frequency over time over the first 200,000 bases for chromosome 1 of Homo sapiens, for all four nucleotides and the two dinucleotides, using the optimal window lengths. We find that the different window lengths give somewhat different views of the data. As expected, the exact locations of bursts are identified most accurately by the

shortest window length. However, the significance of each burst is seen more clearly from the line corresponding to the longest window length, since that line takes a fairly constant value throughout most of the sequence. Thus, there is value in using multiple window lengths, although in this case two window lengths may be sufficient.

Smoothing of time series

An example of a time series with multi-scale structure comes from the *Infrawatch project* [Knobbe et al., 2010, Vespier et al., 2012], see Section 3.5. The data contains structure at three time scales: a high frequency component generated by individual cars and trucks passing on the bridge, a medium frequency component generated by traffic jams, and a low frequency component generated by weather effects (e.g., temperature).

We tested the proposed method on this data with the following parameters. Since the scale space is potentially very large and the frequencies below 1 Hz (window length 10) are not interesting, we constructed Ω by using powers of $\sqrt{2}$, starting at window length 10: $\Omega = \{10, 14, 20, \ldots, 231705\}$. Furthermore, we used $N = 1{,}000$ and $k = 3$.

The result is presented in Figure 6.11. It is difficult to say whether the three window lengths correspond directly to the three time scales that are present in the data, because the window lengths correspond to different views and not to frequency bands, as studied by Vespier et al. [2012]. Still, we find that the three window lengths give different views of the data, each of which represents a different time scale. The substantial difference between the window lengths is most clear in the zoomed-in figure.

6.6 Conclusion

We have introduced the novel problem of identifying a set of window lengths that contain the maximal amount of information in the data, and we have presented a generally applicable optimisation problem that users could employ. We have shown that the optimisation problem is NP-hard in general, but that it can be approximated efficiently algorithmically and solved analytically for certain simple statistics and data distributions. We studied the performance of the proposed optimisation algorithm, as well as the results for several statistics on both synthetic data and real data.

We have illustrated that the analytical and empirical results on synthetic data are useful as a baseline for practical use. We have shown that

