Measuring and modelling information flows in real-world networks

Bridget Smart

January 30, 2023

Thesis submitted for the degree of
Master of Philosophy
in
Applied Mathematics and Statistics
at The University of Adelaide
Faculty of Engineering, Computer and Mathematical Sciences
School of Mathematical Sciences

THE UNIVERSITY
of ADELAIDE

Contents

Signed Statement **xiii**

Acknowledgements **xv**

Abstract **xvii**

1 Introduction **1**
1.1 Motivation 2
1.2 Structure and contributions 5

2 Survey of techniques to measure pairwise similarity in point processes **9**
2.1 Motivation 9
2.1.1 Scope 10
2.2 Conceptual Framework 12
2.3 Related work 15
2.3.1 Previous surveys 16
2.4 Applications 17
2.5 Notation 20
2.5.1 Association between time series 20
2.5.2 Mathematical details 25
2.5.3 Stochastic processes 26
2.6 Heuristic approaches 28
2.6.1 Correlation measures 29
2.6.2 Distance metrics 31
2.6.3 Inter-event time metrics 33
2.6.4 Time coordination windows 33
2.6.5 Labelling coordinated events 34
2.7 Information-theoretic approaches 37
2.7.1 Dealing with real data 40
2.7.2 Choice of entropy estimator 42

2.7.3 Examples in the literature . . . 43
2.8 Stochastic approaches . . . 44
2.8.1 Homogeneous Poisson processes . . . 46
2.8.2 Hawkes processes . . . 46
2.8.3 General considerations for stochastic approaches . . . 50
2.9 Machine learning approaches . . . 51
2.9.1 Examples of machine learning approaches . . . 52
2.10 Considerations . . . 55
2.10.1 Practical considerations . . . 55
2.10.2 Pairwise relationships . . . 57
2.10.3 Classifying pairwise relationships . . . 59
2.11 Conclusion . . . 61

3 A simulation framework to evaluate association between point processes 63

3.1 Introduction . . . 63
3.1.1 Gap in the literature . . . 65
3.2 Mathematical background . . . 65
3.3 Pairwise relationships between point processes . . . 67
3.3.1 Types of association . . . 67
3.3.2 Association in context . . . 67
3.3.3 Existing approaches . . . 67
3.4 Data simulation . . . 69
3.4.1 Simulation model . . . 69
3.4.2 Traversing the set of processes . . . 71
3.4.3 Coordinating state . . . 71
3.4.4 Generating event times . . . 72
3.4.5 Additional details . . . 74
3.4.6 Algorithm . . . 75
3.4.7 Parameters . . . 76
3.5 Evaluating performance . . . 79
3.5.1 Evaluating against directly coordinating behaviours . . . 79
3.5.2 Evaluating against similarity of generating processes . . . 81
3.5.3 Representing a range of behaviours . . . 82
3.6 Sample results . . . 86
3.6.1 Comparing evaluation heuristics . . . 86
3.6.2 Event counts . . . 91
3.6.3 Average time between self excited events . . . 93
3.6.4 Distribution of coordinated relationship strengths . . . 94
3.7 Conclusion . . . 95

4 Publication: #IStandWithPutin versus #IStandWithUkraine: The interaction of bots and humans in discussion of the Russia/Ukraine war 97
4.1 Summary . . . 97
4.2 Mathematical background: entropy estimators in social media . . . 98
4.2.1 Measuring aggregated information flow . . . 99
4.2.2 An empirical significance test of information flow . . . 100
4.3 Statement of Authorship: #IStandWithPutin versus #IStandWithUkraine: The interaction of bots and humans in discussion of the Russia/Ukraine war 102
Publication . . . 105

5 Publication: The entropy rate of Linear Additive Markov Processes 119
5.1 Summary . . . 119
5.2 Statement of Authorship: The entropy rate of Linear Additive Markov Processes . . . 121
Publication . . . 123

6 Conclusion 133

A Complete sample results for evaluation of association measures 137
A.0.1 Metrics . . . 138
A.0.2 Infomation-theoretic . . . 141

Bibliography 143

List of Figures

2.1 Figure showing a broad classification of several of the approaches outlined in this review, comparing them on how explainable the approach is, and whether the method has an event- or process-level focus. 14

2.2 Comparison of the target behaviour of a subset of techniques. This aims to describe which types of behaviour will be most strongly detected using each approach. 15

2.3 Overview of information types used to measure similarity and coordination in point processes. 21

2.4 Directly coordinating processes. Here, X generates events which Y directly responds to. The time delay between the two dependent events is denoted δ_i. 22

2.5 Indirectly coordinating processes. Here, the parent process generates events which both child processes respond directly to. The time delay between the two dependent events is denoted δ_i and δ_j. 23

2.6 Processes with similar generating functions. 24

2.7 Demonstration of the same set of events from sources X and Y represented as a pair of simple point processes (i) and as a marked point process (ii). The labels for the marked point process are shown underneath the process, and this additional information is encoded into the point process. While containing equivalent information, one of these two frameworks can be more useful in particular approaches, so it is useful to keep this alternative construction in mind. 26

2.8 Smoothed function 30

2.9 Hard and soft thresholds. 36

2.10 Process of discretising a sequence with a window size of w. 41

2.11 An example of the underlying Hawkes process model for the event times shown at the top. For the given process X, with events $x_1, ..., x_8$, we can represent the counting process (i), and the intensity function (ii). In this figure, an exponential function was used for the self-exciting intensity. (i) Shows the counting process, $N(t)$, where the event count increases as each event occurs. This function is strictly monotonically non-decreasing. (ii) represents the base intensity with $\hat{\lambda}_0$, the y-intercept for the intensity function. As each event occurs, there is a spike due to $\phi(\cdot)$ with this decaying exponentially until the next event occurs. 48

3.1 Example of event times from a trio of coordinating processes. Dark grey indicates the respective process is currently in a coordinating state. Coloured arrows indicate source-target relations between events, with colours corresponding to the matching relationship shown in the underlying coordination network (left). Before $T_{\text{burn-in}}$, no processes are able to coordinate. 70

3.2 Example of topological ordering 71

3.3 Diagram of the HMM for each process. 72

3.4 Distributions of edge weights in A before normalisation. 85

3.5 Performance for association method using a co-occurrence window of size 1 for measuring directly coordinated behaviours (i, iii) and measuring similarity between generating processes (ii, iv). The best performance occurs with a fixed coordination delay of 1 unit (I). Performance drops for coordination delays which result in fewer directly coordinated event pairs which fall outside the co-occurrence window. Interestingly, performance is comparably low for uniformly distributed delays and fixed delays which fall outside the target window, although the low proportion of coordinated target events from a uniform delay would be low. These trends do not continue when assessing similarity between generating processes (ii), with the association method using a co-occurrence window of size 1 not able to provide much discrimination between different coordinated event delays. The differences between performance for each of the target behaviours is also evident in how the methods perform when processes spend different amounts of time in coordinating states. In (iii), the association method using a co-occurrence window of size 1 is able to best detect directly coordinated behaviours when more time is spend in a coordinating state. (iv) shows performance is not affected greatly by the amount of time spent in either a non-coordinating or coordinating state, with a slight trend for better performance for a higher proportion spend in a coordinating state. 88

3.6 Performance for an association measure which considered the median inter-event times between pairs of processes. In (i), performance is assessed against detecting directly coordinated behaviours and in (ii) performance is considered for detecting similarity between generating functions. 89
3.7 To understand how the length of coordination window impacts performance against coordinated time delay functions when detected directly coordinated behaviours, approaches with a window size of 1 unit (i), 5 units (ii) and 10 units (iii) were considered. It is expected that co-occurance counting will perform well when coordinated event delays are drawn from a distribution with a matching window size when considering directly coordinated behaviours. For a co-occurrence window of size 1, we can see best performance for a fixed delay of 1 unit and an exponential delay with lower mean delay. For the delay of 5 units, we can see this trend continues, with good performance for both exponential delays and a fixed coordination delay of 1 and 5 units. For a co-occurrence window of 10 units, each function has comparable performance, which is expected as most coordinated event delays should fall within 10 units given the considered parameter range. . . 90
3.8 Each of these graphs show the affect of the average number of events per process on the performance of the association measure. (i) and (ii) are results for detecting direct coordination and similarity for an association measure using a co-occurance window of size 10, (iii) and (iv) show these quantities for an association measure which uses the pairwise transfer entropy between processes. 92
3.9 Relationship between detecting directly coordinated behaviour (i) and similarity between generating processes (ii) for an association measure which uses an exponentially penalised inter-event time. There is more variance across the range of values for performance for directly coordinated events, although neither show a strong trend. 93
3.10 Both (i) and (ii) show the performance of a median inter-event time association measure against the variance of coordination relationship strengths. Figure (i) shows performance for directly coordinated processes and (ii) shows performance for detecting generating similarity. There is a general downward trend in (i), although the variance also seems to decrease. As the relationship strengths are drawn from an exponential distribution and normalised, as variance increases, there are fewer, but stronger relationships. This indicates that the median inter-event time is best able to detect direct coordination when there are fewer pairs of coordinating processes rather than many coordinating processes. 94

4.1 Results from the empirical significance test for the unclassified pro-Russian account group. 101

4.2 Average hourly probabilities of bots tweeting query hashtags (top). Hourly frequency of the query hashtags (bottom). The time period we consider is the first fortnight after Russia's invasion of Ukraine. Both plots also include five significant events over this time period. Note that the query hashtags can be found in Section 3. We can observe a significant spike in the bot activity of several bot types on the 2nd and 4th of March. The spike in bot activity on the 2nd of March aligns with Russia's capture of Kherson, and also aligns with a significant increase in pro-Russia hashtags. This spike in activity was due to an increase in activity of pro-Russian bots – likely used by Russian authorities. The spike in bot activity on the 4th of March aligns with when the use of pro-Russia hashtags diminished, but also when Russia captured the Zaporizhzhia nuclear power plant. This spike was due to an increase in activity of pro-Russian bots (before being removed) and an increase in activity of pro-Ukrainian bots – likely by pro-Ukrainian authorities in response to Russian bots. 106

4.3 Average hourly Botometer results showing the daily cycle. The time series observed in Figure 4.2 (top) is averaged based on the hour of the day (UTC time). 109

4.4 Probabilities of bot types based on national lean and bot classification. National lean and bot classification are described in Sections 3.1 and 3.2. . 109

4.5 (a) Aggregated net outward information flows by account type and national lean. The outward flows for each aggregated group predominantly fall in the $(0, 0.5)$ range, however 'ProRussia NonBot', 'ProUkraine Financial', 'ProUkraine Spammer' and 'ProRussia FailedToClassify' net flows tend to be greater. Values above the 80th percentile are labelled – the majority of these represent flows into the 'ProRussia Self Declared' group. (b) Heatmap of empirical p-values for each intergroup net information flow, showing significance of the difference in median outbound flows between groups. This test is used to form the network shown in (c). (c) Significant net information flows between groups. Not pictured are aggregated groups with inadequate sample size. Each aggregated group is coloured by national lean; edges are weighted by magnitude of net information flow. Significant flows out of 'ProRussia NotBot' accounts indicate that information flows from these accounts to other groups, most commonly groups with a balanced or 'ProUkraine' lean. Most intergroup flows for groups with a 'ProUkraine' lean are to other groups with 'ProUkraine' lean, with no significant flows from 'ProUkraine' account groups to 'ProRussia' account groups. (d) Lists the 15 highest self entropy rate values for the aggregated groups, with the top three groups all having a 'ProUkraine' lean. 111

4.6 (a) Shows the lagged cross correlations between the mean hourly Botometer Overall CAP score and the CAP weighted mean compound sentiment. Each lag represents an offset of the CAP weighted mean compound sentiment in hours. These correlations are significant in both lag directions. (b) Considering a scatterplot of the two timeseries with trend and burn-in samples removed, reveals a positive linear relationship between them. (c) The timeseries represent the CAP weighted mean compound sentiment grouped by national lean, with 'ProRussia', 'ProUkraine' and 'Balanced' accounts considered separately. To aid interpretation of these timeseries, the tweet frequency of each type (from all accounts) and some significant event markers are given. Before the 2nd of March there was minimal activity from 'ProRussia' accounts, so the CAP weighted mean compound sentiment estimate has high variance and is of low quality. After March 3rd, there is a spike in Balanced CAP weighted mean compound sentiment, from Balanced accounts, suggesting that these accounts were producing more positive tweets overall, potentially in response to humanitarian corridors opening. 113

4.7 A series of pairwise Granger Causality Tests are performed to examine whether the activity of bot types is Granger-causing changes in discussions of the LIWC categories. The heat maps colour describes the bot effect strength and direction from the Granger Causality Test (over 12 hours/lags) between the time series of hourly bot proportions and the time series of hourly LIWC category proportions. The number in the centre describes the most prolific lag in the Granger Causality Test. We calculate the bot effect strength using the F-score from an F-test on the Granger Causality linear models. Moreover, we calculate the bot effect direction and most prolific lag using the sign and lag (respectively) of the largest β coefficient from Eq. 4 in Section 5.1. We perform a Bonferroni Adjustment on the p-values from the Granger Causality Tests and only show the Bot Types and LIWC Categories with a significant adjusted p-value (< 0.05). 114

4.8 Word Clouds which demonstrate the frequency of words in particular LIWC categories 114

4.9 Lagged cross correlations between the hourly Self Declared bot proportions and the significant hourly LIWC Category proportions (significance is determined from the results in Figure 6). We consider 48 hours/lags for each of these plots and represent the significance threshold using a horizontal dotted line. This shows the extent to which the bots drive changes in online discussion and how long these effects can persist for. 115

5.1 (i) A diagram of the underlying transition matrix for each of the models shown in (ii) - (vi). While these models all produce sample paths with differing dependency structures, they all have the same entropy rate, as proven in Theorem 4.1. In blue, the dependency structure of each model is visualised using Cramér's V for lags up to 40. This provides a measure of correlation for symbolic variables and is analogous to considering a lagged-cross correlation function. In (ii), the dependency structure is shown for a first-order Markov model. In (iii) - (vi) we see the frequency of w_k, or the distribution of the size of the backward step directly influences the shape of the dependency structure. While the Cramér's V is calculated for a lag of 0 for the first-order Markov Model, it is excluded from plots (iii)-(vi) since the value is much larger due to the Markov Chain's memoryless property. It was excluded to allow the similarity between the frequency distribution of step size and dependency structure to be more easily seen. 126

5.2 Plots to show convergence for the normalised entropy estimate value against $\log_2 p_{\text{artificial}}$, which was used to ensure ergodicity. (i) Shows hyperparameter sensitivity for the first-order Markov models and (ii) for the LAMP models. Each plot shows the effect of the weight of this artificial link on both the first-order Markov model estimate and the estimate obtained using the LAMP model for each dataset. We aim to find a region where the estimates are insensitive to the choice of hyperparameter. A dashed black line on each plot indicates the value when the artificial link weight is 2^{-15}. This value was chosen as a global value, since it is a reasonable choice for all dataset model combinations, apart from the BrightKite dataset first-order Markov model, when a value of 2^{-10} was used to obtain the final estimate. This alternative value is indicated by a grey dashed line. The convergence curve for the WIKISPEEDIA and REUTERS datasets overlapped for the first-order Markov model, so the values for the REUTERS dataset was offset by +0.04 for visualisation. Small vertical offset was also added to the REUTERS and BRIGHTKITE datasets in the LAMP model visualisation (+0.01 and -0.005 respectively). 131

A.1 Co-occurrence counts, the coordination window size is 1 unit 138
A.2 Co-occurrence counts, the coordination window size is 5 units 138
A.3 Co-occurrence counts, the coordination window size is 10 units 139
A.4 Median inter-event time 139
A.5 Exponentially penalised inter-event time 140
A.6 Variance of inter-event times 140
A.7 Transfer Entropy 141
A.8 Time agnostic entropy 141

Signed Statement

I certify that this work contains no material which has been accepted for the award of any other degree or diploma in my name, in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text. In addition, I certify that no part of this work will, in the future, be used in a submission in my name, for any other degree or diploma in any university or other tertiary institution without the prior approval of the University of Adelaide and where applicable, any partner institution responsible for the joint award of this degree.

The author acknowledges that copyright of published works contained within the thesis resides with the copyright holder(s) of those works.

I give permission for the digital version of my thesis to be made available on the web, via the University's digital research repository, the Library Search and also through web search engines, unless permission has been granted by the University to restrict access for a period of time.

I acknowledge the support I have received for my research through the provision of a Westpac Future Leaders Scholarship and an Australian Government Research Training Program Scholarship.

Signed: Date: 30/01/2023

Acknowledgements

To my supervisors Professor Lewis Mitchell, Professor Matthew Roughan and Dr Anna Kalenkova, thank you. Your unwavering support, guidance and (often) much-needed advice has not only made this thesis possible but has also supported me to grow as a person and as a mathematician. Thank you for being generous with your time and knowledge, for the coffee, snacks and banter, and for always inspiring me to think outside the box.

Thank you to Josh and Sara for being incredible collaborators, providing invaluable perspective and always sharing your knowledge.

I am deeply grateful to Wills, Christina, Dylan, Saka and Adam for many afternoons filled with laughter, debates and chocolates. Thank you for coming along this journey with me. To Wills and Christina, thank you for always being up for cake, tea or a good derivation; you got me out of many a mathematical quandary.

To all of my amazing friends, thank you for always being there to listen, distract, encourage and support me.

To Grandad, for showing me how curious the world is and teaching me how to never stop asking questions. To my parents, to Nan-nan and to my sister, thank you for your love and for always backing me up 100%. To my cat, Ginger, thank you for keeping the word count increasing, even when I was taking a break. Finally, to my partner Tom, thank you for everything.

Abstract

In many real-world systems, numerous interacting components produce intricate structures and complex emergent dynamics. Modelling this complexity can be challenging, but doing so can produce meaningful insights into both individual and collective behaviours, revealing non-trivial patterns from simple pairwise interactions. This thesis focuses on models which describe how the information which is passed around these complex systems influences how components interact and behave.

To understand the dynamics of complex systems, models to estimate features of pairwise relationships and information flows are critical. Often, existing techniques have not been tested in realistic data settings, or the statistical significance of results is unknown. These limitations result in a number of domain-specific techniques which are inaccessible outside of their originating field, exposing a need for robust, general frameworks to evaluate their performance.

In Chapter 2, we begin by considering measures of association between point processes, a stochastic object often used to represent sets of event times; for example in social media, cognitive neuroscience or financial applications. A survey is conducted of existing techniques across a broad range of fields. These techniques are presented in four classes; heuristic, information-theoretic, stochastic and machine learning based. A major challenge for this body of research is a lack of universal notation making many approaches inaccessible. To overcome this, we present a discussion on types of association, a general notation, and a conceptual framework. The benefits and drawbacks of each class are discussed, highlighting potential applications and practical considerations. While this survey increases the accessibility of techniques across fields, empirical evaluation remains a useful validation tool to ensure approaches are robust. In Chapter 3, a simulation framework is described which allows complex, realistic event time data to be generated for a set of point processes. This framework is used to construct a tool that can be used to evaluate the relevance of techniques to measure association in time series.

While event time data is readily available, it is not unusual to have access to richer data associated with each event. In the case of social media data, posts can be taken as events, with the textual information used to measure information flows and influence between

accounts online. In existing work, non-parametric information-theoretic estimators have been applied to measure information flows between individual accounts online. Here, we extend these estimators to calculate information flows between groups of accounts, and propose an empirical significance test to evaluate which of these flows are meaningful in context. In Chapter 4, these contributions are applied to a Twitter dataset concerning discussion of the 2022 Russian invasion of Ukraine, and is accompanied by an exploration of the potential impacts of these information flows.