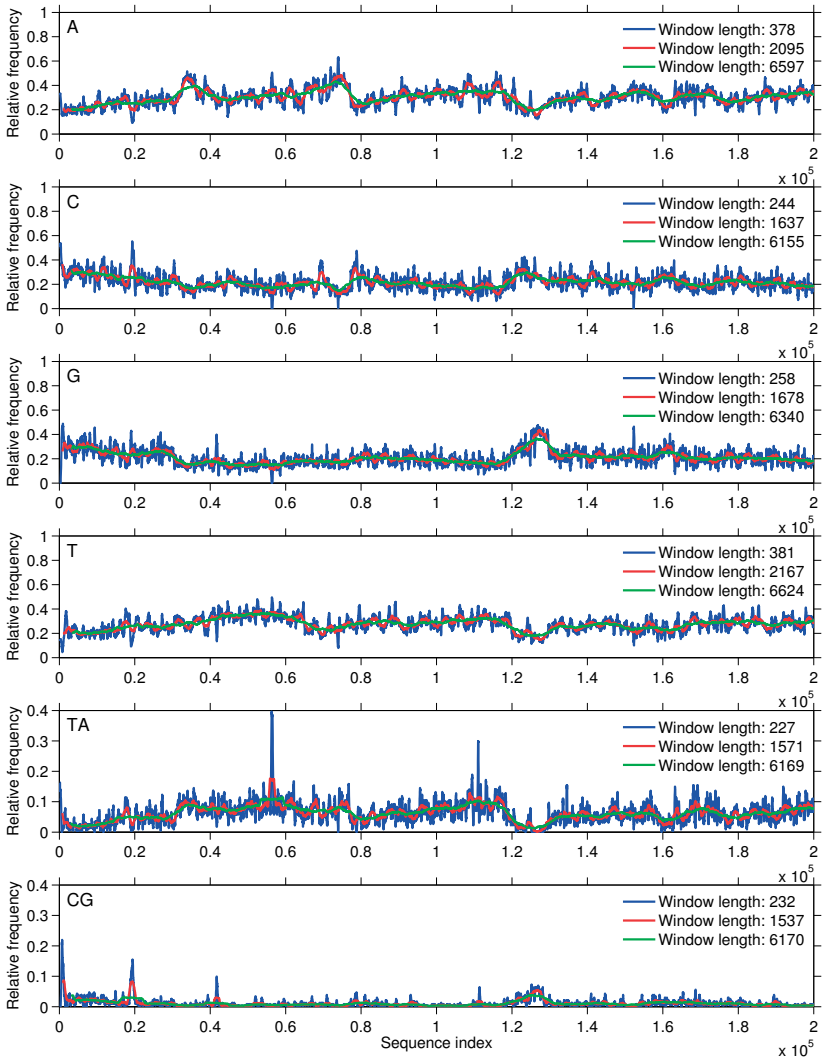

Figure 6.10. Frequency of the studied (di)nucleotides over the first 200,000 bases for chromosome 1 of Homo sapiens, using the best window lengths found by the proposed method.

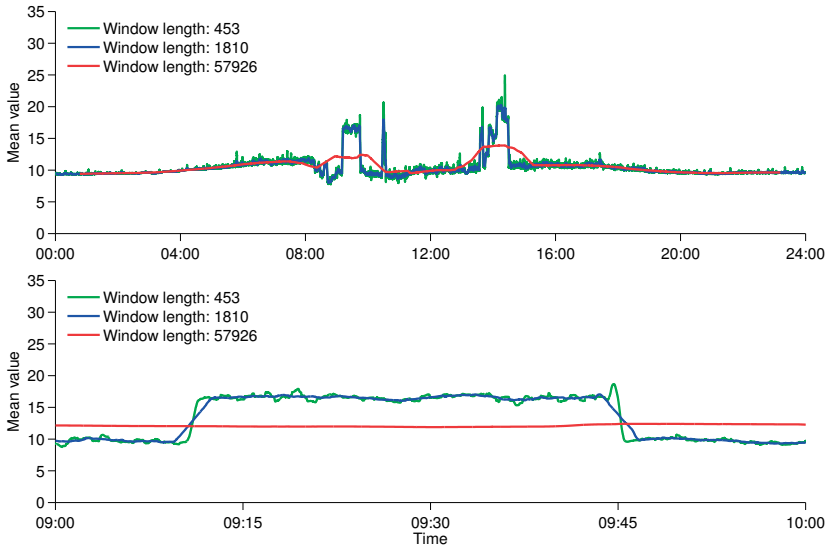

Figure 6.11. Smoothing of a time series of measurements from a strain sensor on a bridge in the Netherlands, using an optimised set of three window lengths. The top figure shows the full sequence (24 hours), while the bottom figure shows a zoom-in on the traffic jam that occurred between 9am and 10am.

sampling can be used to compute the set of window lengths more efficiently, making the method practical for (databases of) event sequences of any size. Finally, we found that the window lengths themselves can reveal interesting properties of the data; among other findings, we have identified relations between the optimal window lengths and (1) the structure of sequences composed of multiple interleaved sources and (2) the burstiness of events.

7. Conclusions and discussion

7.1 Conclusions

We have addressed several aspects of the problem *how to find structure in event sequences*. Specifically, the aspects that were dealt with are: (1) *how to compare event frequencies across (databases of) event sequences*, (2) *how to take into account the multiple testing problem when looking for local frequency deviations in event sequences*, and (3) *which granularities to use when looking for local patterns in an event sequence*.

We have introduced novel computational methods to tackle each of these aspects, discussed the related work, and reviewed existing methods and their aptitude for solving the task where applicable. We have conducted and presented extensive experiments to investigate the utility of each of the proposed methods for various applications.

In Chapter 4, we studied the problem how to compare word frequencies across corpora. By modelling texts as event sequences and a text corpus as a database, we mapped the problem to the question *how to compare event frequencies across databases of event sequences*. This problem is relevant, for example, when a linguist wants to test a hypothesis such as "word X is more frequent in male than in female speech". We have introduced two methods based on resampling and compared and evaluated these methods, along with several existing methods, with respect to their suitability to this task.