Inspired by the desire to explore simulation-based validation for sequential entropy estimators, limitations on the convergence rate of the non-parametric entropy estimator were discovered. To overcome these limitations, work was done to calculate the theoretical entropy rate of a Linear Additive Markov Process (LAMP) model, a model used to generate sequential data with specified long-range dependency structures. As described in Chapter 5, the aim of this work is to enable the behaviour of non-parametric estimators to be explored on more realistic sequences with dependency structures, opening up an avenue for future work.

These results build upon and improve the robustness of existing techniques to measure information flows in real-world networks, ultimately improving our ability to model complex systems appropriately.

Chapter 1

Introduction

In complex systems where many factors interact, sophisticated structures produce interesting and unexpected emergent dynamics. At first glance, the complexity of these systems appears to suggest intricate structures and higher-order effects, however networks of simple pairwise relationships are often adequate models. When numerous pairwise interactions take place over a complex network, their effects combine and compound in surprising ways, producing unexpected patterns from the micro- to the macro scale.

We encounter these complex systems in many places; across our society and all scales in the natural world, from modelling communities in the microbiome (Yuan et al., 2022) to capturing the distributions of objects within astronomical systems (Klishin and Chilingarian, 2016). Complex network models give us a tool to understand the fundamental drivers of these systems, revealing significant insights into the world around us.

It can be difficult to validate techniques within these complex systems, and unpredictable emergent dynamics can make it difficult to assess the suitability of techniques for new contexts. Approaches to measure features of these systems need to be context specific and intentionally designed. Even the most carefully designed methods can have unexpected pitfalls, and validating the performance of techniques on problems with similar and differing features is a critical step.

This thesis aims to increase our understanding of techniques to measure pairwise relationships on networks. We explore and extend existing tools, develop models for simulation and evaluation to ultimately improve the accessibility and robustness of methods for measuring pairwise relationships on complex networks.

1.1 Motivation

In many domains, networks are employed to deal with complexity. They are used to quantify the effects of multiple interacting components, capturing both small and large scale dynamics. While network models can account for complex higher-order relationships between interacting components with various properties or interaction mechanisms, underpinning these network models is the assumption of pairwise causality (Newman, 2018). These pairwise relationships are a cornerstone of complex network models, providing a simple mechanism through which higher-order effects and structures can be modelled. Equipped with a good model of these pairwise relationships, network science can translate simple measurements from a system into complicated insightful models.

Pairwise relationships on networks can be characterised in several ways, and we are often interested in what these relationships tell us about how individual components and the overall network will respond and react to internal and external events. However, when settings are complex, appropriate models of these interactions may not be obvious, a challenge further compounded by inconsistent definitions and methodologies between fields. This is a growing problem in network science which can be overcome by strong mathematical frameworks, general notation and careful definitions which make domain-specific research more accessible, allowing insights to be easily applied to new contexts.

Many techniques to understand relationships on networks are motivated by a theoretical model of the underlying relationship (Zhang et al., 2020; Perinelli et al., 2021). These theoretical models help fill the gap in applications where the network or relationship structure is often unknown or difficult to measure. However, within each domain, research is often conducted independently, partially due to domain-specific notation and jargon limiting its accessibility. Large bodies of research on complex systems exist across network science (Newman, 2018; Albert and Barabási, 2002), finance (Bauwens and Hautsch, 2009), social sciences (Bagrow and Mitchell, 2018; South et al., 2022), geopolitics (Cederman, 2002; El-Sayed et al., 2012), and in the natural sciences (Novelli et al., 2019; Wang et al., 2022; Barabási and Oltvai, 2004). Improving the accessibility of knowledge ensures access to a large body of research, so techniques can be carefully selected and understood, rather than being chosen as a default option.

One of the aims of this thesis, is to improve the tools available for model selection and increase the accessibility of knowledge between fields. Making these concepts generally accessible necessitates a universal language and mathematical framework which is meaningful and applicable. In this work, we will primarily consider these relationships in the context of information flows, working to measure how components influence and respond to each other, and how similar their behaviour is. For many people, information may be primarily associated with human communication, but (Shannon, 1948) introduced information as a fundamental measure of surprise. Understanding information as a measure of

randomness lead to information theory, which can be applied to reveal how components relate within complex systems. Information may represent the flow of ideas between individual components (Bagrow et al., 2019), the presence of a reactive relationship (Shorten et al., 2021), or reveal how similar a pair of processes are (Kwon et al., 2009).

In practice, this problem is compounded by the challenges of working with real data. In real data a ground truth may be unknown, and measurements may be inaccurate, misreported or otherwise compromised. Data sets can look very different between fields, and the goals of modelling can be varied. Tools used to measure relationships on these networks often need to simultaneously infer complex relationships between parts of a network while remaining robust to the traditional challenges which data brings. The complexity of these systems further inhibits our ability to identify appropriate techniques, and evolving contexts mean that the problem space may be constantly changing.

In this thesis, we will develop simulation as a tool to overcome this challenge, using a universally applicable framework to generate data sets which are meaningful across a range of applications, and using these simulated data sets to explore how techniques perform in a range of settings. For this evaluation, we use simulation as it offers a solution to investigate new, complex situations without relying on real data, opening up potential applications to a wide range of data settings.

By generating artificial data sets with known ground truth, simulation can also give insights into how existing and new techniques will perform on large, complex systems where complete information is unavailable. Good models to simulate data are critical for validation, and models with clear behaviour and meaningful parameters are particularly useful for generating data sets with given properties. When simulated data can be generated with features that match the desired context, these artificial data sets can be used to validate and tune techniques. Simulated data also offers a ground truth, allowing empirical assessment of techniques which is not possible generally.

In this first part of this thesis, we consider methods for measuring the association between pairs of point processes. Point processes are often used to represent sets of event times, and occur when only the product of interactions within a network can be observed. This is common in applications where events are easier to detect than relationships, or some components are hidden: for example in financial data, where a change in price of a stock may be driven by a complex underlying relationship between parties (Bauwens and Hautsch, 2009); in seismology where events can be driven by a series of earthquakes and aftershocks (Cai et al., 2022); or in cognitive neuroscience where complex networks are hypothesised to underpin cognitive processes, and their structure is probed using spikes in electrical potential (Victor, 2005). Event data can be used to measure relationships between components in these complex networks, revealing its hidden structure.

This work aims to improve the accessibility and robustness of techniques to measure the

association between pairs of point processes. We begin by generalising notation and definitions, presenting a comprehensive survey of existing approaches and discussing practical considerations, with the goal of improving the accessibility between fields. While improving understanding of these approaches is useful, being able to validate and understand their performance in data settings with ground truth is a critical step in ensuring approaches are robust. We create a simulation and evaluation framework to generate realistic event time data and validate the performance of techniques for a range of behaviours.

Alongside this work, we also build on information-theoretic approaches for sequential data, using the common application of natural language data on online social networks to provide context and explore the uses of an information flow estimator on a real data set. These information flows are used to build networks, which are an intuitive solution for modelling our online social networks.

Information flow estimators can be applied to online social networks, to measure how the public reacts to and forms opinions around topics from politics to public health (South et al., 2022; Graham et al., 2021). The sheer volume of data can provide valuable insights, particularly when these large data sets are understood and managed efficiently. Researchers have used this data to make inferences from public online activity to the private habits of an individual, to predict and explore the occurrence of "echo chambers", and to understand and measure public opinion on specific topics (South et al., 2022; Keller et al., 2020; He et al., 2013; Bagrow et al., 2019). Considering the influence which social media users can have on each other provides an opportunity to understand phenomena such as echo chambers, "fake news" and disinformation campaigns. Understanding how we can better capture and understand information flow online is key to developing strategies to counteract malicious activity, ensure productive dialogue in our online communities, support cybersecurity and promote trust and transparency in online social networks. We use this motivating example to explore how we can quantify and understand types of influence within these networks, providing insight into how we can model and analyse complex interactions between users.

We extend an information flow estimator to measure information flows between groups of users online, using it to measure flows between different classes of accounts on a data set containing Tweets relating to the 2022 Russian invasion of Ukraine. We describe and implement an empirical significance test and use linguistic analysis techniques to add meaning to the measured information flow network. These contributions highlight the practicality of measuring the significance of information-theoretic measures, and the importance of being able to validate results in controversial contexts. Finally, the Linear Additive Markov Model (LAMP) is equipped with a theoretical entropy rate to allow for empirical validation of the performance of information-theoretic entropy rate estimators on sequential data with long-term dependency structures.

1.2 Structure and contributions

This work begins by considering how we can measure association between processes using only event time data. This problem appears in many settings from telecommunications to cellular systems. Existing work uses sets of event times generated from processes to understand the pairwise relationships between them. However, this work is often inaccessible beyond its field of origin due to inconsistent notation and definitions. In these complex data settings, the behaviours and performance of measures of pairwise relationships is often not clearly understood or defined, and quality checking relies on similar applications and hypothesised ground truths. Chapter 2 begins with a survey of existing approaches, proposing a general notation scheme and framework for comparing techniques between fields. This is complemented with a new simulation and evaluation framework described in Chapter 3 to evaluate these measures and assess their performance in a range of settings. This simulation framework is provided as open source PYTHON code, available at `https://github.com/bridget-smart/PointProcessSimulationEvaluaton`.

The second component of this work considers the application of information-theoretic approaches to sequential data, a frequently occurring problem when natural language data is used to infer information flows between sources. This data is readily available on online social networks, and applying these measures can yield insights into their structures. In Chapter 4, information-theoretic measures are applied to a data set collected from Twitter to understand dynamics between automated and human-like accounts in the conversation around the 2022 Russian invasion of Ukraine. These measures are extended to consider information flows between groups of accounts which exhibit similar behaviours, rather than only considering information flows between pairs of users. Unlike other work which considers information flows in online social networks, we develop an empirical significance test to assess the significance of these pairwise flows without a ground truth or well-defined null distribution. Existing work fails to verify the performance of these estimators on realistic data sets where the true entropy value is known, so the final component of this thesis derives the theoretical entropy of a more meaningful simulation model. The Linear Additive Markov Process (LAMP) (Kumar et al., 2017) model has a well-defined long-term dependency structure, and in Chapter 5 we derive a theoretical result for its entropy rate. Open source code is provided for fitting the LAMP model to data and using it to calculate entropy estimates at `https://github.com/bridget-smart/LAMPEntropyEstimates`. Modified code from Kumar et al. (2017) is also provided to fit the LAMP model to data at `https://github.com/bridget-smart/modified_lamp`.

To summarise, the main contributions of this work are:

- In Chapter 2, a survey of existing techniques to measure the association between pairs of point processes. Within this survey, definitions and notation are unified and clearly described in context.
- Several new techniques are proposed to fill gaps identified during the survey, and a conceptual framework is described to allow the general classes of approach to be compared and understood. Practical considerations of each general class of association measure are discussed.
- In Chapter 3, we describe a new simulation framework used to generate event time data for n distinct point processes, which can have complex relationships, self-excited behaviours and periods of inactivity. This framework can be modifed to incorporate prior knowledge which may be available, allowing for the easy generation of realistic data. Implemented in PYTHON, the code for this simulation is available at `https://github.com/bridget-smart/PointProcessSimulationEvaluaton`.
- The flexibility of this data simulation framework is used to develop an evaluation framework, which validates association measures across a wide range of relationship structures and types. This evaluation framework is flexible and can be modified to suit a particular context, by mimicking a realistic range of behaviours and improving understanding of how techniques may perform in practice. Designed to be a useful and robust tool, this evaluation framework can be easily applied to existing association measures. This is demonstrated by its application to an existing transfer entropy package implemented in JULIA.
- In Chapter 4, we extend existing information flow techniques to examine aggregated group activity and describe a new empirical significance test. This allows information flows to be understood between account groups, and for the significance of these values to be calculated.
- This work is used to establish statistical significance of information flows between accounts from a real-world Twitter data set. This data set contains Tweets relating to the 2022 Russian invasion of Ukraine, and accounts are grouped by their national lean and 'botness', as characterised by Botometer. The highest information flows are out of pro-Russian non-bot accounts. Information flows into non-bot account groups are only significant for balanced and pro-Ukraine accounts, with pro-Russian non-bot accounts only exhibiting a net outward information flow of information.
- To add context to the constructed information flow network, we consider the relationship between bot activity and overall non-bot sentiment (with $p = 0.000376$), but find this relationship is significant for both positive and negative lags, indicating there may be confounding factors.

- In Chapter 5, we derive a closed-form solution giving the entropy rate of a LAMP process, and show that the entropy rate is exactly that of the underlying Markov chain regardless of the autocorrelation.
- This result is used to construct a LAMP-based entropy estimator which is applied to four publicly available datasets to obtain entropy estimates, close to or lower than values obtained using a first-order Markov model. Estimated entropy values are 4.88 bits/symbol for the Lastfm dataset, 2.49 bits/symbol for the BrightKite dataset, 3.18 bits/symbol for the Wikispeedia dataset and 3.61 bits/symbol for the Reuters dataset.
- We explore of the impacts of forcing ergodicity in a transition matrix on entropy estimates using the LAMP model. Here, we consider two approaches, using only the largest connected component and adding an artificial fully connected node, with entropy estimates similar between each approach, finding results to be comparable.

Each part of this thesis aims to highlight the importance of validating and exploring techniques to ensure they are appropriate and meaningful in context. By implementing a simulation and evaluation framework, we improve the accessibility and robustness of methods for measuring pairwise relationships between point processes. For non-parametric information flow estimators, we implement an empirical significance test and derive a theoretical entropy for the LAMP model, which can be used in future work to produce a meaningful empirical bias correction for non-parametric entropy estimators and evaluate other methods on data containing long-range dependencies.

In Chapter 6 we conclude with a summary of our contributions and provide an outlook for future work.

Chapter 2

Survey of techniques to measure pairwise similarity in point processes

2.1 Motivation

Association between point processes is important in fields including social sciences (Ng and Carley, 2022; Sharma et al., 2021; Graham, 2020; Park et al., 2021), economics (Cartea and Jaimungal, 2013; Bauwens and Hautsch, 2009; Sancetta, 2017; Wang et al., 2021), geography (Reinhart, 2018; Ogata, 1998; Musmeci and Vere-Jones, 1992) and neuroscience (Tognoli et al., 2020; Bressler and Kelso, 2016; Quaglio et al., 2018), where the relationships between signals are critical to detecting, classifying and understanding systems. Applications include detecting malicious bot accounts online, predicting earthquakes, capturing the relationships between products on the financial market and understanding what neural spikes tell us about the brain's activity. Techniques to measure this association often appear in extremely domain-specific contexts. This work serves to survey these techniques in general terms, aggregate techniques to ensure they are accessible across domains and outline a conceptual framework to assist researchers in selecting appropriate methods.

This work will focus on measuring association between pairs of point processes, and many of the techniques which will be discussed can be extended beyond this pairwise relationship to capture the network structure of the system. Methods which inherently infer the underlying network will not be considered in this survey, and are an area for future work.

In practice, researchers often select a domain-specific technique, as there does not exist a universal, robust and dynamic method to evaluate various approaches. Often notation and definitions are not consistent between fields, as there are minimal interactions between

researchers in different fields, despite very similar goals. Techniques may have been derived from a practical need which may lead to techniques which are not underpinned with the relevant mathematical framework or unknowingly duplicate the behaviour of other approaches. This missing mathematical framework can lead to:

- mathematically equivalent techniques being presented distinctly,
- no possible method to evaluate statistical significance,
- approaches which are inaccessible to other applications,
- no clear understanding of which behaviours can be attributed to idiosyncrasies of an approach, and
- techniques which are not rigorously verified.

These challenges were discussed for cognitive neuroscience in 1967 when Perkel et al. (1967b) recognised the need for a mathematical framework due to diverging approaches of measuring relationships within and between spike trains.

Surprisingly, there are very few examples of survey papers which span a range of fields and multiple classes of techniques, and existing surveys only contain a subset of existing techniques often with a single context in mind (Cliff et al., 2022; Hino et al., 2015).

These existing works focus on presenting a range of techniques, rather than considering conceptual similarities and themes. This work aims to fill this gap with a comprehensive survey, unified notation and comparison framework to improve the ability of researchers to determine the suitability of a given approach for a new context. Being able to aggregate, compare and apply techniques from other fields is critical to ensure the best similarity measure can be selected. Alongside this survey, a simulation and evaluation framework is presented in Chapter 3, which will allow the performance of techniques to be easily rigorously validated in a flexible and customised data setting.

2.1.1 Scope

The scope of this review is broad, but the focus is on providing a framework to unify and consider existing work. This review will begin by describing the challenge posed by measuring association and briefly describing the broad classes of approaches which exist. Following this, a general framework will be described which can be used to conceptually compare approaches, providing context for researchers who may be familiar with a subset of techniques.

We then explore types of association in further detail, outlining the spectrum of behaviours and including examples to provide further context. A notation scheme is proposed, and a series of definitions are given which allow a wide range of existing techniques to be

described and discussed. The goal of this unified notation is to provide a medium for relationships within existing work to be meaningfully explored, and encourage future work to use universal language to describe this broad class of technique.

To give context and provide a starting point for comparison between existing techniques using the proposed conceptual framework and notation, we consider and compare existing techniques. These existing techniques are divided into one of four categories:

- heuristic,
- information-theoretic,
- stochastic, and
- machine learning based.

These categories comprehensively cover existing techniques and allow similarities and differences to be easily identified. A representative selection of techniques with each of these sub-categories will be discussed, alongside their general benefits and drawbacks to provide specific detail within each category. These examples have been chosen to emphasise defining features of each class and represent the spectrum of existing techniques. This chapter will conclude by outlining practical considerations and possible directions for future work.

Many sources were considered when developing this framework and conducting this survey. However the broad scope of this survey, covering a range of contexts and categories implicitly limit the number of techniques which can be considered. Many definitions and techniques have originated from practical need, so a focus was placed on papers which are pivotal within their respective fields and have lead to a number of derivative works, or survey papers which characterise a number of approaches.

Within this work, we present techniques to measure association from works published between 1967 and 2022, however we draw from older works when describing and deriving mathematically rigorous frameworks and definitions. As machine learning is a quickly evolving field, a higher importance was placed on newer papers, with works chosen to allow the general drawbacks and advantages of the class of techniques to be clearly discussed.

2.2 Conceptual Framework

In this review, we consider metric, information-theoretic, stochastic and machine learning approaches. Each of these approaches has particular advantages and disadvantages which are discussed in more detail within each section (Section 2.6, Section 2.7, Section 2.8 and Section 2.9). General considerations are discussed in Section 2.10, and provide further detail than is given alongside this conceptual framework.

When selecting an approach for measuring association between point processes there are a number of factors which need to be considered including:

- physical constraints of the system,
- which features the measure is sensitive to,
- what a result means in context,
- data size,
- scale over which association occurs,
- practical limitations of implementation, and
- desired modelling outcome.

A clear definition of the behaviour of interest between point processes is useful in model selection and an understanding of the desired modelling outcomes is necessary to ensure results are useful and meaningful.

In this work we argue that association can be characterised by a combination of event focused and generating process focused analysis. The first category of approach considers the direct relationship between events within the pair of processes, aiming to measure some feature of these relationships. The second type does not consider individual events, using an explicit or implicit model of behaviour to describe how a process produces events, and aims to measure the relationship between these models for pairs of processes. The differences between these two types of association is further discussed in detail in Section 2.5.1.

Each approach has its own advantages and disadvantages and a brief overview of some considerations associated with each class of method is given for context.

1-2. Metric and information-theoretic approaches are often computationally efficient and summarise a pairwise relationship with a single value or function over time. Underlying each of these metric approaches are a set of assumptions underpinned by a null model. Quaglio et al. (2018) provide an overview of these null models for detecting population level coordination for parallel spike train data.

3. Stochastic approaches can be extremely useful when the underlying model is aligned with the physical system. Once a model is fit, these approaches can give information about how processes respond to each other and their own history, and which systems may be governing when an event occurs. This additional information comes at the cost of assumptions in model specification and difficulty in model fitting.

4. Machine learning approaches can achieve very high accuracy, especially on large data sets. The similarity between the neural network framework and the structure of the coordination network may make these models a natural choice, particularly within computational neuroscience (Rosenfeld et al., 2022; Wang et al., 2022). However, machine learning models need to be carefully designed, as the architecture of the model constrains which data is considered and how it can contribute. A common criticism of machine learning approaches is their black-box nature, which may give the impression of an overly expressive model despite design limitations.

The framework shown in Figure 2.1 assesses approaches on a scale between event-focused, where relationships between events is the target, and generating process-focused, where approaches strive to consider and compare point processes as a general process. On the y-axis, we consider how interpretable the models are. Interpretability describes how well a human can understand the behaviour of an approach from an explanation of how it works. This is similar to the concept of 'explainability' in artificial intelligence (Linardatos et al., 2020; Merry et al., 2021; Simkute et al., 2021; Vilone and Longo, 2021). For the purposes of this work, we will consider how meaningful the process taken by each approach to get a result is and if that process can provide additional insights into the system.

The x-axis in Figure 2.1, describes whether an approach is aiming to capture similarity between the data itself or a process model of the data. This describes whether the method uses relationships between events, or uses event times to estimate features of the generating process which are then compared.

The behaviour which an approach is designed to target does not necessarily directly correlate to the behaviour which the method responds most strongly to. For example estimating and comparing event rates between processes would be a method which approximates a stochastic model to estimate the generating process, but would also produce high association measures for a pair of processes which exhibit strong direct coordination. A robust mathematical underpinning for each approach helps mitigate these issues, and helps quantify which behaviours an approach will be sensitive to in practice.

An approach which places a higher importance on event times is not necessarily the same as an approach which measures similarity between event sequences. An overview of the behaviour which is targeted in practice by each approach is summarised in Figure 2.2. This comparison is based on the findings within this work. Additional methods to evaluate and understand target behaviours is explored further in Chapter 3.

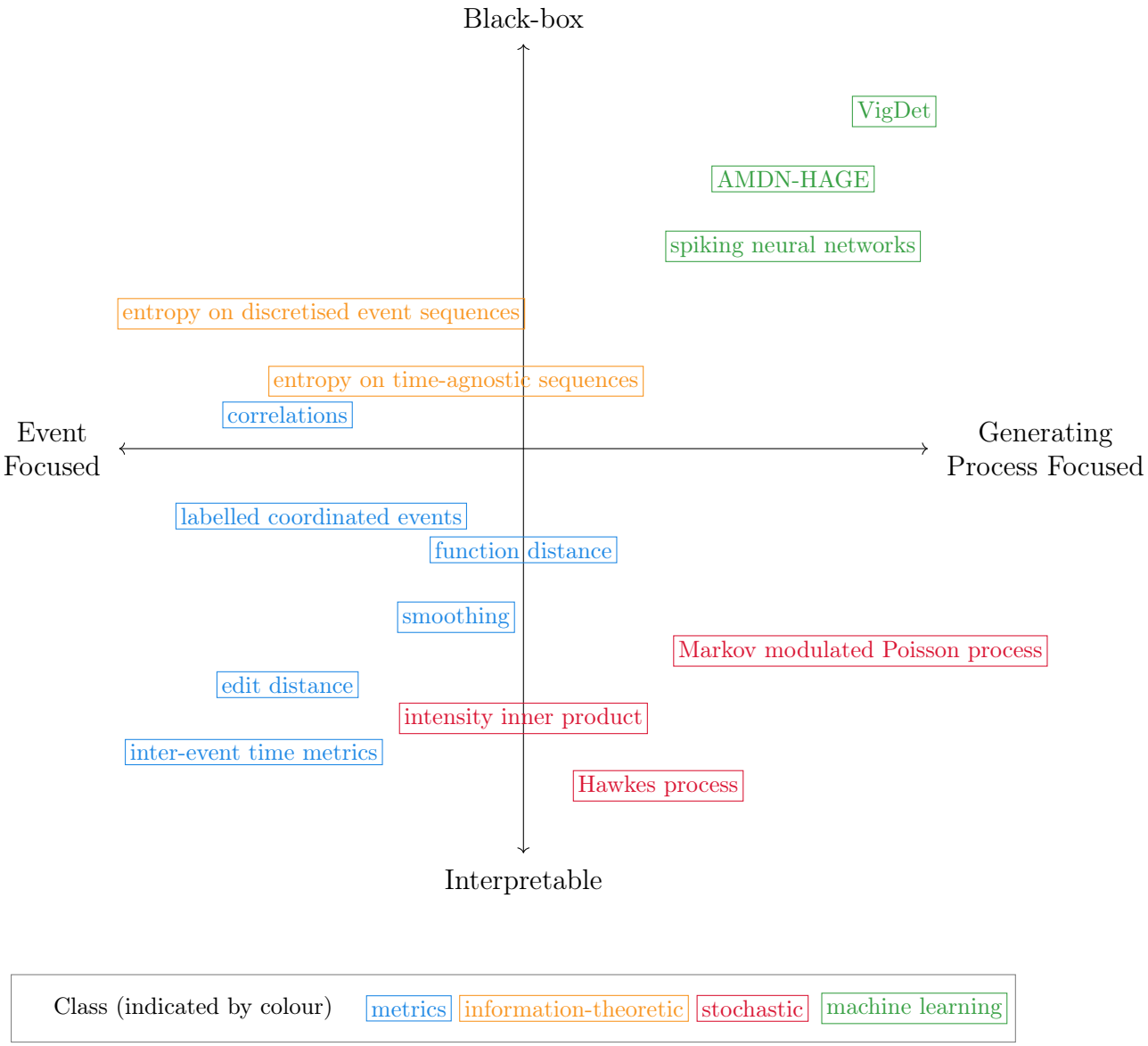


Figure 2.1: Figure showing a broad classification of several of the approaches outlined in this review, comparing them on how explainable the approach is, and whether the method has an event- or process-level focus.

These two frameworks are intended as a conceptual guide, to best frame the advantages and disadvantages of these general approach types. We do not include all possible approaches on these figures and reccommend the use of the simulation framework described in Chapter 3 for evaluation of specific techniques.

Figure 2.2: Comparison of the target behaviour of a subset of techniques. This aims to describe which types of behaviour will be most strongly detected using each approach.

2.3 Related work

Many techniques to evaluate associations between pairs of point processes have evolved from many different problems according to a practical need within experimental domains. As the body of literature within these fields has grown, these techniques have been further tailored to their specific contexts.

Work which contributes to this body of knowledge often proposes a new technique, extends an existing technique, implements a subset of approaches or aims to survey existing approaches within a particular field. This results in a body of work which has a primary focus on the experimental outcomes of approaches, rather than on improving the availability of knowledge and linking concepts in a useful manner.

With a pair of companion papers in 1967, Perkel et al. (1967a) broke this pattern, when they recognised how a lack of rigorous mathematical framework was leading to a number of issues within cognitive neuroscience. They proposed using existing stochastic frameworks to underpin the analysis of single spike trains (Perkel et al., 1967b) and the relationship between simultaneous spike trains (Perkel et al., 1967a).

In this survey, we aim to follow a similar path, focusing on conceptual themes to build a universal and robust language to compare techniques between fields. The need for this work is derived from a similar problem to that identified by Perkel et al. (1967a). The ability to theoretically understand and analyse approaches has far reaching practical

implications and helps drive advances.

While there is a large body of domain-specific literature describing particular association measures, there are only a few works which seek to aggregate this knowledge across disciplines. In this section we give an overview of existing survey papers and describe a number of common applications for measuring association between point processes. Details of association measures are given in the relevant sections.

2.3.1 Previous surveys

We will give an overview of existing survey papers to motivate the need for a comprehensive, up to date survey of existing methods.

In their survey paper, Cliff et al. (2022) present 249 statistics for pairwise interactions across 6 distinct categories. The presented measures are classified as basic, distance similarity, causal indices, information theory, spectral or miscellaneous. Each of these metrics are implemented in their corresponding extendable Python-based toolkit, `pyspi`. While their discussion covers many metrics, it does not cover methods which aim to infer the generative process using stochastic or machine learning based techniques.

The authors organise the techniques using hierarchical clustering, to describe how similar the outcomes from each technique when they are applied to over 1000 sample data sets. The clustering reveals similarities between groups of methods. One example is the group formed by directed information flows, Granger causality and integrated information, all techniques which condition on the history of a process to measure past dependency between processes. While this indicates that their clustering presents meaningful results, it is not clear how their modules should be used by researchers looking to select a metric for a particular data application.

One existing package MMPP, implemented in R by Hino et al. (2015) focuses on applications within neuroscience, providing implementations for a range of metric based techniques to determine similarity between point processes. In the package, they implement functions for co-occurrence metrics, filter-based metrics, inter-event time metrics and intensity inner product metrics which are discussed in Section 2.6. This package focuses on implementations, rather than comparison, however it offers a function to evaluate performance by calculating a kernel function to a distance between points.

There is a body of work which collates techniques for measuring association between spike trains. An overview of metrics and their applications in neuroscience is given in Victor and Purpura (1997). Quaglio et al. (2018) present a number of correlation and entropy based measures of synchronicity, and describe a number of patterns of interest. The authors outline cutting edge approaches within their field, and their suitability for detecting behaviours of interest. Park et al. (2013) give an overview of kernel based

methods. Each of these works focus on application and implementation, however they each consider the advantages and disadvantages of each possible method.

Similar works exist for other applications, for example for Hawkes processes in social media (Rizoiu et al., 2017), temporal and stochastic models in finance (Bauwens and Hautsch, 2009) and spatio-temporal stochastic models for earthquake modelling (Reinhart, 2018; Ogata, 1998).

The work will consider stochastic, information-theoretic and machine learning approaches in addition to heuristic approaches. Proposing a general notation and mathematical definitions we will consider the rich library of existing methods. We will go beyond the scope of existing literature and provide a framework for direct comparison between methods, consider what existing approaches aim to measure and how what they are measuring differs and define types of pairwise similarity between point processes. Our contribution will help researchers more clearly define their problem, understand features of their data, and reach a clear understanding on which methods are appropriate and useful for their goals.

Alongside a review of current methods, we also provide a framework to assist in the choice of approach (Figure 2.1). This will be supplemented by a simulation based evaluation package in Chapter 3.

2.4 Applications

Detecting similarity or coordination between point processes is a problem which frequently arises in many real-world applications, where time-based signals can be easily measured, but underlying relationships are difficult to probe. In this section, a selection of applications will be discussed in general terms to give context to concepts discussed in this survey. It is important to note that the measurement of the relationship between pairs of point processes is not only limited to these applications and appears in many fields.

Within each application the scale of the problem can vary drastically. In social media applications, it is not unusual to consider many thousands of users (Waller and Anderson, 2021; Sayyadiharikandeh et al., 2020; Ver Steeg and Galstyan, 2012), while in cognitive neuroscience it is more typical to consider hundreds (Victor, 2005; Shorten et al., 2021). The scale of the system affects the required speed of algorithms and the sensitivity required. Smaller scale systems have lower computational demands which often allows stronger modelling assumptions to be applied.

Cognitive neuroscience

Measuring the similarity between point processes allows researchers to understand and build network models of neurons where it may only be possible to capture time series data of potential spikes. In cognitive neuroscience, researchers measure electrical impulses to understand coordination dynamics within the brain to understand cortical function within and between cortical regions (Tognoli et al., 2020; Bressler and Kelso, 2016).

This body of literature is well developed, with work ranging from metrics which use thresholds to establish and measure coordination, to more complicated models which use stochastic methods to approximate the underlying processes and dependencies which cause a spike to occur. Quaglio et al. (2018) gives a good overview of correlation-based measures to establish relationships between parallel spike chains. They also present a thorough description of spatio-temporal patterns which occur in the field, with a focus on directly coordinated events.

Since these spikes may be caused by features which are not captured within the dataset, some approaches in this field aim to measure the amount of information which a process contains about another, or model how the generation processes impact each other. Since data may be collected from a series of locations across a neuronal system, some approaches including those based in Hawkes processes, use single process with events being labeled by their location of origin (Tognoli et al., 2020; Quaglio et al., 2018).

Detecting malicious actors online

Malicious actors in online social networks often coordinate to push narratives which may be deceiving or untrue but which aim to benefit an organisation, country, individual or cause. These campaigns often coordinate to take advantage of the mechanics behind social media algorithms, where large amounts of activity around a particular topic are rewarded, and can cause that topic to spread wider and faster across the network. This ability for information to go 'viral' is used by these malicious organisations to share and spread these narratives, and other strategies are used to target this information and construct narratives which will resonate with their chosen audience. These campaigns can be comprised of automated (*bot*) accounts or human-run accounts, often organised via *troll factories*, where paid actors operate in unison for a particular cause. A well-known troll factory is the Internet Research Agency, an organisation based in Russia which has been active in a range of topics, from interfering in protests and politics in the United States to spreading anti-Ukrainian sentiment (Duskaeva and Konyaeva, 2016; Wilde and Sherman, 2022; Bastos and Farkas, 2019; Doroshenko and Lukito, 2021).

Due to coordinated activity being rewarded by popular social media algorithms, researchers in this field often work to measure how coordinated activity is on an event-by-event level, to measure how coordinated a pair of accounts are. By measuring the

coordination between pairs of events, the overall coordination between a pair of users is quantified, and can be used to predict if an account is a malicious account which forms part of one of these coordinated campaigns (Graham et al., 2020; Giglietto et al., 2020; Keller et al., 2020; Schäfer et al., 2017).

Seismic activity

Temporal-spatial point process models are commonly used in the measurement, modelling and detection of seismic activity. After an earthquake occurs, it is common for aftershock sequences, or smaller earthquakes to follow in the surrounding region. This activity is characterised by the clustering and self-exciting nature of subsequent seismic activity. Often researchers use self-exciting stochastic models, and it is common for these sequences to be modelled using Poisson processes or Hawkes processes which allow self-exciting processes to be modelled (Reinhart, 2018; Ogata, 1998). Similar models are used to model crime (Park et al., 2021).

To incorporate spacial and temporal data into models, Hawkes processes are extended to self-exciting spatio-temporal point processes. These models can generate events with a time and spacial component and have a self-exciting generation process which is conditional on the history of the process, making them ideal for modelling seismic activity sequences (Musmeci and Vere-Jones, 1992; Ogata, 1998).

Financial data

Temporal point processes are used to predict and measure interactions between financial products, for example modelling stock, asset or index prices and their reactions to external events or events of other financial products. One example is the work performed by Cartea and Jaimungal (2013), which capitalises on the availability of data associated with high-frequency trades to model asset prices across a day, using model insights to shape high-frequency trading strategy.

Accurate and fast models can be very lucrative, and various stochastic model types are employed in an attempt to better capture and predict price changes (Bauwens and Hautsch, 2009). These models aim to understand evolving relationships in real-time, so stochastic methods able to capture conditional intensity dependencies are often employed (Sancetta, 2017; Wang et al., 2021).

2.5 Notation

2.5.1 Association between time series

As the problem space differs between fields, the similarity which researchers aim to capture also differs. On one end of the spectrum, the aim is to measure *direct coordination* between pairs of point processes, where events in a target process are versions of those in a source process, with some time delay. On the other end, the direct relationship between events is not important, and the aim is to measure how similar the sequences are by determining the relationship between their generating processes. We denote this *generating similarity*. Depending on the application, it may be desirable to measure a combination of these two behaviours or consider processes which may only sometimes coordinate.

It is useful to think about the various ways which two point processes may be interacting to define types of similarity. Consider two point processes, generated by processes X and Y. Let the event times from the two processes be given by $\{x_1, ..., x_{n_x}\}$ and $\{y_1, ..., y_{n_y}\}$ respectively.

Given this pair of point processes, there are several values we will consider when defining the similarity measures. The *inter-event* time involves matching up pairs of events. Often it is proposed that there exists a mapping between sequences where events x_i have a corresponding matching event y_{m_i} for a subset $i \in M \subset \{1, ..., n_x\}$. Then the inter-event times,

$$\delta_i^{(X,Y)} = x_i - y_{m_i},\ \forall\ i \in M.$$

Definition 2.5.1 (Inter-event times)**.** *The inter-event times for target process X with source process Y, $\delta_i^{(X,Y)}$, will be the set of all $\delta_i^{(X,Y)} = x_i - y_{m_i},\ \forall\ i \in M$.*

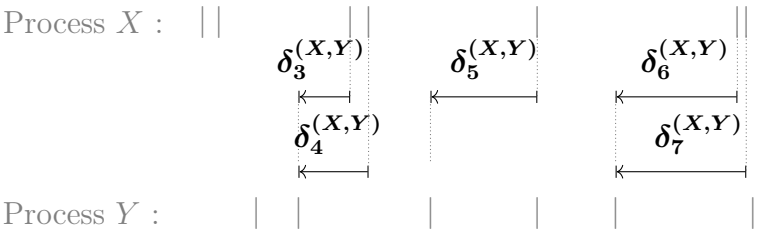
In practice, these inter-event times are often inferred by taking the most recent event time, where

$$m_i = \underset{k}{\operatorname{argmax}}\, y_k < x_i.$$

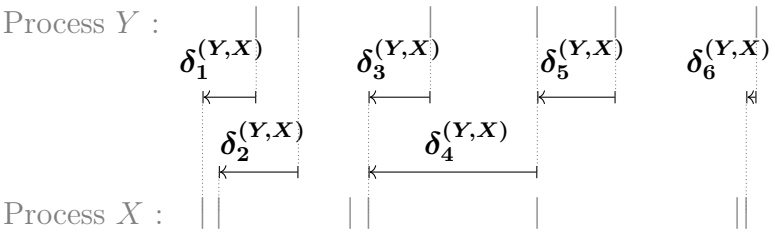
Throughout this work, we use this mapping to derive the inter-event times. It is important to note that the set of inter-event times is dependent on the direction of relationship which is considered (Figure 2.3 (i), (ii)).

Some approaches also consider similarity between the sets of event times, without considering a relationship between them. Some of these approaches use the intra-event times from within a single process, or the inter-event times for process X with process X (Figure 2.3 (iii)). For clarity, these will be referred to as the intra-event times.

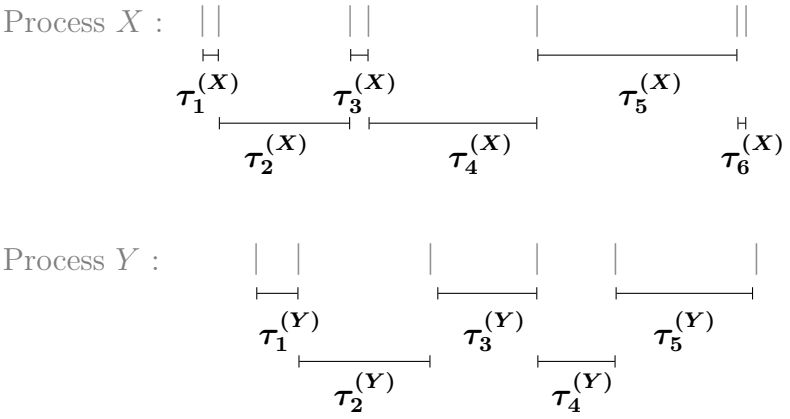
(i) Inter-event times from $X \to Y$



(ii) Inter-event times from $Y \to X$



(iii) Intra-event times



(iv) Time-agnostic :

A A B B A A B (AB) B A A B,

where (AB) represents simultaneous events, A an event in process X and B and event in process Y.

Figure 2.3: Overview of information types used to measure similarity and coordination in point processes.

Definition 2.5.2 (Intra-event times)**.** *The intra-event times for process X, is the set of all time intervals, τ, between events within process X. Let*

$$\tau_i^{(X)} = x_i - x_{i-1}$$

for $i \in [2, n_X]$, where n_x is the total number of events in processs X.

A third type of information is the order of event (Figure 2.3 (iv)). This time agnostic approach does not use the times of the events, but only considers the originating process and the event order. Such an approach can reveal behaviours in settings where the times of events are less important than the causal relationships between processes.