We concluded that the choice of the test, or more specifically, the representation of the data that is used in the test, matters, both in theory and in practice, as evidenced by experiments and case studies on two text corpora. We found that assuming that all words in a corpus are independent samples may lead to overestimating the significance of frequency

differences. Also, we demonstrated that the overestimation is related to the burstiness of words and that there exist bursty and non-bursty words at any frequency level, thus the overestimation occurs at all frequency levels.

In Chapter 5, we studied the problem of *how to take into account the multiple testing problem when looking for local frequency deviations in event sequences*. We introduced a novel statistical test for assessing the significance of event frequencies in subsequences when using a sliding window. The test provides strong control of the family-wise error rate and takes into account the dependency structure of overlapping subsequences. We argued that the exact p-values are difficult to compute and based the test on an easy-to-compute upper bound. We have shown experimentally that the test offers substantially increased power compared to existing alternatives.

We investigated the utility of the test on linguistic and biological sequences and found several novel and interesting patterns. We have illustrated that meaningful results can be obtained, and that the method remains sufficiently powerful even when testing hundreds of millions of hypotheses concurrently. We concluded that the proposed method is simple, fast and powerful and that the method can produce meaningful results on various types of data.

In Chapter 6, we studied the problem *which granularities to use when looking for local patterns in an event sequence*. We introduced the novel problem of identifying a set of window lengths that contain the maximal amount of information in the data, and we presented a generally applicable optimisation problem. We have shown that this optimisation problem is NP-hard in general, but that it can be approximated efficiently and solved exactly for certain simple statistics and data distributions. We studied the performance of the proposed optimisation algorithm, as well as the results for several statistics on both synthetic data and real data.

We have illustrated that the analytical results and the empirical results on synthetic data are useful as a baseline for practical use. We have shown that sampling can be used to compute the set of window lengths more efficiently, making the method practical for (databases of) event sequences of any size. Finally, we found that the window lengths themselves can reveal interesting properties of the data; for example, we identified relations between the optimal window lengths and (1) multi-scale structure of sequences and (2) the burstiness of events.

In short, we have shown that the methods introduced in this thesis can be used to compare and explore (databases of) event sequences with high computational efficiency, increased accuracy, and in novel ways.

7.2 Discussion

Several aspects of the main question *how to find structure in event sequences* have been addressed, but many questions remain, and new questions have emerged. For example, we found that the assessments of the bootstrap test and Welch's t-test are highly similar, so further research into their similarity and differences could lead to interesting insights. Also, it appears that the methods discussed in Chapter 4 could also be used to compare frequencies of subsequences, i.e., n-grams or collocations, between databases of event sequences. However, that was left outside the scope of this thesis.

In Chapter 5, we argued that computing the exact p-values under the null hypothesis is computationally costly, but a proof showing that the problem is indeed hard remains elusive. Similar to the comment about Chapter 4, it would appear that with some modifications, the method could be applicable to find significant deviations in local frequencies of subsequences instead of just single events. However, it is not immediately obvious how many occurrence opportunities there are for a given subsequence, thus this was left for further research.

In Chapter 6, we discussed the use of a small set of window lengths to analyse local patterns in event sequences, and we left the number of window lengths k as a choice to the user. The optimal number of window lengths to use in a certain setting depend on various factors, such as the data, the sliding window statistic, and the problem the user is trying to solve. There is a trade-off, as using higher k provides more information, but also higher cognitive load. A potentially fruitful direction for further study is selecting k by exploring the value of the loss function, similar to how the number of clusters is typically selected in clustering.

Considering the contributions presented in the three chapters, some questions regarding their intersection emerge: the method proposed in Chapter 5 takes into account the multiple testing problem directly, in order to avoid loss of statistical power, while in Chapter 4 a post-hoc correction (to control for the false discovery rate) is used. The post-hoc correction may bias the p-values in the conservative direction, which could perhaps

be avoided in similar fashion as in Chapter 5.

Also, in Chapter 6 it is argued that when mining local patterns using a sliding window, this is often best done using a small set of window lengths. However, in each of the experiments in Chapter 5, we used only one hand-picked window length. The reason is that using optimised window lengths may introduce an anti-conservative bias, and if multiple window lengths are used, there are further dependencies that should be taken into account. Thus, further research in this direction is warranted.

Bibliography

D. Aloise, A. Deshpande, P. Hansen, and P. Popat. NP-hardness of Euclidean sum-of-squares clustering. *Machine Learning*, 75(2):245–248, 2009.

E. G. Altmann, J. B. Pierrehumbert, and A. E. Motter. Beyond word frequency: Bursts, lulls, and scaling in the temporal distributions of words. *PLoS ONE*, 4 (11):e7678, 2009.

S. Argamon, M. Koppel, J. Fine, and A. R. Shimoni. Gender, genre, and writing style in formal written texts. *Text*, 23(3):321–346, 2003.

D. Arthur and S. Vassilvitskii. k-means++: the advantages of careful seeding. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2007.

R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman, Boston, 1999.

P. Baker. Times may change, but we will always have money: Diachronic variation in recent British English. *Journal of English Linguistics*, 39(1):65–88, 2011.

A.-L. Barabási. The origin of bursts and heavy tails in human dynamics. *Nature*, 435:207–211, 2005.

A. Baron, P. Rayson, and D. Archer. Word frequency and key word statistics in historical corpus linguistics. *Anglistik: International Journal of English Studies*, 20(1):41–67, 2009.

A. Bell. Language style as audience design. *Language in Society*, 13:145–204, 1984.

Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(1):289–300, 1995.

G. Benson. Tandem repeats finder: a program to analyze DNA sequences. *Nucleic Acids Research*, 27(2):573–580, 1999.

G. Bernardi. Isochores and the evolutionary genomics of vertebrates. *Gene*, 241 (1):3–17, 2000.

D. Biber. *Variation across speech and writing*. Cambridge University Press, Cambridge, 1988.

D. Biber. *Dimensions of register variation: A cross-linguistic comparison*. Cambridge University Press, Cambridge, 1995.

D. M. Blei and P. I. Frazier. Distance dependent Chinese restaurant processes. *Journal of Machine Learning Research*, 12:2461–2488, 2011.

C. Bourgain, E. Genin, H. Quesneville, and F. Clerget-Darpoux. Search for multifactorial disease susceptibility genes in founder populations. *Annals of Human Genetics*, 64(3):255–265, 2000.

British National Corpus. XML edition (version 3), distributed by Oxford University Computing Services on behalf of the BNC Consortium, 2007.

T. Calders, N. Dexters, and B. Goethals. Mining frequent items in a stream using flexible windows. *Intelligent Data Analysis*, 12(3):293–304, 2008.

B. Chiu, E. Keogh, and S. Lonardi. Probabilistic discovery of time series motifs. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2003.

K. W. Church and W. A. Gale. Poisson mixtures. *Natural Language Engineering*, 1(2):163–190, 1995.

M. K. Das and H.-K. Dai. A survey of DNA motif finding algorithms. *BMC Bioinformatics*, 8(Suppl 7):S21, 2007.

T. De Bie. Maximum entropy models and subjective interestingness: an application to tiles in binary databases. *Data Mining and Knowledge Discovery*, 23 (3):407–446, 2011.

E. D. Demaine, A. López-Ortiz, and J. I. Munro. Frequency estimation of internet packet streams with limited space. In *Proceedings of the European Symposium on Algorithms (ESA)*, 2002.

T. Dunning. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):51–74, 1993.

C. Elkan. Clustering documents with an exponential-family approximation of the Dirichlet compound multinomial distribution. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2006.

S. Evert. How random is a corpus? The library metaphor. *Zeitschrift für Anglistik und Amerikanistik*, 54(2):177–190, 2006.

M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the Internet topology. In *Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM)*, 1999.