An example of each of these three information types is given in Figure 2.3.

In the first case, if process Y was **directly coordinating** with process X, we would expect to see a strong relationship between x_i and y_{m_i}. We denote X to be the source or parent process and Y to be the target or child process (Figure 2.4). We define direct coordination to occur when all events in process Y are triggered by events in process X, and have times $x_i = y_{m_i} + \delta$, for δ from some distribution (Definition 2.5.3). In data, this may present as a short time delay between an event in the source process and an event in the target process. It is not necessary that the dependent events occur in the same order as their parent process, as the time delay between the pair of events (x_i, y_{m_i}) may be larger than the delay for the subsequent pair of events, $x_i + \delta_i > x_{i+1} + \delta_{i+1}$.

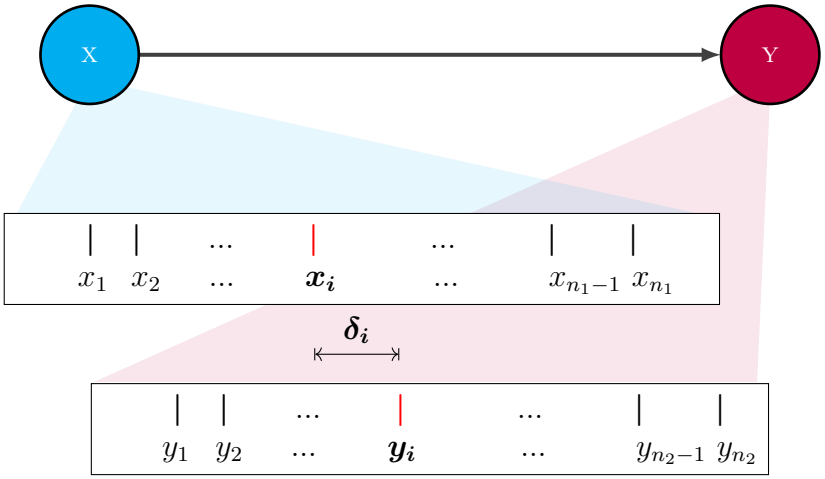


Figure 2.4: Directly coordinating processes. Here, X generates events which Y directly responds to. The time delay between the two dependent events is denoted δ_i.

Definition 2.5.3 (Direct Coordination)**.** *Direct coordination between two temporal point processes occurs when an event in process Y is triggered by a set of events in process X,*

and the relationship between the two events obeys the relationship,

$$x_i = y_{m_i} + \delta,$$

for δ from some distribution and for some mapping between i and m_i.

A looser definition of **coordination** does not require there to be a short time delay between pairs of coordinated events. The probability of an event in process Y may increase if an event occurs in process X. This leads to events which are coordinated, but we do not always see an event pair (x_i, y_i). This looser definition of coordination is useful and echoes the approach taken by stochastic methods, where we consider how events in the history of one process impact the probability we see an event in another process.

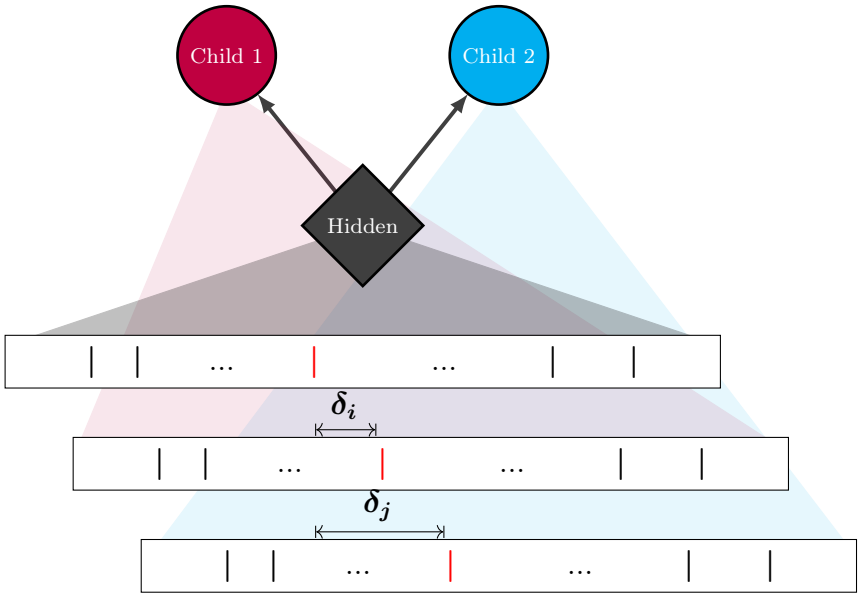


Figure 2.5: Indirectly coordinating processes. Here, the parent process generates events which both child processes respond directly to. The time delay between the two dependent events is denoted δ_i and δ_j.

Indirect coordination, depicted in Figure 2.5, occurs when a hidden parent process is coordinating with two dependent processes, which are observable (Definition 2.5.4). The two dependent processes will exhibit similar event time patterns, and there may be behaviour which appears similar to direct coordination if there is a mechanism which causes the events in one process to have longer delays than the other process. This may occur when observable event times are effected by a factor external to the measured processes.

Definition 2.5.4 (Generating process similarity)**.** *Generating process similarity between point processes is used to refer to similarity between the underlying generating processes. For a process with a known generating distribution, if another process generates events from the same or a similar distribution, we say these two processes are similar.*

Each of these types of coordination result in events occurring in the target process which are delayed versions of events in the source process. Both types of coordination describe a direct relationship between pairs of coordinated events.

While coordination describes a relationship between corresponding events, it can also be useful to capture the **generating similarity** between the generating process behind pairs of point processes. When a process is generating events from a known distribution, if another process generates events from the same distribution, there is a similarity between the event times of the two processes. This generating process similarity requires a metric which captures information about the generating distribution rather than considering dependence on event times in an alternative process.

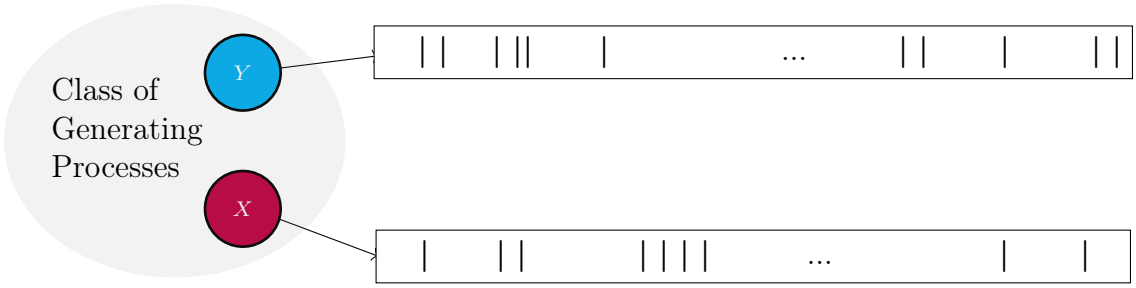


Figure 2.6: Processes with similar generating functions.

In a real-world setting, for example in malicious online campaigns, agents may be directed by a principal to spread a particular narrative and may each work toward this goal independently. Since each agent has a similar goal and environmental conditions, for example working hours or performance metrics, the time stamps of their activity may exhibit similar patterns, but there may not be a deliberate coordination of activity.

While **coordination** is concerned with the relationship between event times with a pair or processes, **similarity** describes processes comprised of event times which come from similar generating processes.

2.5.2 Mathematical details

A point process is a random process consisting of points in a space. Here we consider temporal point processes, a set of events which occur in continuous time where the state space is the nonnegative real numbers. We define a temporal point process as the following:

Definition 2.5.5 (Temporal Point Processes)**.** *Temporal point processes are random processes consisting of points in the non-negative real numbers, representing a collection of events in continuous time. For a process, X with n_x realisations representing event times, we let $x_i, i \in [1, n_x]$ represent the time of the i^{th} event.*

While event times are labelled according to their originating process, within this work several approaches use continuous time. Throughout this work, we will use t to denote a variable representing continuous time.

These event times can be considered as continuous random variables, or a discrete variables, where time is broken up into discrete chunks, with the size of a single time unit determined by a choice of hyperparameter. A discrete framework can be useful in reducing the computational complexity for some approaches. This discrete framework also makes sense when events can only occur at discrete times due to a physical constraint in the measurement or generating process, such as limited measurement precision.

When considering the similarity or coordination between pairs of point processes in real-world datasets, what these temporal point processes represent can differ. For a *simple point process*, a single set of times will represent events from a single source. Online, we might model the activity of each account as a temporal point process representing the times which the account shared a post or engaged with content online. In seismology, each earthquake or event can be represented as an instance drawn from a temporal point processes.

In some contexts distinct point processes are represented by a single process, in others a single point process is used to represent all processes in a system, with a label associated with each point within the point process to distinguish events from various originating processes. When these labels are present, this is referred to as a *marked point process*.

These two alternative ways of formulating what a point process looks like can alter the scope of the techniques which can be applied (Figure 2.7).

There is scope for a combination of these formulations to be used to consider inter-group activities, with all agents from a single group considered as a single point process, allowing the inter-group similarity to be considered, rather than considering the processes on a single agent level.

For the purposes of comparison, we will focus on the case where distinct point processes are used for each distinct generating process or actor. The second formulation will briefly be discussed within Section 2.8 Stochastic Approaches, although is an area for future work.

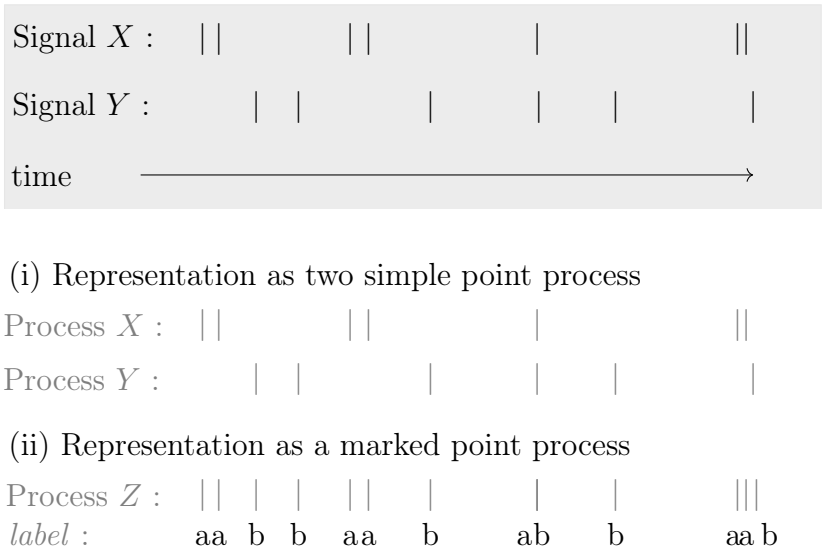


Figure 2.7: Demonstration of the same set of events from sources X and Y represented as a pair of simple point processes (i) and as a marked point process (ii). The labels for the marked point process are shown underneath the process, and this additional information is encoded into the point process. While containing equivalent information, one of these two frameworks can be more useful in particular approaches, so it is useful to keep this alternative construction in mind.

Behind each of these point process is a generating process. Outside of simulation, these processes are not known, and it can be the goal of modelling to understand these. The generating process generates events within the temporal point process. Where an explicit definition is not given, throughout this work the generating process for process X will be denoted $G^{(X)}$.

2.5.3 Stochastic processes

The following concepts are used in many stochastic methods used to model and understand event time data. These concepts underpin a number of measures of association between pairs of point processes. With event time data, alongside notions of inter- and intra-event times, the number of event occurrences is also of interest. A *counting process* is used to capture this.

Definition 2.5.6 (Counting process)**.** *A counting process, denoted $N(t)$, is a stochastic process used to describe the number of events which have been observed at a given time, t. This process can only take integer values, must be non-negative and monotonically non-decreasing. We have that the number of events between two times, $t_1 < t_2$, is given by $N(t_2) - N(t_1)$.*

Intensity Function

The intensity function of a stochastic point process describes the probability of an event occurring, and characterises the point process. Often, intensity functions are conditional on the history of a process, and the most common definition for an intensity function in literature uses the limit as $h \to 0$ of the probability of a single event occurring in a time window with width h (Definition 2.5.7). The intensity function is usually denoted $\lambda(t)$.

Definition 2.5.7 (Intensity Function)**.** *The intensity function of a point process is defined as,*

$$\lambda(t) = \lim_{h \to 0} \frac{\mathbb{P}(N(t+h) - N(t) = 1)}{h}.$$

When this function is conditionally dependent on the history of the process, this definition changes slightly (Definition 2.5.8).

Definition 2.5.8 (Conditional Intensity Function)**.** *The conditional intensity function of a point process at time t_i, is defined as,*

$$\lambda(t|t_1, ..., t_{i-1}) = \lim_{h \to 0} \frac{\mathbb{P}(N(t+h) - N(t) = 1|t_1, ..., t_{i-1})}{h},$$

where $t_1, ..., t_{i-1}$ are the events which occur before time t.

Memoryless property

For event times, the memoryless property describes when the intra-event time is only dependent on the most recent event time, and does not depend on any other previous events. While many real-world systems do not exhibit the memoryless property, it is a useful approximation as it allows well-defined distributions to be applied, generating many useful results in understanding random processes.

Poisson point process

A Poisson point process is a counting process which depends only on the current state (Definition 2.5.9 below). These random processes are meaningful models often used to model event occurrences in many settings, and they are equipped with a number of theoretical results.

Definition 2.5.9 (Poisson point process)**.** *A counting process, $N(t)$ for $t \in [0, \infty)$, with fixed intensity λ is a Poisson process with rate λ, if*

1. *$N(0) = 0$,*
2. *$N(t)$ depends only on the current state, or that each increment is independent, and*
3. *the events are independent and identically distributed, and the number of events in an interval of length w is a Poisson random variable with mean $w\lambda$.*

2.6 Heuristic approaches

Heuristics are used to describe the class of measures which describes the association between a pair of processes with a single value, without strictly requiring the function to be underpinned by a rigorous theory. This class of approach includes metrics, but also includes functions which do not satisfy all of the requirements to be a metric. Approaches which do not satisfy the definition of a metric are frequently used in literature and are motivated by intuitive or informal parallels drawn between the relevant context and mathematical theory.

Metrics are functions which produce a single value which describe the distance or similarity between objects.

Definition 2.6.1 (Metric)**.** *A metric is a non-negative real-valued function, sometimes referred to as a distance function, $D : X \times X \to \mathbb{R}$, which satisfies the following three conditions:*

(i) D must always be positive, except for the trivial case, i.e. $D(X, Y) = 0 \iff X = Y$,
(ii) D must be symmetric, i.e. $D(X, Y) = D(Y, X)$,
(iii) D must satisfy the triangle inequality, i.e. $D(X, Z) \leq D(X, Y) + D(Y, Z)$.

Many functions have been defined to to measure pairs of point processes, both with labelled and unlabelled events (Hino et al., 2015). Some of these measures do not satisfy conditions (i) - (iii) above, so are not formally defined as a metric, but are still a real-valued function which can be used to summarise a set of event times, inter-event times, intra-event times or a sequence using a single real value. We refer to such measures as heuristics.

In this work, we will outline heuristic and metric approaches in general terms, with a few particular examples explored. These examples have been chosen to highlight the range of approaches, and allow the advantages and disadvantages of approach types to be discussed.

Our work provides a comprehensive description of general classes of similarity functions

for marked point processes, providing several examples within each class.

When considering pairs of point processes, there are several types of information available, summarised in Section 2.5.1 and depicted in Figure 2.3. Considering event times directly allows inter-event times, event times and delays to be used to measure features of the event times within a single process, or to consider the relationship between two processes. A time-agnostic approach does not use event times and considers only the order which they occur. In a pairwise setting, this allows a sequence of event order to be generated, which can reveal behaviours in settings where the times of events are less important than the causal relationships between processes.

2.6.1 Correlation measures

Since the work of Spearman (1904) on measuring the association between two variables or datasets, correlation measures have become an integral statistical tool (Pearson, 1920). The Spearman and Pearson correlations are used extensively in statistical analysis to measure the strength of linear relationship between two datasets. This application has been extended to measure the association between point processes, with a body of literature focusing on spike-train classifications.

Directly calculating the correlations is not meaningful, as the data contained within a point process represents a set of event times which must be monotonically increasing. This will result in a positive correlation for all pairs of time series containing sufficient points. In this work, we consider two methods to overcome this problem; discretising event times and approximating processes with continuous functions.

Discretising event times

One approach uses a time window to discretise event times, producing an event frequency for each distinct time bin (Figure 2.10). These frequencies are used to construct estimates of the probabilities of an event occurring within a particular time bin. The correlation is calculated using the joint probability of these frequency probability estimates by assuming independence under the null hypothesis. This assumption is tested by assuming some null distribution over the event times, for example a Binomial or Poisson distribution (Schneidman et al., 2003; Grün et al., 2002; Osborne et al., 2008). This technique is sometimes called *peri-stimulus time histogram analysis* and is common in cognitive neuroscience (Jurjuţ et al., 2019; Garcia-Garcia et al., 2020; Schreiber et al., 2003).

While this simple approach is easily understood and communicated, results are highly variable with respect to the choice of binning parameter. There is also an implicit assumption about the distribution of event times for independent processes which may not be true across applications.

Mapping to continuous functions

A similar technique maps the sets of event times to a continuous function. This is constructed by associating a spike with each event and smoothing these event spikes using an continuous function, resulting in a smoothed function as shown in Figure 2.8. This is typically chosen to be a exponential function (van Rossum, 2001).

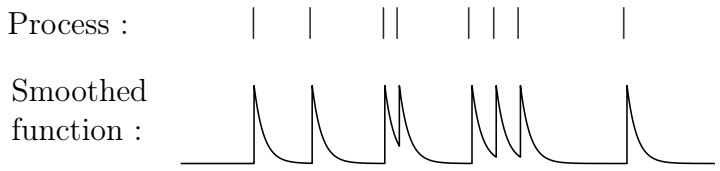


Figure 2.8: Smoothed function

A common formulation of the mapping from discrete event times to a real-valued continuous function of continuous time, for process X for some set of event times $\{x_1..., x_{n_x}\}$. Here, t is used to denote continuous time to avoid confusion. This mapping is given by

$$f_x(t) = \frac{1}{n_x} \sum_{i=1}^{n_x} h_\tau(t - x_i) \cdot u(t - x_i) \tag{2.1}$$

where the choice of $h_\tau = \exp(-|t|/\tau)/(2\tau)$, corresponds to Laplacian smoothing (Borrajo et al., 2022; Hino et al., 2015). The function $u(\cdot)$ is the Heaviside step function, given by

$$u(t) = \begin{cases} 1, & t \geq 0 \\ 0, & t < 0. \end{cases}$$

By smoothing these event times, binning has less of an effect and the correlation measures tend to be more stable (Schreiber et al., 2003).

Using this approach Schreiber et al. (2003) proposed the following correlation measure between processes X and Y:

$$\text{cor}(x, y) = \frac{\int_0^\infty f_x(t) f_y(t) \mathrm{d}t}{\sqrt{\int_0^\infty f_x(t)^2 \mathrm{d}t} \sqrt{\int_0^\infty f_y(t)^2 \mathrm{d}t}}. \tag{2.2}$$

Using this correlation produces a more stable similarity estimate over time histogram analysis techniques while only relying on a single parameter, τ. It also allows the information contained within an event to be sustained over some period, as determined by the smoothing parameter.

Other approaches use alternatives to measure the similarity or distance between the smoothed functions (van Rossum, 2001).

Expressing sets of event times in these ways allow associations between point processes to be explored at various time scales. The histogram approach uses more parameters than the smoothing approach, but results are highly dependent on arbitrary choices made to group events or produce continuous functions when these methods are applied. It is necessary for these parameters to be carefully chosen to produce sensible results, which often requires intimate domain knowledge, for example when instantaneous measurements are taken, but the effects are known to follow a decay structure.