L. Geng and H. J. Hamilton. Interestingness measures for data mining: A survey. *ACM Computing Surveys (CSUR)*, 38(3):9, 2006.

C. Giannella, J. Han, E. Robertson, and C. Liu. Mining frequent itemsets over arbitrary time intervals in data streams. Technical Report TR587, Indiana University, 2003.

A. Gionis, H. Mannila, T. Mielikäinen, and P. Tsaparas. Assessing data mining results via swap randomization. *ACM Transactions on Knowledge Discovery from Data*, 1(3):14, 2007.

L. Golab, D. DeHaan, E. D. Demaine, A. López-Ortiz, and J. I. Munro. Identifying frequent items in sliding windows over on-line packet streams. In *Proceedings of the ACM SIGCOMM Conference on Internet Measurement (IMC)*, 2003.

S. T. Gries. Syntactic priming: A corpus-based approach. *Journal of Psycholinguistic Research*, 34(4):365–399, 2005a.

S. T. Gries. Null-hypothesis significance testing of word frequencies: A follow-up on Kilgarriff. *Corpus Linguistics and Linguistic Theory*, 1(2):277–294, 2005b.

S. T. Gries. Dispersions and adjusted frequencies in corpora. *International Journal of Corpus Linguistics*, 13(4):403–437, 2008.

N. Haiminen, H. Mannila, and E. Terzi. Determining significance of pairwise cooccurrences of events in bursty sequences. *BMC Bioinformatics*, 9:336, 2008.

S. Hanhijärvi. Multiple hypothesis testing in pattern discovery. In *Proceedings of the International Conference on Discovery Science (DS)*, 2011.

M. A. Hearst. Multi-paragraph segmentation of expository text. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, 1994.

A. Hinneburg, H. Mannila, S. Kaislaniemi, T. Nevalainen, and H. Raumolin-Brunberg. How to handle small samples: bootstrap and Bayesian methods in the analysis of linguistic change. *Literary and Linguistic Computing*, 22(2): 137–150, 2007.

Y. Hochberg. A sharper Bonferroni procedure for multiple tests of significance. *Biometrika*, 75(4):800–802, 1988.

K. Hofland and S. Johansson. *Word Frequencies in British and American English*. Longman, London, 1982.

R. Jin and G. Agrawal. An algorithm for in-core frequent itemset mining on streaming data. In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, 2005.

R. M. Karp, S. Shenker, and C. H. Papadimitriou. A simple algorithm for finding frequent elements in streams and bags. *ACM Transactions on Database Systems*, 28(1):51–55, 2003.

S. M. Katz. Distribution of content words and phrases in text and language modelling. *Natural Language Engineering*, 2(1):15–59, 1996.

L. Kaufman and P. J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley & Sons, Hoboken, 1990.

D. Kifer, S. Ben-David, and J. Gehrke. Detecting change in data streams. In *Proceedings of the International Conference on Very Large Data Bases (VLDB)*, 2004.

A. Kilgarriff. Comparing corpora. *International Journal of Corpus Linguistics*, 6 (1):97–133, 2001.

A. Kilgarriff. Language is never, ever, ever, random. *Corpus Linguistics and Linguistic Theory*, 1(2):263–276, 2005.

E. F. Kirkness, V. Bafna, A. L. Halpern, S. Levy, K. Remington, D. B. Rusch, A. L. Delcher, M. Pop, W. Wang, C. M. Fraser, and J. C. Venter. The dog genome: Survey sequencing and comparative analysis. *Science*, 301(5641):1898–1903, 2003.

J. Kleinberg. Bursty and hierarchical structure in streams. *Data Mining and Knowledge Discovery*, 7(4):373–397, 2003.

A. Knobbe, H. Blockeel, A. Koopman, T. Calders, B. Obladen, C. Bosma, H. Galenkamp, E. Koenders, and J. Kok. InfraWatch: Data management of large systems for monitoring infrastructural performance. In *Proceedings of the International Symposium on Intelligent Data Analysis (IDA)*, 2010.