Fundamentally, this approach does not associate event pairs so cannot be used to measure the presence of direct coordination between a pair of processes. Discretising or continuously approximating event times removes exact information about when an event has occurred, leading to an approach which tends to compare event frequencies. In applications where it is ideal to consider how close rates are between sets of event times, and we can make meaningful choices for the parameters when producing smoothing functions or binning event times these approaches may be meaningful, although their use should be carefully considered.

2.6.2 Distance metrics

Typical distance measures, for example Euclidean distances, are not applicable to temporal point processes. A temporal point process contains a variable-length list of ordered event times, which does not translate to a meaningful embedding in Euclidean space. There is imposed structure due to monotonically increasing event times.

While not meaningful when applied to the raw event times, measures based on Euclidean distance or inner products can be applied to transformed data, where a function or vector captures information about the behaviour of the process, and a distance between these representations is considered.

Generally, with distance based functions, it is important to carefully consider what the embedding space represents in context. When realisations on a point process are transformed, the transformed points represent some function of the original event times. This transformation likely changes the contextual meaning of the relationship between these event times, and may obscure or highlight features in the data.

Edit distance

Edit distance is defined as the number of edits, typically counting the number of insertions and deletions, required to transform one sequence into another. Edit distance uses a non-Euclidean metric to calculate a distance and is used in applications including spike train metrics, modelling foreign currencies and account classification on social media (Victor and Purpura, 1997; Victor, 2005; Suzuki et al., 2010; Kim et al., 2019). Considering time as a discrete variable makes it possible to calculate this edit distance between sequences, although this can be computationally intensive for higher resolution or larger data sets.

Kim et al. (2019) extends this by proposing the *time-sensitive semantic edit distance (t-SED)*. This metric, designed to classify accounts in online networks, uses the idea that account activity is defined by digital traces described by a feature vector containing both content and timing. These digital traces are characterised by the content and time of each tweet a user produces. Their proposed measure weights the edit distance between content embeddings using exponentially weighted inter-event times, to calculate a time-sensitive distance metric. Kim et al. (2019) use their measure to quantitatively measure semantic and temporal similarity between accounts, applying this metric to distinguish between left and right leaning trolls in a Russian troll data set. t-SED uses the weights of traditional edit distance to down weight content with greater temporal distance or smaller word similarity, weighted by exponentially penalised time delays.

Edit distances assume that the similarity between two processes is determined by how congruent they are, and do not aim to capture coordinated behaviour types. This class of approaches penalise each operation or type of operation equally, and do not allow for some events to be classed as more important than others.

Distance between continuous functions

It is possible to evaluate the distances between continuous functions, which represent sets of event times. See Section 2.6.1, for an example of using an exponential decay to map discrete event times to a continuous-valued function.

Implicitly, these approaches assume that the continuous function represents the set of event times corresponding to the generating process. This assumption may not be as strong as directly evaluating distance measures on the set of event times itself as information is lost in the continuous approximation. However, when the choice of continuous function improves the quality of the representation of the generating function of the point process, using the continuous function may provide a superior measure for the similarity between the pair of generating processes. This can occur when the continuous function is meaningful, and adds information, for example when data points are missing but a model is used to infer missing values.

Rudin (1991) provides a comprehensive discussion of distance in function spaces. Measures which operate over the space of all continuous real-valued functions in a Euclidean space $\mathbb{R}^n$ are appropriate. One approach is to consider the norm between two functions, which provides a measure analogous to distance in euclidean space, where smaller values indicate more similarity between these functions.

One challenge posed by this approach is the indirect relationship between a metric in function space and the physical features of the generating processes.

2.6.3 Inter-event time metrics

One straightforward approach to measure the coordination between a pair of point processes is to directly consider the inter-event times between a pair of processes. This class of metrics aims to measure the time between corresponding events in the two processes. Some approaches consider statistics on these inter-event times. Others use these inter-event times combined with a threshold to determine if a pair of corresponding events is a coordinated pair of events or not.

Penalised time delays

Transformations applied to the set of inter-event times allows for a prior importance to be applied to the length of the delay between events. This function, $d(\cdot)$, can be set up to reward shorter inter-event times and penalise longer delays.

The advantage of particular statistics and choices of $d(\cdot)$ are that they can be chosen to capture a particular type or length of inter-event time, and careful choice of statistic can make these measure robust to outliers or other confounding factors within the data. Conversely, this freedom makes it possible for a metric to be constructed which does not measure the intended aspect of the relationship between the set of event times, giving a result which may falsely indicate a particular behaviour is occurring when it is not. The simulation framework described in Chapter 3, allows the constructed measures to be evaluated against simulated data, giving insights into the behaviour being captured.

Choosing $d(\cdot)$ to exponentially penalise the time delay produces a measure which affords a larger importance to shorter time delays. This approach is useful when pairs of events with a short time delay need to be given a larger weighting. An example is $d(\delta) = b \exp\{a\delta\}$, for some constants a and b.

2.6.4 Time coordination windows

In bot detection on social media, it is typical to use inter-event times to determine a *coordination window*, or time period within which a pair of events is considered to be coordinated. The choice of this threshold function has a large effect on the results which

are produced. This time window should be selected by an domain-knowledge expert, or through the use of uses labelled data sets to establish an understanding of a reasonable choice of coordination window. Alternatively, some works use a statistical method to establish a threshold value (Giglietto et al., 2020; Graham et al., 2020; Keller et al., 2020; Schäfer et al., 2017).

Time windows can also be set by knowledge of the physical constraints of the system. If a limiting factor prevents a coordinated event to occur until a particular time period has passed, for example if a signal needs to travel from process A to process B, or there is a activation time for B to react, it is not possible to observe a coordinating event until adequate time has passed. In such a system, processes only have the ability to react to events in the past with some predetermined delay, so a pair of coordinated events which occur in a shorter time window must only be occurring together by chance. In systems where randomness is small, and events are mostly deterministic, this type of approach is suitable.

Graham (2020), produced the PYTHON package `coordination-network-toolkit`, designed for use with Twitter data to classify events as coordinated or not, provided the respective accounts produce similar or congruent content within a predetermined time window. A single coordinated event between two accounts leads to their relationship being classed as 'coordinated'. This straightforward approach is particularly useful on social media data sets, where the size of the datasets and computation times pose challenges.

2.6.5 Labelling coordinated events

Similar to time coordination windows, the process of classifying coordinated events enables researchers to clearly define the behaviour of interest. These labels can be used to filter data, or statistics on these labels can be used to quickly summarise the behaviour of a process.

Producing a label involves moving from a continuous value to a discrete label, which involves many decisions regarding features of interest of these pairwise behaviours. Once each event is labelled, aggregating these labels provides an opportunity to further refine or develop the classification mechanism, making labelling a tempting solution when the true classification of some processes is known.

Event labels are often time-homogeneous, with inter-event time being the only aspect considered (Graham et al., 2020; Keller et al., 2020; Schäfer et al., 2017). It is typical for the spread of the coordinated pairwise events to have no bearing on the label given to the pairwise relationship between the two processes, which is typically also determined by a threshold (Gansel and Singer, 2012; Ng and Carley, 2022; Graham et al., 2020). When this approach is used, the details are highly context dependent.

One approach is to count the number or proportion of coordinated events between two processes, labelling the pair of processes as 'coordinated' once this value exceeds a threshold (Palm et al., 1988; van Rossum, 2001). In Graham et al. (2020) and Keller et al. (2020), Twitter accounts retweeting within a coordination-window of one second more than once are determined to be 'co-tweeting' accounts. In real-world applications where true relationships are impossible to know, the choice of coordination window and threshold on the number of coordinated events is difficult to determine.

Ideally, it would be possible to construct a null distribution to ascribe some probability that a pair of processes are coordinated conditional on the number of coordinated events, which would allow for a coordination label to be produced with a predetermined uncertainty. It can be difficult to construct a meaningful null distribution, although this method is discussed in the context of backbone networks in Section 2.10.3.

Thresholds

Time coordination windows can be used to label coordinated events, implemented by thresholding inter-event times. This can be meaningful when the threshold value represents a distinguishing value or physical constraint, however its suitability can be limited when inter-event times follow a power law distribution. Using a threshold is commonplace in detecting coordination, where two events are said to be coordinated if they occur within some small time window of each other. A threshold can represent a relative value, for example a value 80% larger than all other values, or it may represent a distinguishing value, for example a time delay which exactly corresponds to a particular type of response event.

While thresholds can be used to ignore events which occur in time windows where coordination is not possible, potentially due to physical constraints, it is more common for thresholds to be used to mask values which are larger than the critical value. There are several types of thresholds, each with their own advantages and disadvantages.

Generally, thresholds produce a very transparent process, where values which are meaningfully chosen can clearly define and capture particular types of behaviour. For example, if considering coordination between a pair of radio transmitters and receivers, only considering events which occur after the shortest possible response time and before the longest, guarantees that only events which could be coordinated are considered.

The simplicity of thresholds makes them simple to implement, and the effects of a threshold are clear. They can be efficiently applied to large datasets and there is often a direct relationship between choices of critical values and event labels.

We will now describe three types of thresholds; hard thresholds, soft thresholds and adaptive thresholds. Each of these will be presented as a threshold designed to ignore

smaller values and retain larger values, although the discussion applies to both types.

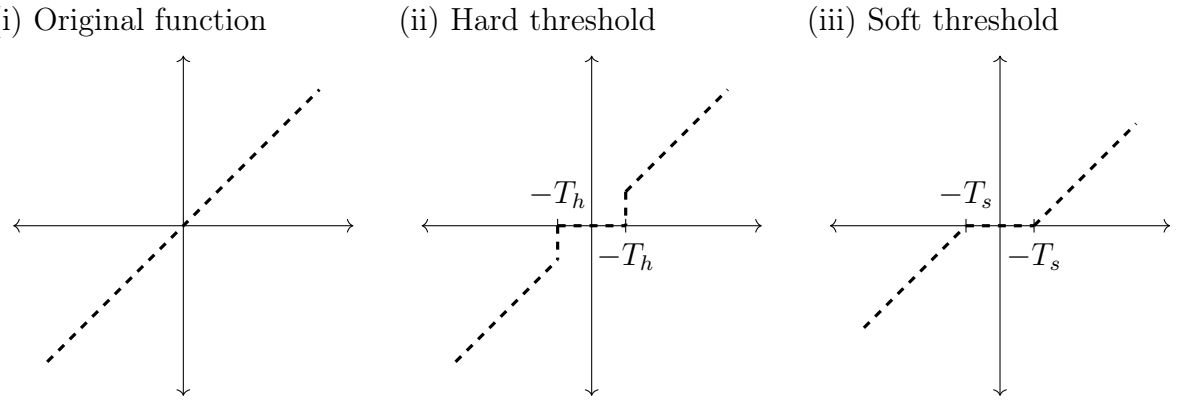


Figure 2.9: Hard and soft thresholds.

A hard threshold masks all values which lie above the threshold value, T_h. This can be expressed by the function:

$$T_h(\delta) = \begin{cases} \delta, & \text{if } \delta \geq T_h \\ 0, & \text{if } \delta < T_h. \end{cases}$$

Note that by construction, δ is always strictly greater than zero, so it is not necessary to consider the absolute value. A soft threshold masks all values which lie below the threshold value, T_s and shrinks those which lie above it toward zero . This can be expressed by the function:

$$T_s(\delta) = \begin{cases} \delta - T_s, & \text{if } \delta \geq T_s \\ 0, & \text{if } \delta < T_s. \end{cases}$$

Both hard and soft thresholds result in a subset of values being masked. Unmasked values different between the two approaches, with soft thresholds shifting the remaining δ toward zero (Figure 2.9). The resulting inter-event times from a hard threshold approach have a range which starts from T_h. This will impose ranges on statistics, for example the mean δ of 'coordinated' events cannot be smaller than T_h. Soft thresholding shifts these values toward zero, so mitigates these issues.

Since the choice of critical value in both hard and soft thresholding is time and process invariant, the same value is applied to all events from any process in the system. In systems where different processes may result in different physical features and constraints, it may be ideal to have value which varies depending on the process and events in question. One approach to this is adaptive thresholding.

Adaptive thresholding uses a threshold value which is calculated locally. Examples of adaptive thresholds consider a function of previous inter- and intra-event times to calculate a dynamic threshold value for determining the window of coordination.

An example of an adpative threshold, proposed by Quian Quiroga et al. (2002), aims to determine if x_i is a coordinated event with y_j. The threshold is given by,

$$T_a = \frac{1}{2}\min\left\{\tau_i^{(X)}, \tau_{i-i}^{(X)}, \tau_j^{(Y)}, \tau_{j-1}^{(Y)}\right\},$$

where $\tau_i^{(X)}$ is the intra-event time given by $x_{i+1} - x_i$.

This corresponds to an event which occurs in half the time of the four intra-event intervals which surround the corresponding pair. This measure of synchronisation uses information about the timings within processes, helping account for processes which contain events generated with different rates. This features makes an adaptive approach attractive in cases where some processes may include many events within a short time frame, providing a simple means to overcome this challenge.

2.7 Information-theoretic approaches

In many contexts, we are interested in understanding the temporal relationship between point processes, and how information flows between them. Information theory provides a framework to measure various aspects of this relationship including mutual information and information flows. These flows can reveal how much information one set of event times reveals about another and can provide a meaningful value that represents the pairwise relationship.

Information theory was born from the work of Shannon (1948), who worked to understand and model how information is communicated. His work introduced Shannon entropy, a measure of information that has applications in many fields, including compression and measuring relationships between probability distributions (Cover and Thomas, 1991). Shannon's original estimator has been extended and exploited to understand relationships between point processes by many researchers (Shorten et al., 2021; Mijatovic et al., 2021; Melnik and Usatenko, 2016; Ameli and Sassi, 2018; Spinney et al., 2017).

For a random variable, the *entropy* provides a measure of how much information it contains, or alternatively, how unpredictable or uncertain that variable is. Often, entropy is calculated using the random variable's probability distribution (Definition 2.7.1).

Definition 2.7.1 (Self Entropy)**.** *For a random variable, X, with probability distribution*

$p(x)$, the self-entropy is given by

$$H(X) = -\sum_x p(x) \log_2 p(x),$$

with the convention that $0 \log 0 = 0$.

The logarithm in this measure makes it more intuitive, as it forces the information contained in two identical processes to be double that of each process individually. The base of this logarithm alters the units which the information is measured in. It is common to use a logarithm with base 2, which produces information measured in bits. By measuring information, the values produced using these methods can be extremely meaningful in context, unlike some of the other approaches discussed.

Here, we are concerned with relationships between pairs of point processes, so it is natural to consider entropy measures which consider flows of information, like cross or transfer entropy. By considering the joint and conditional distributions of pairs of variables, we are able to define the joint (Definition 2.7.2), conditional (Definition 2.7.3) and cross (Definition 2.7.4) entropies for random variables X and Y.

Definition 2.7.2 (Joint Entropy)**.** *For random variables X and Y, the joint entropy is given by*

$$H(X, Y) = -\sum_x \sum_y p(x, y) \log_2 p(x, y),$$

where $p(x, y)$ is the joint probability distribution of X and Y.

Definition 2.7.3 (Conditional Entropy)**.** *For random variables X and Y, the conditional entropy is given by*

$$H(X|Y) = -\sum_y \sum_x p(x, y) \log_2 p(x|y).$$

Definition 2.7.4 (Cross Entropy)**.** *For random variables X and Y, the cross entropy is given by*

$$H(X||Y) = -\sum_x p(x) \log_2 q(y),$$

where $p(x)$ and $q(x)$ are the probability distributions of X and Y respectively.

While the above definitions provide entropy estimates using probability distributions, the entropy rate is also well-defined for sequences. The entropy rate describes the average information per element of a sequence. This is a natural extension of the concept of total entropy for sequences, where it may be useful to know how the entropy changes as the length of the sequence grows. As the entropy rate gives the information per unit, this measure can be more easily compared between sequences of variable length.

In practice, it is often desirable to calculate the entropy of a sequence where the probability distribution is intractable or difficult to obtain. To overcome this challenge, Wyner and Ziv (1989) propose a non-parametric entropy rate estimator using match lengths. This estimator takes advantage of the general properties of samples from a stochastic sequence, described in the independent and identically distributed (i.i.d.) case by the asymptotic equipartition theorem.

When continuous time measurements are discredited to a binary sequence where 1 represents the presence of an event, and 0 the absence, discrtee information-theoretic measures are applicable. For a discrete random process, X, which takes values from a finite set of symbols, let X_a^b denote the string $(X_a, X_{a+1}, ..., X_b)$. For such a process, the Kontoyiannis temporal entropy rate estimator (Kontoyiannis et al., 1998) is given by,

$$H(X) = \lim_{n\to\infty} \frac{n \log n}{\sum_{i=1}^{n} \Lambda_i^l},$$

where Λ_i^l denotes the length of the shortest substring, l starting at index i which does not appear in the first $i + l - 1$ symbols.

Self entropy was extended in work by Ziv and Merhav (1993) to measure relative entropy between a pair of sequences. The relative entropy estimator proposed by Ziv and Merhav (1993) was modified to calculate cross entropy by Bagrow et al. (2019) for a fiexd window size. The cross-entropy between two sources describes the amount of information required to describe the next symbol in one source given the history of the other. The Kontoyiannis estimator, is better able to capture long-term dependencies within the data than frequency based estimators. It can be extended to measure the cross entropy between two sources (Ziv and Merhav, 1993; Bagrow et al., 2019). To do so, it uses the history of one sequence and the future of the other to gain insight into the quantity of information shared between the two processes over time.

This time-synced cross-entropy rate estimator between a target, $\mathcal{T}$, and source, $\mathcal{S}$ at an instant i in time is given by,

$$h(\mathcal{T}||\mathcal{S}) = \frac{N_{\mathcal{T}} \log_2 N_{\mathcal{S}}}{\sum_{i=1}^{N_{\mathcal{T}}} \Lambda_i(\mathcal{T}|\mathcal{S}_{\leq t(T_i)})}, \tag{2.3}$$

Alternatively, for a set of event times, the *transfer entropy* can be used to estimate the amount of information for some target process, Y, which is contained in the history of a source process X conditioned upon the history of the target process (Definition 2.7.5).

Definition 2.7.5 (Transfer Entropy)**.** *For random variables X and Y, the transfer entropy rate at time t is given by*

$$T_{X\to Y} = H(Y_t|Y_{t-1:t-L}) - H(Y_t|Y_{t-1:t-L}, X_{t-1:t-L}),$$

where L gives the length of the history which is considered and $X_{g,h}$ denotes the process X between times h and g.

Transfer entropy is frequently described in terms of mutual information, which measures how much uncertainty is reduced in one variable with full knowledge of another (Definition 2.7.6). Independent variables will have a mutual information of 0, and fully dependent variables will have mutual information equal to the self-entropy of either variable, which will be equal. The transfer entropy is equal to the mutual information conditional on the history of the target process and the source process.

Definition 2.7.6 (Mutual Information)**.** *For random variables X and Y, the mutual information is given by*

$$I(X;Y) = \sum_x \sum_y p(x,y) \log_2 \frac{p(x,y)}{p(x)p(y)}.$$

2.7.1 Dealing with real data

Entropy estimates can be formed either indirectly using an estimate of the probability density, or directly using the sequences themselves. This leads to three distinct approaches; discrete time bins, estimating probability densities and time agnostic sequence generation. The preferred approach usually dependent on context and the available data.

For natural language data, including post content from online social media platforms, non-parametric estimators can produce an entropy estimate which accounts for long-term dependencies in the sequence over frequency-derived probability estimators, without the need for complex large models to explicitly capture this behaviour (South et al., 2022).

While non-parametric sequence-based estimators can be intuitively applied to discrete sequences, point processes often represent continuous event times. Moving from continuous time to discrete data results in a loss of information, as the resolution of each data point is reduced. We will describe several approaches to formulate entropy estimates from pairs of point processes, however other approaches are possible. An overview is shown in Figure 2.3.

1. Discrete time bins

One approach is to use discrete-time bins to generate a sequence, with values representing the number of event occurrences within each bin. When a bin size is chosen it is used to divide up the total observation time into a number of intervals, each with width equal to the chosen bin size. Within each of these intervals, the number of events within each interval is recorded and event time data is discarded. An example is given in Figure 2.10.

With a small enough time bin, each bin can contain at most one event and these sequences can use a binary value to indicate the presence or absence of an event.

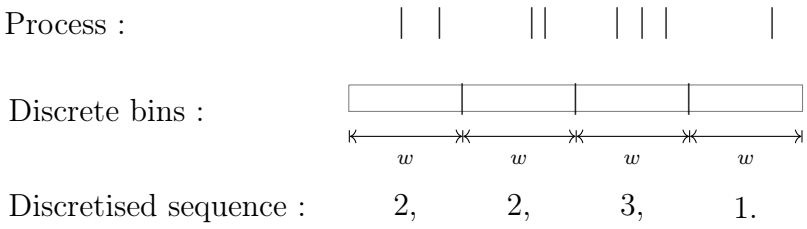


Figure 2.10: Process of discretising a sequence with a window size of w.

Results can be biased and extremely sensitive to the width of the time bin, so information is lost when a single continuous time realisation is represented by an interval. When multiple events can occur within a single bin, we do not retain information about possible within-bin intra-event relationships or event orderings. Conversely, choosing an extremely small time bin may lead to sparse sequences and computational challenges as sequences may become extremely long (Feutrill and Roughan, 2021).

2. Estimating probability densities

It is possible to estimate the generating probability density functions for the given sequence using a variety of model-fitting approaches. Frequentist approaches use data to obtain parameter estimates for a model, and are commonly used to formulate entropy estimates. Bayesian settings allow for prior information to be included in the inference, and may be more appropriate where there is limited data.

The problem of model fitting to sequence-based data is well studied (Kass and Ventura, 2001; Perkel et al., 1967a; Kershenbaum et al., 2021; Shorten et al., 2021). Models which result in a well-defined probability distribution typically allow the entropy estimates to be easily obtained from analytic results. Techniques including drop-out and cross-validation can improve convergence between the fit model and the true generating distribution, further improving these estimates. Extensive work has been done in this area (Kontoyiannis et al., 1998; South et al., 2022; Bagrow et al., 2019; Shorten et al., 2021), and will not be discussed in detail here.

Markov models are often used to model sequential data, and they are typically used to represent the state of a system across time. Markov chains have well-defined entropy rates, allowing the entropy to be easily estimated when the underlying chain is well-conditioned. Typically, a first-order Markov model can only account for first-order dependencies in a sequence, so higher-order models may be considered for more complex sequences. Higher order Markov models have a polynomial parameter space, so alternative formulations like

the Linear Additive Markov Process (LAMP) model can capture long-range dependencies while retaining the simple formulation for the entropy estimate of a first-order Markov chain (Kumar et al., 2017). This is further discussed in Chapter 5.

3. Sequences of events, a time-agnostic approach

When event times are not available or useful, we can use event order to understand relationships between processes. When measurements are unreliable, this approach can remove data uncertainty from the model. In cases where the delay between events is extremely varied or unpredictable, the event times can be used to generate a time-agnostic sequence of events, labelled by the originating process. Shown in Figure 2.3, this approach can be easily extended beyond the pairwise formulation, and uses non-parametric entropy estimates to measure the predictability of the sequence.

This time-agnostic approach is undirected as the self entropy of a binary sequence is unaffected by the choice of symbols.

2.7.2 Choice of entropy estimator

In the literature, it is more common to use mutual information or transfer entropy to analyse the relationships between point processes than cross entropy estimators. For pairwise comparison, the transfer and cross-entropy estimators both produce a directed measure of information flow, which may also be interpreted as a measure of how predictable one sequence is given the other. More explicitly, the cross entropy, $H(X, Y)$, describes how much surprise there is for an event in process X, given we know the history of a process Y. Transfer entropy, $T_{X \to Y}$, from process Y to process X describes how much surprise there is for an event in process X given we know the history of Y conditional on X. Using this conditional relationship more directly measures the causal relationship between the pairs of point processes.

Hence, transfer entropy produces more meaningful estimates when applied to sequences representing event occurrences, formed through discrete binning, where this conditional distribution can reveal information about causal pairwise relationships (Shorten et al., 2021). Transfer entropy is commonly used in neuroscience, and has also been used to understand earthquakes, social behaviour and financial products (Shorten et al., 2021; Novelli et al., 2019; Liu et al., 2022; Ver Steeg and Galstyan, 2012).

The cross-entropy can be useful as it does not use the history of the target process, and can be applied in settings where data may be incomplete. Cross-entropy estimates may be more useful as a tool to measure the distance between generating distributions, either through sequential estimation or application to estimated distribution models.

2.7.3 Examples in the literature

Entropy measures are frequently used in cognitive neuroscience, with the work of Quaglio et al. (2018) providing a comprehensive overview of several metric and information-theoretic measures. Their work uses entropy as a measure for the likelihood of particular spike patterns and estimated correlations. This work uses binning, which can lead to issues with bias.

A continuous time transfer entropy is given by Mijatovic et al. (2021). In their work, they use a Bayesian inversion to estimate the probability density of events within a process using the measured data. This method uses the convergence between a discretised process and its continuous analog in the limit. This method is constrained by the rate of convergence in the limit, and may be subject to the biases associated with discrete time bins.

An alternative estimator is proposed by Shorten et al. (2021), to estimate the transfer entropy from event time data using a k-nearest neighbours-based estimator. This overcomes the bias issues which discrete time bins present. The authors also describe a procedure for empirical significance testing which accounts for conditional independence. The focus of this work is in the neuroscience setting, where additional challenges are posed by constraints in the experimental set-up. Electrodes used to collect spike data are subject to limited sampling rates and spatial resolution. In this setting, correlations between event times and event sequences which are tied to neurological processes occur on time scales which vary by two orders of magnitude, making discrete bins unsuitable to capture both types of behaviour. Rather than using a discrete time approximation, Shorten et al. (2021) build on the work of Spinney et al. (2017) to propose their continuous time transfer entropy estimator, given in Equation 2.4.

$$T_{X\to Y} = \lim_{t\to\infty} \frac{1}{T} \sum_{i=1}^{n_y} \ln \frac{\lambda_{y|x_{<t},y_{<t},z_{<t}}[x_{<x_i}, y_{<x_i}, z_{<x_i}]}{\lambda_{y|y_{<t},z_{<t}}[y_{<x_i}, z_{<x_i}]}, \tag{2.4}$$

where T is the length of time considered; $\lambda_{y|x_{<t},y_{<t},z_{<t}}[x_{<x_i}, y_{<x_i}, z_{<x_i}]$ is the instantaneous firing rate of the target process, Y, conditional on the histories of the source process, X, and any other conditioning processes, represented by Z. $\lambda_{y|y_{<t},z_{<t}}[y_{<x_i}, z_{<x_i}]$ is the instantaneous firing rate conditional on the history of the process and its conditioning processes. The firing rate describes the probability of an event in a given time interval, with respect to the size of that interval. This work overcomes many of the challenges faced by researchers using transfer entropy estimators on spike train datasets, with the added ability to consider conditioning processes.

Outside of neuroscience, transfer entropy has been applied to understand network topology, with a case study considering event times from wireless transmission signals. In the

work from Laghate and Cabric (2018), the assumptions of an autoregressive time series model are met, making Granger causality and transfer entropy equivalent measures. This equivalence allows the authors to utilise an existing non-parametric measure for transfer entropy, which uses a shortening time window to estimate the co-occurrence probability and transfer entropy for a pair of processes (Schreiber, 2000). This estimator is similar to the work of Mijatovic et al. (2021). By testing for significance of the Granger causality, the authors can reconstruct receiver and transmitter network structures, and estimate response times. Laghate and Cabric (2018) also perform sensitivity analysis by varying response probability, discrete bin size, idle time and causal lag, finding their entropy estimates increase for larger lags and response probabilities.

2.8 Stochastic approaches

This class of technique includes any approach which assumes the point processes are stochastic. Stochastic processes have a behaviour which is defined by a random variable, and it is assumed that event times are drawn from these distributions rather than coming from a deterministic process.

Stochastic approaches may overlap with some heuristic models but differ in their underlying assumptions. Rather than producing a single value to describe a pairwise relationship, stochastic approaches primarily focus on fitting a model to the observed data. These models can provide more insight into processes than a single value and can be applied beyond assessing the relationship between two point processes. Many stochastic models have been used to measure and understand event time data, for example Hawkes processes for financial (Filimonov and Sornette, 2015) or social media data (Rizoiu et al., 2017), Markov modulated Poisson processes for web traffic data (Scott and Smyth, 2003), or kernel-based intensity estimation to predict wildlife-vehicle collisions (Borrajo et al., 2022).

In the literature around modelling point processes, there is a large focus on the use of Hawkes processes. The scope of this review will reflect this, however it is important to note this is not a comprehensive review of all possible stochastic approaches but rather an illustration of advantages and disadvantages of this general class of approach.

Intensity inner product

By applying techniques which measure similarity between functions, as outlined in Section 2.6.2, we can consider the similarity between intensity functions. In the stochastic setting, the intensity function is related to the probability of an event occurring, so the similarity between these functions indicates similar event time patterns, rather than similarity between the generating processes.

Kernel methods are commonly used to assess the similarity between intensity functions

calculated using spike train data (Park et al., 2013; Paiva et al., 2009). These methods allow event time data to be mapped into a space which is equipped with a well-defined inner product. The reproducing Hilbert kernel space is commonly selected as it allows for convex optimisation and distances can be calculated with reasonable computational complexity (Park et al., 2013). The reproducing Hilbert kernel space allows distances between functions to be computed, and assigns higher similarity scores to functions which are close in value around the same points.

These approaches map the set of event times to a feature space. This feature space has a kernel, κ, which is a positive definite function which takes in two arguments. In the feature space, the set of event times, $\boldsymbol{x}$, becomes $\kappa(\boldsymbol{x}, \cdot)$. This kernel is used to measure the similarity between the two point processes. The choice of kernel is extremely important, as it can imply an infinite feature space, or not provide distinct enough mappings to produce meaningful results (Park et al., 2013).

Some research uses time bins to transform the set of event times or use a smoothing function, which can lead to loss of information and bias in the data (Chi and Margoliash, 2001; Park et al., 2013).

Paiva et al. (2009); Hino et al. (2015) describe an *inner product* measure for event times, which is based on the intensity function. This approach directly uses event times without discretisation, and is underpinned by a Gaussian smoothing transformation. The use of Gaussian smoothing allows a distance metric to be clearly defined in terms of the estimated intensity. The work of Paiva et al. (2009), derives a result for distance from intensity functions, rather than considering event co-occurrences as in Chi and Margoliash (2001); Chi et al. (2007).

For an intensity function approximation given by

$$\hat{\lambda}_{\boldsymbol{x}}(t) = \frac{1}{n} \sum_{i=1}^{n} h_\tau(t - \boldsymbol{x}_i).$$

Paiva et al. (2009) define their similarity metric (Equation 2.5) and distance metric (Equation 2.6) for the sets of event times $\boldsymbol{x}$ and $\boldsymbol{y}$:

$$\kappa(x, y) = \int_{-\infty}^{\infty} \mathrm{d}t \hat{\lambda}_{\boldsymbol{x}}(t) \hat{\lambda}_{\boldsymbol{y}}(t), \tag{2.5}$$

$$\kappa(x, y) = \int_{-\infty}^{\infty} \mathrm{d}t (\hat{\lambda}_{\boldsymbol{x}}(t) - \hat{\lambda}_{\boldsymbol{y}}(t))^2. \tag{2.6}$$

This general method can be applied to many intensity functions which come from Poisson, Hawkes or non-parametric approaches.

2.8.1 Homogeneous Poisson processes

A Poisson process is a special case of a point process. Realisations from a Poisson process consist of event times which lie on the non-negative real line. Each realisation is independent and interevent times are drawn from independent and identically distributed exponential distributions each with rate λ.

Poisson processes are *memoryless*, each event time only depends on the previous event.

When λ is constant, this Poisson point process is time-homogeneous. This process can also be non-homogeneous when the intensity of the Poisson processes is a function of time, $\lambda(t)$.

For a time-homogeneous Poisson point process, the expected intra-event time is $1/\lambda$.

A Poisson process provides a framework for describing a single point process, but does not naturally extend to pairwise comparison. One approach is to directly estimate the intensity function and use an intensity inner product, but these approaches can be equivalent to applying a continuous smoothing, as discussed in Section 2.6.

An alternative approach is to use a Markov modulated Poisson process, (MMPP). This is a non-homogeneous Poisson process, where the intensity function is described by a Markov Process. This model aims to mimic point processes which have two distinct patterns in their event times, usually a steady rate of events combined with periods of burstiness (Scott and Smyth, 2003). This framework is not limited to two distinct patterns, as the underlying Markov process can have m states, each which correspond to a different behaviour. It is possible for model parameters to be given by prior information, however when given event time data, researchers work to infer these.

Sometimes the Markov modulated Poisson process is described as the superposition of m Poisson processes, which may be active or inactive at a given time, t. Which processes are active is determined by the underlying hidden Markov model (Fischer and Meier-Hellstern, 1993; Scott and Smyth, 2003). The parameters of this model can be found using a maximum likihood estimation technique, for example using EM algorithm as described by Turin (1996), or via Bayesian inference.

This model can be advantageous when understanding the hidden Markov model provides additional insight into the data, for example when there are several possible mechanisms behind event generation and it is desirable to perform event attribution. It is also suitable when a process is not self-exciting, and event times exhibit the memoryless property.

2.8.2 Hawkes processes

Originally described by Hawkes (1971), a Hawkes process is a *self-exciting* point process with a conditional intensity function dependent on both the history of the process and a

base intensity function in the form given in Definition 2.8.1 (Figure 2.11). Events in the history of a self-exciting process cause increases in the intensity function, increasing the likelihood of a subsequent event.

We define a Hawkes process in Definition 2.8.1.

Definition 2.8.1 (Hawkes Process)**.** *A Hawkes Process is a type of point process with a conditional intensity function in the form:*

$$\lambda(t|\mathcal{H}_t) = \lambda_0(t) + \sum_{i|t_i<t} \phi(t - t_i).$$

Here, $\mathcal{H}_t$ represents the history of the process, or all events up to time t_i. The base intensity function, $\lambda_0(t)$, represents the rate at which events occur independently of each other. This quantity is often constant, but can be a function of continuous time. $\phi(\cdot)$ is the intensity function which represents self-exciting events.

Sometimes called the *excitement intensity*, $\phi(\cdot)$ is the intensity which corresponds to self-exciting events. The choice of $\phi(\cdot)$ affects how the history of the process influences the overall intensity. By choosing a monotonically non-increasing $\phi(\cdot)$, more recent events have a larger influence on the overall intensity or rate of event occurrence.

For the Hawkes process to be stable, and the average intensity to be finite, we require

$$\int_0^\infty \phi(u)du < +\infty.$$

A typical choice for $\phi(\cdot)$ is an exponential function, which means historic events have decaying importance with respect to the overall intensity. Events which occur due to $\phi(\cdot)$ are considered offspring of previous events, and $\int_0^\infty \phi(\cdot)dt$ gives the average number of offspring from a single event.

As described above, the univariate Hawkes process is used to describe a single point process. This framework can be extended to model multiple point processes, which are able to excite events in other processes (Hawkes, 1971). In this multivariate Hawkes process, a matrix representation is used to capture both self- and cross- exciting intensity functions.

Many extensions to these self-exciting processes exist, each with its own advantages and disadvantages. Once a Hawkes process has been fit to the data, it is possible to use cross-exciting relationships to develop pairwise measures, or compare similarity between intensity functions.

To fit the univariate Hawkes process to data, it is necessary to estimate parameters for the baseline intensity function, and parameters for the self-exciting intensity decay kernel.

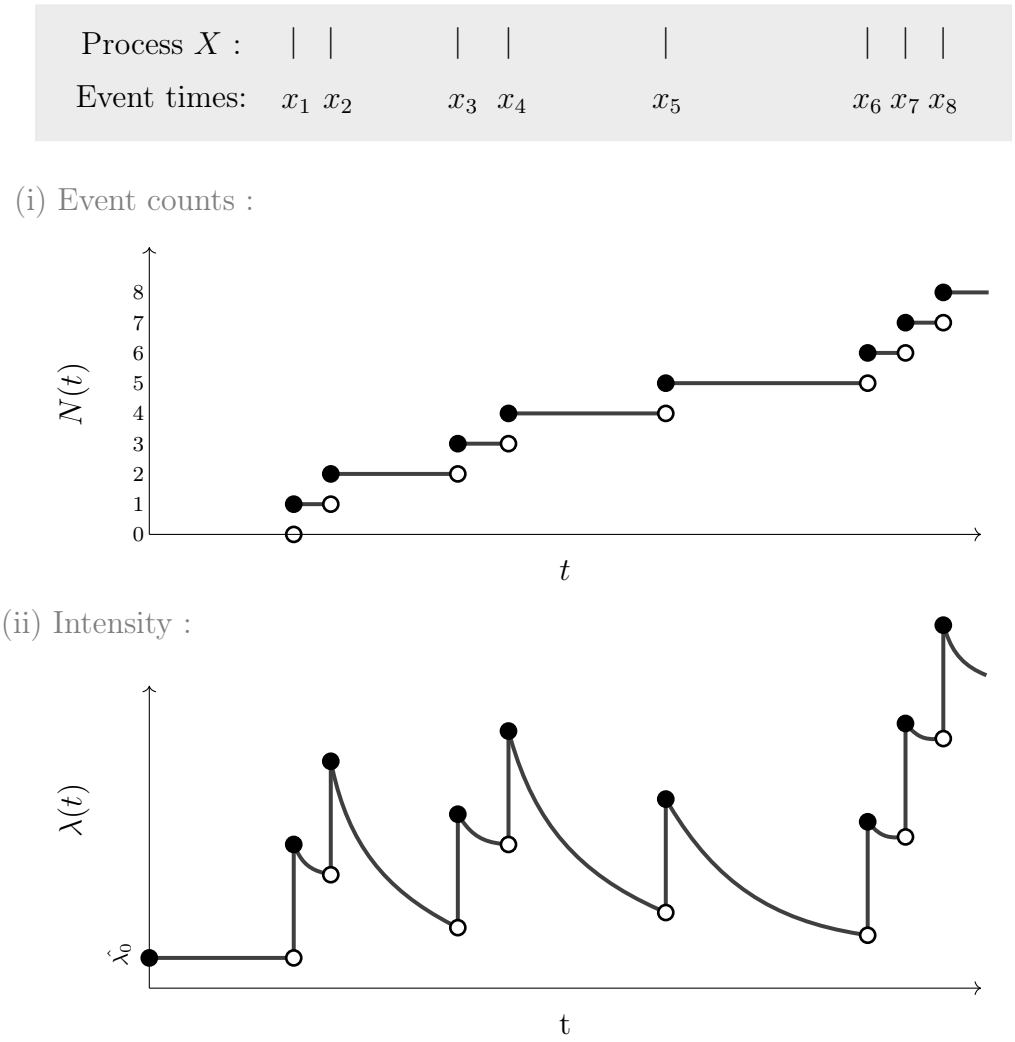


Figure 2.11: An example of the underlying Hawkes process model for the event times shown at the top. For the given process X, with events $x_1, ..., x_8$, we can represent the counting process (i), and the intensity function (ii). In this figure, an exponential function was used for the self-exciting intensity. (i) Shows the counting process, $N(t)$, where the event count increases as each event occurs. This function is strictly monotonically non-decreasing. (ii) represents the base intensity with $\hat{\lambda}_0$, the y-intercept for the intensity function. As each event occurs, there is a spike due to $\phi(\cdot)$ with this decaying exponentially until the next event occurs.