R. Kumar, J. Novak, P. Raghavan, and A. Tomkins. On the bursty evolution of blogspace. *World Wide Web: Internet and Web Information Systems*, 8(2): 159–178, 2005.

T. Lappas, B. Arai, M. Platakis, D. Kotsakos, and D. Gunopulos. On burstiness-aware search for document sequences. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2009.

D. Y. W. Lee. Genres, registers, text types, domains and styles: clarifying the concepts and navigating a path through the BNC jungle. *Language Learning & Technology*, 5(3):37–72, 2001.

G. Leech and R. Fallon. Computer corpora – what do they tell us about culture? *ICAME Journal*, 16:29–50, 1992.

E. L. Lehmann and J. P. Romano. *Testing Statistical Hypotheses*. Springer, New York, third edition, 2005.

J. Leskovec, J. Kleinberg, and C. Faloutsos. Graph evolution: Densification and shrinking diameters. *ACM Transaction on Knowledge Discovery from Data*, 1 (1):2, 2007.

C. Li, B. Wang, and X. Yang. VGRAM: improving performance of approximate queries on string collections using variable-length grams. In *Proceedings of the International Conference on Very Large Data Bases (VLDB)*, 2007a.

Y. Li, W.-K. Sung, and J. J. Liu. Association mapping via regularized regression analysis of single-nucleotide-polymorphism haplotypes in variable-sized sliding windows. *American Journal of Human Genetics*, 80(4):705–715, 2007b.

J. Lijffijt and S. T. Gries. Correction to Stefan Th. Gries' "Dispersions and adjusted frequencies in corpora". *International Journal of Corpus Linguistics*, 17 (1):147–149, 2012.

J. Lijffijt, P. Papapetrou, K. Puolamäki, and H. Mannila. Analyzing word frequencies in large text corpora using inter-arrival times and bootstrapping. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*, 2011.

J. Lijffijt, T. Säily, and T. Nevalainen. CEECing the baseline: Lexical stability and significant change in a historical corpus. In *Outposts of Historical Corpus Linguistics: From the Helsinki Corpus to a Proliferation of Resources (Studies in Variation, Contacts and Change in English 10)*, 2012.

J. Lijffijt, P. Papapetrou, and K. Puolamäki. A statistical significance testing approach to mining the most informative set of patterns. *Data Mining and Knowledge Discovery*, in press.

C.-H. Lin, D.-Y. Chiu, Y.-H. Wu, and A. L. P. Chen. Mining frequent itemsets from data streams with a time-sensitive sliding window. In *Proceedings of the SIAM International Conference on Data Mining (SDM)*, 2005.

C. Loader. Fast and accurate computation of binomial probabilities. Unpublished manuscript, 2000.

R. E. Madsen, D. Kauchak, and C. Elkan. Modeling word burstiness using the Dirichlet distribution. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2005.

H. B. Mann and D. R. Whitney. On a test of whether one of two random variables is stochastically larger than the other. *Annals of Mathematical Statistics*, 18 (1):50–60, 1947.

H. Mannila. Local and global methods in data mining: Basic techniques and open problems. In *Proceedings of the International Colloquium on Automata, Languages and Programming (ICALP)*, 2002.

H. Mannila and M. Salmenkivi. Finding simple intensity descriptions from event sequence data. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2001.

H. Mannila, H. Toivonen, and A. I. Verkamo. Discovery of frequent episodes in event sequences. *Data Mining and Knowledge Discovery*, 1(3):259–289, 1997.

F. J. Massey. The Kolmogorov-Smirnov test for goodness of fit. *Journal of the American Statistical Association*, 46(253):68–78, 1951.

R. A. Mathias, P. Gao, J. L. Goldstein, A. F. Wilson, E. W. Pugh, P. Furbert-Harris, G. M. Dunston, F. J. Malveaux, A. Togias, K. C. Barnes, T. H. Beaty, and S.-K. Huang. A graphical assessment of p-values from sliding window haplotype tests of association to identify asthma susceptibility loci on chromosome 11q. *BMC Genetics*, 7:38, 2006.