For the exponential case with a constant base intensity function, this is a total of three parameters, λ_0, a and τ, where $\phi(t) = a \exp(-\tau t)$.

The challenge in using Hawkes processes lies in parameter estimation and the careful choice of modelling assumptions. Researchers compare choices of decay functions, but when these cannot be informed by the context, it is difficult to ensure these decisions are meaningful.

Maximum likelihood estimation is a common technique for parameter estimation, with the likelihood for a set of parameters θ in the case of exponential decay is given by

$$L(\theta) = \prod_{i=1}^{n} \lambda(T_i) e^{-\int_0^{T_i} \lambda(t)dt}.$$

Maximising this likelihood is equivalent to maximising the log likelihood function. This alternative formulation removes the product, allowing terms to be summed which is less prone to precision errors. This is especially important when likelihood values come very close to 0.

Rizoiu et al. (2017) use this approach to fit a marked Hawkes point process to social media data, using an exponential kernel. Using this model allows them to postulate causal links between events, by retaining data about event offspring. A marked Hawkes process is particularly well suited to retweet cascades, which occur when a single tweet is retweeted many times and an event makes the post more likely to be seen and thus shared again. They fit this with both an exponential and power law decay function for the self-exciting intensity, using their models to predict the total number of events which a tweet will generate.

Using a maximum likelihood model, Bauwens and Hautsch (2009) fit a Hawkes process to high-frequency financial data, modifying the self-exciting intensity model to include additional variables which are theorised to influence trades.

Kirchner (2016) describe a Bayesian estimator which builds on the work of Bacry and Muzy (2016). Both of these estimators use data to generated a discretised counting process, which bins and counts events. In Kirchner (2016), these binned event counts are approximated using a time series model which allows the baseline intensity and excitement functions to be calculated. These formulations can be applied to a univariate or multivariate Hawkes process. Cai et al. (2022) use a Hawkes process model to learn directly learn causal structure between point processes, describing a sparse optimisation procedure and applying their method to synthetic data and data from a metropolitan cellular network.

Outside of parametric approaches, machine learning is also a popular tool for working with Hawkes processes. One approach, outlined in work from Zhang et al. (2020), uses the attention and self-attention mechanism (Vaswani et al., 2017). This is a machine learning tool which can be applied to sequential data, for example a set of event times, and can

be used to understand which components on the input vectors contribute most to the predictive power of the machine learning algorithm. This self-attention mechanism was originally proposed as an alternative to a neural network based model, with recurrence, gating or convolutions (Vaswani et al., 2017). It requires less parameters and produces a more explainable result. This self-attention is used to estimate the self-exciting component of the intensity function by approximating the influence of all previous events on the event at a given time. Gaussian error linear units and softplus functions are used to transform the influence vector from the self-attention mechanism with a linear embedding matrix. The softplus function is used to estimate the decay parameter which is required to be non-negative.

2.8.3 General considerations for stochastic approaches

If the primary task is to summarise a pairwise relationship between point processes, stochastic process based models can be unsuitable due to modelling assumptions, for example imposing a memoryless model onto a sequence which is characterised by long-term dependencies, or the model complexity which can be a limiting factor when fitting a model to a large number or processes. The choice of stochastic framework can have large implications for results, for example a choice of exponentially decay for a self-exciting intensity function in a Hawkes process can prevent the model from detecting causal relationships between point processes which occur over the medium or longer term scale.

The decision to use a marked point process can prevent inter-process relationships from being explicitly defined within the model, which can be extremely limiting within some applications. For example, in spike train data, not allowing inter-process event stimulation does not correspond with the hypothesis that neurons precipitate events in their neighbouring neurons.

Generally, this class of approaches can take a large amount of time to train and fit, especially for more complex models. To fit accurate models, training can require large volumes of data, which may not be available. In some cases, convex optimisation may not be possible which can pose further challenges (Kirchner, 2016).

Using stochastic models can be extremely advantageous as a correctly designed stochastic framework can give explicit models for the relationships between processes. This can provide understanding on the scale of the relationship, how the relationship propagates over time, and an offspring likelihood for particular events (Hawkes, 1971; Rizoiu et al., 2017).

2.9 Machine learning approaches

Machine learning can be a very powerful tool in pattern recognition and model fitting. These capabilities make it highly appropriate for learning generating distributions or event-time patterns which can be given a classification label. The strength of machine learning lies in its ability to extract non-linear patterns from extremely large datasets, often at the expense of human interpretability. When considering point processes, many approaches in the literature aim to take advantage of these strengths, focus on detecting patterns of behaviour of complex relationships. A large body of work looks to detect coordinated activity on social media (Luceri et al., 2020; Alothali et al., 2018; Alizadeh et al., 2020).

Machine learning models can be designed to take in data in a variety of forms, so some approaches in this space enrich event data with additional features. Here, we will focus on approaches that only use event time data, since blended approaches may not be transferable between applications and so do not fall into the scope of this review.

There is no shortage of machine learning methods in this problem-space, with approaches broadly falling into two categories. The first, classification, is used by researchers to label coordinated accounts or coordinated events (Luceri et al., 2020; Alothali et al., 2018; Alizadeh et al., 2020; Henelius et al., 2016), taking a process level approach to considering similarities. These approaches involve drawing conclusions from patterns within the data and can be supervised, semi-supervised or unsupervised. Moving away from event-level data, the second general approach strives to model the underlying generating processes. Generative models are particularly well suited to this task and are employed to produce a function which describes the generating process (Rosenfeld et al., 2022; Bodnar, 2021; Wen et al., 2021). This approach allows a direct or indirect measure of the pairwise relationship between processes. Unlike stochastic approaches, an explicit model does not need to exist for machine learning approaches, so less assumptions need to be made about the process's structure.

Similar to other classes of approach, measuring and labelling relationships between point processes can form a two-component process. Where machine learning approaches from either category result in a representation of a process or set of event times, clustering approaches are commonly used to identify clusters of processes which exhibit similar behaviours.

When well designed, machine learning approaches also reduce reliance on research or expert driven expectations of behaviour, and can reveal unexpected behaviours and patterns effectively. It is also possible to continually train these models to understand or measure how the behaviours and relationships of the processes are changing in an evolving context.

While the models used often lack explainability, it is possible for machine learning approaches to produce richer insights than metric or information-theoretic approaches, as vector representations of a number of output features can be produced, rather than a single scalar.

Fitting these models often requires significant compute power, or large volumes of training data, but approaches like transfer learning mitigate these issues. Transfer learning allows the researcher to take pre-trained existing model, fine-tuning it with a small sample of training data. Many pre-trained models are available for download, and where similar problems exist, carefully designing a machine learning approach can be both powerful and able to produce results with more reasonable data and computation requirements.

Since machine learning approaches are often aiming to identify complex patterns and hidden relationships, the quality of the training data is extremely important. For social media applications, many researchers use datasets from Twitter where accounts have been labelled (Alizadeh et al., 2020; Luceri et al., 2020; Ng and Carley, 2022; Nicolas Koumchatzky and Andryeyev, 2017). Using these labels in a supervised machine learning approaches can inhibit the validity of results. These labels may be derived from other models, and the new model may be replicating behaviour of the labelling technique, rather than providing useful insights.

Within the class of stochastic approaches discussed in Section 2.8, machine learning approaches are used alongside optimisation techniques to fit these models to large or complex datasets (Zhang et al., 2020; Hawkes, 1971).

2.9.1 Examples of machine learning approaches

The first type of approach directly considers event times, and the relationships of event times for a particular process. The general process begins by passing the set of intra-event or event times into a machine learning model which learns a representation of this input data. This representation is used to summarise the behaviour of a process, or is returned directly, where it can be used as a representation of the process.

The model AMDN-HAGE (Attentive Mixture Density Network with Hidden Account Group Estimation) proposed by Sharma et al. (2021) jointly considers influence between processes as well as labelling account groups using distance from a learned 'normal' behaviour. This model is designed for coordinated actor detection on social media. By considering a combination of population-level dynamics and individual process behaviours, this model aims to identify groups of abnormal accounts which influence the inter-group behaviour. Unlike other approaches which use a coordination window, this is learnt from the model, which considers the distribution of future event times conditioned on the history of all processes in the population.

Model Outline

1. Attentive Mixture Density Network (AMDN), aims to represent the event times, $x_0^{(s)}, ..., x_{n_x}^{(s)}$ from a single process, $X^{(s)}$, for all s in the set of processes. Here $X^{(s)}$ is used as a label to denote the originating process. It does so using two encoders:

    - *History encoder*, represents each event prior to time t_i, as a tuple containing a process label and event time, $(X^{(s)}, x_j^{(s)})$, for all $j < i$.

    - *Event encoder* predicts $(X^{(s)}, t_i)$ given the history.

2. Hidden Account Group Estimation (HAGE). This component aims to maximise distance between groups of account embeddings using a mixture of gaussian multivariate mixture distributions.

    HAGE assumes there exists a continuous distribution over the account embedding space and this representation has a meaningful real-world interpretation.

The model parameters for each of these two components are learnt, using a bilevel optimiation formulation solved iterately with the EM algorithm. The output is a maximum likelihood of the joint model, with account embeddings, activity trace model and group membership label.

Zhang et al. (2021) propose VigDet (Variational Inference for Group Detection). This model extends ADAM-HAGE, allowing prior knowledge to be included, using a coordination window to first construct a dense graph where edge weights represent a pairwise coordinated event count. The edge weights are filtered using a power function hypothesis, as described in Section 2.2. This measure is imperfect as it is highly sensititve to the choice of coordination window, and inadvertently assigns higher 'coordination' values to pairwise relationships involving processes with more event instances.

Both of these models are unsupervised, but labelled data sets provide opportunity for their performance to be assessed. The inclusion of prior information in the VigDet model contributes to improved results on a 2016 labelled Twitter set containing tweet times of accounts known to be part of the IRA's campaign around the 2016 U.S. Election but these improvements are marginal, with a maximum F1 score improvement of 0.022. Both of these approaches out perform coordination window approaches, which has an F1 score of 24.6 compared to the 93.3 and 95 of the ADAM-HAGE with k-means and VigDet models respectively (Zhang et al., 2021).

Endo et al. (2021) describe a convolutional neural network used to estimate synaptic connectivity. This neural network was trained on the spike train data from 400 synapses, and uses all event times to make a prediction about the relationship. This model, and similar models contain many thousands or hundreds of thousands of parameters, so require

large amounts of data to train and typically consider all events within a process. This makes these models unsuitable for continuous or temporally-dependent estimation.

In cognitive neuroscience, spiking neural networks are designed to mimic our understanding of the neural system, and are used to provide a comprehensive model for spike train data (Endo et al., 2021; Wang et al., 2022; Rosenfeld et al., 2022).

The model proposed by Wang et al. (2022) uses radix encoding to reduce the volume of data required over the large requirements for a more traditional neural network model. This model is particularly useful for sets of event times which occur over short time periods, offering significantly reduced computational times with slight improvements in accuracy. Radix encoding offers an alternative to using an approximation of the intensity function, giving non-uniform importance values to each event time. These values are calculated using a linear combination of events within the process's self-history and a weighted contribution of events in the history of other processes.

While this work, and other work using spiking neural networks ascribe cause of an event to a single event in the history of all processes, Rosenfeld et al. (2022) weaken this assumption, allowing distributions of events to be considered. This is achieved by enhancing the architecture with a generative adversarial network (GAN). A GAN is a module which is composed of a generator and a discriminator. The generator aims to generate samples with particular labels to fool the discriminator, which is shown both generated and true data. This module is well suited to learning distributions, and is applied to add a probabilistic aspect to the spiking neural network.

2.10 Considerations

In this survey we present four classes of approach; heuristic, information-theoretic, stochastic and machine learning based. Each class of approaches has its own advantages and disadvantages, and there are a number of factors which affect the suitability of a particular approach. Alongside a discussion of these practical considerations, we consider how pairwise relationships can be interpreted in context of the larger system, covering significance testing and moving beyond pairwise relationships to a network level representation.

We will begin with an overview of some of these practical considerations in Section 2.10.1. A major distinction between types of association comes from the focus of an approach. Approaches which primarily consider event times vary from approaches which consider the generating similarity between processes. In Section 2.10.2 we give an overview of the general approaches and assumptions within each class which affect the target association relationships and how the significance of values can be empirically determined, concluding with a discussion on classifying these estimated values in Section 2.10.3.

2.10.1 Practical considerations

In the real-world, physical constraints on a system can result in patterns within the data which do not reflect a significant association. Patterns or abnormalities in distributions of timings may be due to limitations arising from the process generating events, or a feature in the detection process.

Inversely, it is possible to detect behaviours which appear to be coordinated, but physical constraints mean that it is not possible for one process to have detected and generated a response event in the time window considered. These limitations need to be considered in the unique context of the data, and the validity of the approach with respect to the context needs to be considered with adequate domain-level knowledge.

Alongside assessing each of these approaches for the suitability in the context of the data, it is also necessary to consider how each of these approaches may be used. In some applications, it is important that similarity between point processes can be calculated and assessed in close to real-time, to establish understandings in evolving situations. This may be the case for financial applications, or where similarity between point processes is used to understand the relationship between natural events, for example in predicting aftershocks after an earthquake (Ogata, 1998) or detecting malicious accounts online (Jamison et al., 2019). In these applications, it is necessary that these metrics can be quickly and efficiently calculated, and that values can be meaningfully be updated as new data comes in. These conditions preclude many of the machine learning based approaches which require time and large amounts of data to train. The memory and computation times can be large for big data sets. Some approaches can be optimised for efficient

computation, but this consideration should inform the final choice of estimator.

Where these approaches are used to inform decisions, it can be incredibly important that the behaviour of an approach is interpretable. Threshold approaches, particularly those where the choice of threshold value can be justified within the context of the data can be a very useful when the aim is detect behaviour which occurs within a significant time window. For example, in work from Novelli et al. (2019), a statistically significant threshold is used, which can provide insights into physical constraints within a system.

Some approaches use more abstract representations of generating processes or the event times, which may hinder the ability of the results to be meaningfully interpreted. The concept of explainability describes the ability to explain and interpret results or labels which an approach produces. The VigDet model from Zhang et al. (2021), learns an embedding for each process, which is only meaningful for relative comparison, with the embedding not providing insight into the process itself. Explainability can make a model useful as part of an informed decision making process, with the merits of explainability a topic of debate within the artificial intelligence community (Vilone and Longo, 2021; Simkute et al., 2021; Merry et al., 2021; Linardatos et al., 2020). However, a model does not need to be explainable to be useful, and a model which provides information beyond an association value or classification can give insight into data. Stochastic approaches often fit a model to each process, with this model containing information about the rate of coordinated and self exciting events, and other features of the generating process (Rizoiu et al., 2017; Wu et al., 2022). Work from uses Hawkes processes to learn causal structures between processes alongside a measure of their pairwise similarity (Cai et al., 2022).

This is a brief discussion of some aspects which should be considered alongside performance when selecting an approach to measure or label similarity between point processes or their generating functions.

Significance testing

Once a value has been obtained, it can be useful to understand the significance of that value in context of the system. This can be obtained through significance testing. When a robust mathematical framework underpins an approach, an explicit significance test may exist. In some cases, the relevant null distribution is difficult to calculate or obtain and it may be more practical to perform empirical significance testing.

While outside the scope of this review, if a model is fit to describe the probability distribution of the generating sequence, it is also possible to perform significance testing using the estimated probability distribution. These significance tests are often performed in conjunction with the model fit. While this is beneficial if possible, it does not make sense to formulate a model for the underlying probability distribution only to obtain an analytic form for the significance.

An an example, the process for establishing an empirical significance value is outlined below.

Once pairwise association estimates have been obtained, shuffling event labels provides a robust method to estimate the null distribution and in turn, the significance of each pairwise value. This approach is outlined in Smart et al. (2022); South et al. (2022) for entropy estimators, and is suitable for sequences that use event labels to formulate the sequences upon which the entropy is calculated.

The general process is outlined below.

1. Given a set of event times $\{a_1, ..., a_n\}$ with labels $\{b_1, ..., b_n\}$ where b_i is the process name responsible for event at time a_i, take a random ordering of these labels.
2. Generate the new sequence for estimation according to a technique outlined above.
3. Produce an association estimate for each pairwise relationship using these shuffled values.
4. Repeat the above items until adequate samples have been obtained.
5. The ranked samples are estimates from the null distribution, and calculating the quantile of the rank of the observed entropy estimate provides an empirical p-value.

When techniques to evaluate the association are computationally intensive, this process is expensive for relatively short sequences. Using optimised algorithms can help overcome this. Careful consideration of the required number of samples or minimum required error on the empirical p-value can avoid unnecessary calculations.

This approach can be limited when we are testing for a relationship between processes X and Y which are both conditional on a set of processes Z, performing significance testing by shuffling the event times in process Y. As described by Shorten et al. (2021), this does not retain relationships with other conditional processes, so these dependencies may be falsely attributed to the process X, resulting in a higher rate of false positives. Shorten et al. (2021) describes a method to overcome this. By shuffling the values of Y for each conditionally dependent value of Z, the conditional relationship is maintained and the testing is valid. However, such an approach requires this conditional relationship to be understood.

2.10.2 Pairwise relationships

In this review, we have considered four broad classes of techniques to measure the similarity between pairs of point processes. Some of these approaches aim to approximate the process's behaviour, while others analyse event data directly. Approaches fall along a spectrum between these two extremes, and the distinction between approaches can be

subtle. In practice, often the implementation of event level approaches supplement these results by extrapolating to a relationship level value or label which may implicitly make assumptions about the generating process. The suitability of both the event and process level approach should be considered with respect to the data context to ensure the combined approach will produce meaningful results.

This distinction is most pronounced for heuristic approaches. In the literature, there are two broad classes of approaches which exist. Local decisions consider the context of the metric value with respect to the concerned process or its neighbouring processes, while global decisions consider this value in the context of the wider network.

By considering statistics on the set of pairwise inter event times, it is possible to measure features of the relationship between pairs of point processes. Moving from a set of inter-event times to a label or measure of similarity for the pairwise relationship poses its own set of challenges.

Measures which consider the median event time aim to capture the typical behaviour between the two processes. Similar to edit-distance based approaches, using the inter-event time does not allow for any events to be classed as more important, giving equal weighting to all events. Both of these approaches produce a value which describes how coordinated or similar the relationship between the two processes is. Being able to interpret this value in context is incredibly powerful. The strength of the relationship can be relative or not clearly linked to the measured values, so being able to establish a notion of significance is an area of research.

Information-theoretic approaches strive to measure the level of information which one process contains about another, resulting in a relationship level value. These measures are thus inherently pairwise, although since they use standard units of measurement, can be compared between systems. Sequences of event times are treated as realisations from a generating process, and more realisations result in estimates which lie closer to the true value. Using smaller subsets of event times can lead to bias within the estimates, which can impact the quality of results. Information theoretic approaches often give results in the units of information, which can be modified to provide a meaningful measure for information flows between approaches (Shannon, 1948).

Stochastic approaches either focus on modelling each point process separately (Filimonov and Sornette, 2015; Zhang et al., 2020), or using a single stochastic model to represent groups of processes (Hino et al., 2015; Bacry and Muzy, 2016). Both of these approaches treat event times as realisations from a stochastic process which may take the form of a Poisson, Hawkes or Markov model. This model can provide additional insights into the pairwise relationships between processes, as they do not produce a single comparative value, and features of the models can be contrasted. This framework does not lend itself as readily to summarising pairwise relationships with a single value, so if this richer

modelling approach is not beneficial, the assumptions imposed through the choice of stochastic model can limit the validity or accuracy of results.

The transparency and explainability of machine learning approaches is a well-known drawback, with these approaches often producing very high accuracies for classification and pattern recognition tasks, without a modelling process which is meaningful in context. These approaches can primarily consider event times or generating processes, and are very flexible and adaptive in their target behaviours. These methods evolve quickly and can be computationally demanding to implement, so may not be suitable in all applications.