A. Mueen, E. J. Keogh, Q. Zhu, S. Cash, and B. Westover. Exact discovery of time series motifs. In *Proceedings of the SIAM International Conference on Data Mining (SDM)*, 2009.

M. L. Newman, C. J. Groom, L. D. Handelman, and J. W. Pennebaker. Gender differences in language use: an analysis of 14,000 text samples. *Discourse Processes*, 45:211–236, 2008.

B. V. North, D. Curtis, and P. C. Sham. A note on the calculation of empirical p-values from Monte Carlo procedures. *The American Journal of Human Genetics*, 71(2):439–441, 2002.

M. P. Oakes and M. Farrow. Use of the chi-squared test to examine vocabulary differences in English-language corpora representing seven different countries. *Literary and Linguistic Computing*, 22(1):85–100, 2007.

S. Papadimitriou and P. Yu. Optimal multi-scale patterns in time series streams. In *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD)*, 2006.

P. Papapetrou, G. Benson, and G. Kollios. Mining poly-regions in dna sequences. *International Journal of Data Mining and Bioinformatics*, 6(4):406–428, 2012.

M. Paquot and Y. Bestgen. Distinctive words in academic writing: A comparison of three statistical tests for keyword extraction. In *Corpora: Pragmatics and Discourse; Papers from the International Conference on English Language Research on Computerized Corpora (ICAME)*, 2009.

K. Pearson. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Philosophical Magazine Series 5*, 50(302):157–175, 1900.

P. Rayson. *Matrix: A statistical method and software tool for linguistic analysis through corpus comparison*. PhD thesis, Lancaster University, 2003.

P. Rayson and R. Garside. Comparing corpora using frequency profiling. In *Proceedings of the Workshop on Comparing Corpora (WCC)*, 2000.

P. Rayson, G. Leech, and M. Hodges. Social differentiation in the use of English vocabulary: Some analyses of the conversational component of the British National Corpus. *International Journal of Corpus Linguistics*, 2(1):133–152, 1997.

P. Rayson, D. Berridge, and B. Francis. Extending the Cochran rule for the comparison of word frequencies between corpora. In *Proceedings of the International Conference on Statistical Analysis of Textual Data (JADT)*, 2004.

S. E. Robertson and S. Walker. Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. In *Proceedings of the ACM SIGIR International Conference on Research and Development in Information Retrieval (SIGIR)*, 1994.

G. D. Ruxton. The unequal variance t-test is an underused alternative to Student's t-test and the Mann-Whitney U test. *Behavioral Ecology*, 17(4):688–690, 2006.

M. Salmenkivi and H. Mannila. Using Markov chain Monte Carlo and dynamic programming for event sequence data. *Knowledge and Information Systems*, 7(3):267–288, 2005.

S. K. Sarkar and C.-K. Chang. The Simes method for multiple hypothesis testing with positively dependent test statistics. *Journal of the American Statistical Association*, 92(440):1601–1608, 1997.

J. P. Shaffer. Multiple hypothesis testing. *Annual Review of Psychology*, 46:561–584, 1995.

L. Sörnmo and P. Laguna. *Bioelectrical Signal Processing in Cardiac and Neurological Applications*. Elsevier Academic Press, Boston/Amsterdam, 2005.

K. Spärck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21, 1972.

Standardised-spelling Corpus of Early English Correspondence. Compiled by Terttu Nevalainen, Helena Raumolin-Brunberg, Samuli Kaislaniemi, Jukka Keränen, Mikko Laitinen, Minna Nevala, Arja Nurmi, Minna Palander-Collin,

Tanja Säily and Anni Sairio. Standardised by Mikko Hakala, Minna Palander-Collin and Minna Nevala. Department of English / Department of Modern Languages, University of Helsinki, 2012.

B. Szmrecsanyi. Language users as creatures of habit: A corpus-based analysis of persistence in spoken English. *Corpus Linguistics and Linguistic Theory*, 1 (1):113–149, 2005.

R. Tang, T. Feng, Q. Sha, and S. Zhang. A variable-sized sliding-window approach for genetic association studies via principal component analysis. *Annals of Human Genetics*, 73(6):631–637, 2009.

H. T. T. Toivonen, P. Onkamo, K. Vasko, V. Ollikainen, P. Sevon, H. Mannila, M. Herr, and J. Kere. Data mining applied to linkage disequilibrium mapping. *American Journal of Human Genetics*, 67(1):133–145, 2000.

J. C. Venter et al. The sequence of the human genome. *Science*, 291(5507):1304–1351, 2001.

U. Vespier, A. Knobbe, S. Nijssen, and J. Vanschoren. MDL-based analysis of time series at multiple time-scales. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*, 2012.

G. I. Webb. Layered critical values: A powerful direct-adjustment approach to discovering significant patterns. *Machine Learning*, 71(2–3):307–323, 2008.

W. Weibull. A statistical distribution function of wide applicability. *Journal of Applied Mechanics*, 18(3):293–297, 1951.

B. L. Welch. The generalization of 'Student's' problem when several different population variances are involved. *Biometrika*, 34(1–2):28–35, 1947.

F. Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1 (6):80–83, 1945.

F. Yates. Contingency table involving small numbers and the χ^2 test. *Supplement to the Journal of the Royal Statistical Society*, 1(2):217–235, 1934.

Y. Zhu and D. Shasha. Efficient elastic burst detection in data streams. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2003.

G. K. Zipf. *Human Behaviour and the Principle of Least Effort: An Introduction to Human Ecology*. Addison-Wesley Press, Cambridge, 1949.

DISSERTATIONS IN INFORMATION AND COMPUTER SCIENCE

Aalto-DD133/2012 Ahlroth, Lauri
Online Algorithms in Resource Management and Constraint Satisfaction. 2012.

Aalto-DD158/2012 Virpioja, Sami
Learning Constructions of Natural Language: Statistical Models and Evaluations. 2012.

Aalto-DD20/2013 Pajarinen, Joni
Planning under Uncertainty for large-scale problems with applications to wireless networking. 2013.

Aalto-DD29/2013 Hakala, Risto
Results on Linear Models in Cryptography. 2013.

Aalto-DD44/2013 Pylkkönen, Janne
Towards Efficient and Robust Automatic Speech Recognition: Decoding Techniques and Discriminative Training. 2013.

Aalto-DD47/2013 Reyhani, Nima
Studies on Kernel Learning and Independent Component Analysis. 2013.

Aalto-DD70/2013 Ylipaavalniemi, Jarkko
Data-driven Analysis for Natural Studies in Functional Brain Imaging. 2013.

Aalto-DD61/2013 Kandemir, Melih
Learning Mental States from Biosignals. 2013.

Aalto-DD90/2013 Yu, Qi
Machine Learning for Corporate Bankruptcy Prediction. 2013.

Aalto-DD128/2013 Ajanki, Antti
Inference of relevance for proactive information retrieval. 2013.

Many types of data, e.g., natural language texts, biological sequences, or sensor data, contain sequential structure. Analysis of such sequential structure is interesting for various reasons, for example, to discover recurring patterns, to detect that data consists of several homogeneous parts, or to find parts that are surprising compared to the rest of the data. The main question studied in this thesis is how to identify global and local patterns in event sequences. Within this broad topic, several subtopics are addressed: comparison of event frequencies across sequences, finding areas where particular events are surprisingly frequent or infrequent, and choosing the best granularity for finding local patterns in event sequences. The main contributions are computational methods that can be used to compare and explore databases of event sequences with high computational efficiency, increased accuracy, and that offer new perspectives on the sequential structure of data.

Aalto-DD 205/2013

ISBN 978-952-60-5474-2
ISBN 978-952-60-5475-9 (pdf)
ISSN-L 1799-4934
ISSN 1799-4934
ISSN 1799-4942 (pdf)

Aalto University
School of Science
Department of Information and Computer Science
www.aalto.fi

BUSINESS + ECONOMY

ART + DESIGN + ARCHITECTURE

SCIENCE + TECHNOLOGY

CROSSOVER

DOCTORAL DISSERTATIONS