2.10.3 Classifying pairwise relationships

Once each pairwise relationship has been assigned a value which describes the association between two processes, it may become desirable to classify these values into broad classes of behaviours. Similarly, this classification can be applied broadly to all pairwise relationships in the system, although local approaches provide a means for the value assigned to a pairwise relationship to be considered in the process's local context.

Many of these approaches consider the entire system as a network, with individual processes represented by nodes, and the value assigned to the pairwise relationship represented by the edge weight. Decisions can be made globally across this network, subjecting each edge weight to a uniform decision rule. These decision rules can be made using a global parameter, informed by information across the entire system, or a local parameter, inferred using information contained within a smaller subset of the network, to classify the relationship.

Network science provides numerous approaches to classify nodes in a network (Fortunato and Hric, 2016; Fortunato, 2010; Xiao et al., 2022). For pairwise relationships, information is assigned to the edges in a system rather than the nodes, so this problem differs from many node clustering techniques.

Global decision parameters

A decision rule can be equipped with a single parameter to classify all the pairwise relationships in a system. These approaches can be very straightforward and explainable, but can lack the subtlety required when pairwise relationships cannot be directly compared across the network (Hand and Vinciotti, 2003; Zhou et al., 2003; Serrano et al., 2009). This may be the case when the range of values assigned to pairwise relationships is large, and the variance of this range changes across the system.

One straightforward method to classify these pairwise relationships is through the use of thresholds. Thresholds provide a simple means to break up these values into classes, with

values which exceed a particular critical value or lie within a certain range given the same label. A discussion of threshold choice can be found in Section 2.6.5.

Local decision parameters

Localised parameters provide a relationship level decision which is derived using local features. These classifications can use an adaptive threshold based on surrounding features, for example removing all edges which exceed the average weight for a given node.

Local decision rule: backbone networks

While thresholds can be a valuable tool to make global decisions, backbone networks offer an alternative for local decisions. This approach begins by constructing a weighted network, where nodes represent processes and the edge weights represent the strength of the coordination or similarity between respective pair of processes. The backbone network approach aims to thin a weighted network, leaving edges which appear significant on a local scale. Rather than considering the distribution of edge weights across the entire networks, backbone networks consider this distribution locally, preserving edges which disproportionately carry the weight for a node. These weights are normalised by the weighted degree of the node and their significance is established using a null hypothesis of uniformly distributed edge weights (Serrano et al., 2009). The choice of a uniform distribution results in a process which rewards skewed distributions amongst the edge weights for a node, retaining edges which are relatively larger than the other edges from a node.

The use of a null hypothesis allows a significance level to be selected, which may be more meaningful than a global percentile based threshold, as this significance is assessed on a local level. For example, a significance level of $\alpha = 0.05$ will preserve edges corresponds to a 95% certainty that a preserved edge's weight does not come from a uniform distribution. This approach can be meaningful, but in a backbone network, a node with edges which all have similar weights will have its edges removed, which can pose challenges where we expect to see high levels of coordination amongst some of the processes.

A similar approach can be constructed with an alternative formulation of null distribution, where the statistical significance of edges is established locally. This approach tends to be advantageous where the network may exhibit multiple scales and the use of a threshold may mask interesting behaviour on the small-scale. Constructing a thinned network with a custom null distribution may be powerful, but is is crucial that the effects of the choices made are considered and understood. Such an approach may also pose computational challenges when considering large volumes of data.

2.11 Conclusion

The focus of this chapter was on unifying existing work to measure association between pairs of point processes, considering approaches from a wide range of fields and across four classes. We present a conceptual framework to allow the relationships between these approaches to be more easily considered, with a focus on assessing the suitability for specific contexts.

One of the primary focuses of this work was on drawing a distinction between data and process focused association, proposing definitions for both directly coordinated behaviour and similarity between generating processes. These definitions are presented alongside a rigorous and general mathematical framework to ensure each approach can be associated with a strong theoretical basis.

This review of types of association and mathematical definitions aim to help improve understanding of the theoretical basis, statistical significance and behaviours of various techniques, to ultimately improve the accessibility between applications.

While this review achieves these aims and outlines a number of considerations, actual behaviour and idiosyncrasies of measures of association should be verified for a given context. In Chapter 3, we outline a simulation and evaluation procedure which uses simulation to validate the performance of association measures across a wide range of data settings and give an example of using this framework to assess suitability.

Chapter 3

A simulation framework to evaluate association between point processes

3.1 Introduction

Measuring association between point processes is critical to understanding relationships between processes across many domains. These relationships can take many forms and a wide range of measures exist, each with its own strengths and sensitivities. Often, the practical behaviour of a measure doesn't meet theoretical expectations, making empirical evaluation an essential tool to ensure modelling procedures are appropriate and well designed. This chapter aims to provide a robust tool to simulate realistic point process data and evaluate functional measures of pairwise similarity, complementing and building upon Chapter 2, to improve the accessibility and validity of approaches used to measure association between point processes.

Previously no framework existed for the comparison and evaluation of association measures for point processes. Many association measures have been developed independently and are experimentally rather than theoretically motivated, which has led to a body of work which is disconnected and is difficult to unify. Making this vast body of knowledge available across fields opens up more options for measuring association, while simultaneously highlighting the importance of robust selection tools.

In existing work, association measures are often only applied to real data. While this produces real-world insights, a ground truth is often unknown and full knowledge of the setting is not available. Simulated data is an essential tool when validating methods and measuring the effects of hyperparameters, as the ground truth is known. Having a known ground truth makes it possible to understand which features of a dataset are important and influence the performance of a particular approach.

Even with simulation, the range of possible relationships between point processes, and a lack of universally accepted definition for 'similarity' and 'coordination' make it difficult to compare and assess approaches. We use definitions of association types proposed in Chapter 2, and propose methods to assess the performance of association measures for direct coordination and generating process similarity. While these two behaviours are implemented, the proposed simulation framework which can easily be adapted to mimic specific settings. By specifying each source's generating process, self-excitation rate and the impact of external events, this framework can simulate a broad range of behaviours and complex relationships.

In this work, the ability of the data simulation to modulate process relationship and behaviours is used to assess the performance of similarity measures with respect to different data features. This is designed to provide transparency into which features these methods respond most strongly to and which measure is most suitable for a given context, to assist researchers in making informed decisions when measuring pairwise similarity between point processes.

The main contributions of this work are:

- A flexible tool to simulate data representing an arbitrary number of coordinating processes, where the coordination structure is dictated by an underlying coordination network and user input determines the type of association between the pairs of processes.
- A tool to evaluate performance of similarity measures against a range of behaviour types and network structures. These similarity measures can be imported from external libraries, and example code is given to use a function from an external Julia library.
- An open source code-base implementing both the simulation and evaluation procedure. `https://github.com/bridget-smart/PointProcessSimulationEvaluaton`
- A discussion of evaluation output for assessing performance against both direct coordination and generating process similarity using example heuristic and information-theoretic measures. These sample results highlight the importance of carefully defining target behaviour and demonstrate how the framework can be used to identify measures which are unsuitable for particular contexts.

In Section 3.4, we begin by outlining the framework used to simulate data, describe the input parameters and their effect, then provide an outline of the data simulation algorithm. Section 3.5 describes the evaluation procedure for measures of pairwise association in point processes, and Section 3.6 provides a discussion of example output for heuristic and information-theoretic methods chosen to explore the capabilities and limitations of the simulation framework. Complete sample output in given in Appendix A.

3.1.1 Gap in the literature

This work addresses a need for a robust and flexible simulation framework for validating and comparing approaches. In existing literature, methods to measure association are domain-specific and not widely considered or adopted. There is a body of work which applies a technique developed in one field to data from another application (Cai et al., 2022; Kobayashi and Lambiotte, 2016; Kim et al., 2019). Several works strive to increase the accessibility of techniques between fields by describing and implementing a collection of approaches, but these are generally limited to heuristic approaches (Cliff et al., 2022; Hino et al., 2015). Chapter 2 provides a more comprehensive overview of existing techniques, a conceptual framework and a review of practical considerations. While this discussion is useful, the experimental behaviour of approaches is difficult to anticipate given the complexity of the data setting.

Work from Victor (2005) which focuses on spike train data, aims to evaluate how similarly approaches perform on a sample real-world data set without using simulation. This procedure does not aim to capture how each measure performs with respect to different features within the data and is not generally useful for applications which vary from the sample data set. Unlike this work, it also does not allow a user-specified measure of evaluation, which is usually computed with a matrix-distance or correlation (Alothali et al., 2018; Hino et al., 2015). The MMPP package from Hino et al. (2015) provides an implementation in R of many distance and similarity measures for point processes, makes approaches accessible from a range of disciplines, but does not provide a framework for understanding and evaluating these measures.

3.2 Mathematical background

Several definitions are given to provide context to the simulation and evaluation framework described. Each set of event times will be represented by a temporal point process.

Definition 3.2.1 (Temporal Point Processes)**.** *Temporal point processes are random processes consisting of points in the non-negative real numbers, representing a collection of events in continuous time. For a process, $X^{(j)}$ with n_x realisations representing event times, we let $x_i, i \in [1, n_x]$ represent the time of the i^{th} event.*

This modifies the notation from Chapter 2 to allow more than two processes to be easily represented using the index j. Throughout this chapter, we will use the notation t_j to represent a simulated time associated with process j within algorithm descriptions. The reason for this is twofold: these simulated times are drawn from continuous distributions and so represent random variables; and these times do not necessarily represent event times for a particular process.

Each point process is governed by a generating process which defines how event times are generated. These event times may be self generated, or derived from external events or events in other temporal point processes. In this work, generating processes are approximated using a mixture of functions and random processes to mimic the complexity which occurs in real-life systems.

A self-exciting process is a process where the conditional likelihood of an event occurring is conditional on events in the process' history. A Hawkes process is a self exciting process with an intensity function in the form,

$$\lambda(t|\mathcal{H}_t) = \lambda_0(t) + \sum_{i|t_i<t} \phi(t - t_i) \quad \text{(Definition 2.8.1).}$$

Here, $\mathcal{H}_t$ represents the history of the process, or all events up to time t_i. The base intensity function, $\lambda_0(t)$, represents the rate at which events occur independently of each other. This quantity is often constant, but can be a function of continuous time. The intensity function $\phi(\cdot)$ represents the intensity of self-exciting events.

The choice of decay function, $\phi(\cdot)$, is typically monotonically non-decreasing so the conditional likelihood jumps when an event occurs then gradually decays. A typical choice is an exponential decay kernel, where $\phi(\cdot) = e^{-\delta(\cdot)}$, (Rizoiu et al., 2017; Filimonov and Sornette, 2015).

If an event in process X, x_i, is coordinated with a matching event in process Y, y_{m_i}, we say that y_{m_i} is a child event of x_k. As in Chapter 2, we use the convention that

$$m_i = \underset{k}{\operatorname{argmax}}\, y_k < x_i,$$

for a *directly coordinated* event pair x_i and y_{m_i}.

If a process is able to coordinate with events in other processes, we say that this process is in a *coordinating* state. In this model, this state is governed by a continuous-time hidden Markov process, which is a memoryless stochastic process (Definition 3.2.2, Definition 3.2.3).

Definition 3.2.2 (Markov Process)**.** *The process $S_n,\ n \in \mathbb{N}$ is a Markov process with realisations $s_n,\ n \in \mathbb{N}$ with transition matrix P if for all $n \in \mathbb{N}$ and $w, s_0, ..., s_{n-1} \in \mathbb{S}$,*

$$P(S_n = w \mid S_0 = s_0, ..., S_{n-1} = s_{n-1}) = P(S_n = w | S_{n-1} = s_{n-1})$$

Definition 3.2.3 (Hidden Markov Process)**.** *A hidden Markov process is a Markov process where each state is not directly measurable or observable.*

We will use these processes within the data simulation framework in following sections.

3.3 Pairwise relationships between point processes

3.3.1 Types of association

Pairwise association between point processes exists across a range of fields and takes a number of different forms, so it is crucial that our simulation model is complex enough to capture a range of association types and underlying network structures.

There are several distinct dimensions of relationships between point processes. In Chapter 2, we argue that existing work aims to measure *direct coordination*, or *generating similarity* between pairs of point processes. These two measures of association capture a range of complex relationship types, and help us understand the differences between them.

Since there is not a universally accepted definition of direct coordination or similarity, we use the two definitions (Definition 2.5.3, Definition 2.5.4) proposed in Chapter 2, which we will use throughout this chapter.

3.3.2 Association in context

The meaning of similarity or coordination between a pair of point processes vary in different contexts. It is important to clearly define the target behaviour when selecting an association measure. This target behaviour is often considered when an association measure is designed, but this is not often validated.

In online social network analysis, researchers aim to identify accounts which *coordinate*, where accounts share similar or identical post features including content and links within a short time period (Graham et al., 2020; Giglietto et al., 2020; Keller et al., 2020; Schäfer et al., 2017). Cognitive neuroscientists aim to measure direct coordination to measure and understand cortical function within and between cortical regions (Tognoli et al., 2020; Bressler and Kelso, 2016).

In other areas, for example in natural disaster or financial modelling, it is more desirable to determine the nature of the relationship between the generating processes. In these areas, stochastic models are often employed as they allow easy comparison between models of the underlying generating processes (Sancetta, 2017; Wang et al., 2021; Musmeci and Vere-Jones, 1992; Ogata, 1998).

3.3.3 Existing approaches

As discussed in Chapter 2, existing techniques to measure pairwise association fall into one of four main areas.

1. Heuristic approaches use statistical features of the timings of the two processes, often looking at the differences between pairs of event times, to measure the similarity. Using the event times is ideal in applications where similarity needs to be computed quickly or the objective is clearly defined and can be directly measured.

2. Information-theoretic approaches aim to measure the amount of information flowing between pairs of processes. Transfer entropy is a common directed measure of information flow which is used (Shorten et al., 2021; Mijatovic et al., 2021; Melnik and Usatenko, 2016; Ameli and Sassi, 2018; Spinney et al., 2017).

3. Stochastic approaches, especially those fitting event times to a Hawkes process model, aim to directly model the underlying generating process, considering both the baseline rate of events and how the rate of events in other processes impacts this rate. These models can be limiting due to modelling assumptions, but can give rich insight into the generating processes and how the processes interact.

4. The final approach uses machine learning methods, example to learn event density functions (Rosenfeld et al., 2022; Bodnar, 2021; Wen et al., 2021), or cluster processes to extract communities which exhibit similar behaviour or relationships (Luceri et al., 2020; Alothali et al., 2018; Alizadeh et al., 2020; Henelius et al., 2016). Machine learning approaches can lack explainability, can require large volumes of data, computational resources or time to train and run, and may not produce results which are meaningful outside of the immediate data application.

While these techniques are well established within their domains, they are often not applied outside of the field they originate in. Since these techniques are often extremely specialised to detecting or measuring a particular type of similarity between pairs of point processes, it can be challenging for their suitability on similar or different problems to be accurately gauged.

This simulation and evaluation framework aims to allow these approaches to be compared, and their applicability to various problems established. In this work, we will consider evaluation for heuristic and infomation-theoretic approaches, as these two classes summarise the pairwise association with a single value. This reduces the complexity of comparing results with a ground truth value and minimises sources of error within the simulation framework. In future work, we will extend these evaluation techniques to allow stochastic and machine learning approaches to also be considered.

3.4 Data simulation

This simulation framework generates event time data for N distinct point processes each with a variable number of events. These processes are characterised by their interactions, complex relationships, self-excited behaviours and periods of inactivity.

The framework is designed to be flexible and robust, while taking advantage of direct relationships between parameters and behaviours to maintain a clear ground truth for evaluation. Association between processes is represented by an underlying direct coordination network structure, and similarity between parameters which control their generating processes. These combined effects allow a spectrum of association types to be represented, and the behaviour of each process to be individually controlled. The simulation moves between a number of processes to generate event times, and the combination of these processes leads to realisations which cannot easily be described by a simple model. The complete set of event times is generated sequentially for each process, with any dependent coordinated processes considered first.

This simulation is event focused, with the generating process dictated by an underlying Hidden Markov Model (HMM). This HMM describes if a process is in a coordinating a non-coordinating state, which changes the mechanism of event generation.

At any time point, each process is able to produce coordinating or non-coordinating events and the HMM allows processes to produce both self-generated and coordinated events interchangeably or remain in one state for long periods. For example, a social media user who occasionally coordinates and is usually in a non-coordinating state but occasionally changes to a coordinating state, or a user who frequently alternates between coordinating and not, can both be represented within a single simulation.

The flexibility of this simulation can be used to mimic a range of data sets, from situations where processes only directly respond to one another to situations where similarity exists between point processes which never directly coordinate. As the processes for self-generated and coordinated events are flexible, this simulation can reproduce a large range of realistic data settings and can easily be adapted.

3.4.1 Simulation model

This model generates event times for N distinct point processes. These point processes can produce both self-generated and directly coordinated events, with *coordination periods* describing which event type is produced at a given time.

A Hidden Markov Model (HMM) is used to construct these coordination periods. Once the hidden coordination periods for each process are known, events are generated sequentially, drawing new event times from the generation process for either self-generated or

coordinated events. When in a coordinating state, each process may only coordinate with processes with which it has direct coordination relationships.

These relationships are based on an underlying coordination network, represented by a directed acyclic graph. This underlying coordination graph represents the direction and strength of the pairwise direct coordination relationships between point processes. The nodes in this graph, $A = (N, E)$, represent processes, with the weight of edge a_{ij} representing the strength of the relationship between the source process i and the target process j.

It is necessary for events in possible source processes to be generated before event times can be calculated for a target process. This is achieved by generating the full set of event times for each processes according to their reverse topological order.

The general process for simulation is:

1. Derive a reverse topological ordering, O, from A of all processes.
2. For each process i in O:
   (a) Simulate coordination periods using the HMM.
   (b) Generate the full set of event times with reference to the set of possible source processes, S_i.

A visual example of simulated event times for three processes is shown in Figure 3.1. In this figure, each row represents a single process, with grey bars representing the hidden coordination periods, with arrows corresponding to coordination relationships between pairs of events.

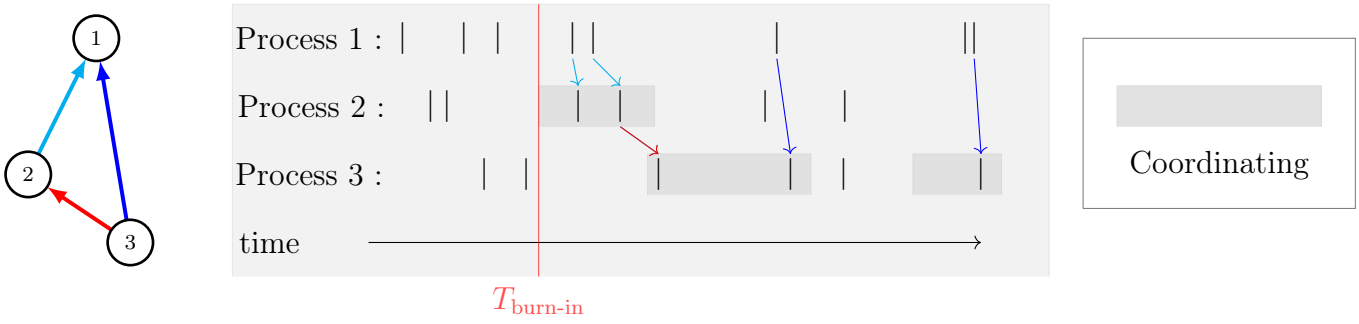


Figure 3.1: Example of event times from a trio of coordinating processes. Dark grey indicates the respective process is currently in a coordinating state. Coloured arrows indicate source-target relations between events, with colours corresponding to the matching relationship shown in the underlying coordination network (left). Before $T_{\text{burn-in}}$, no processes are able to coordinate.

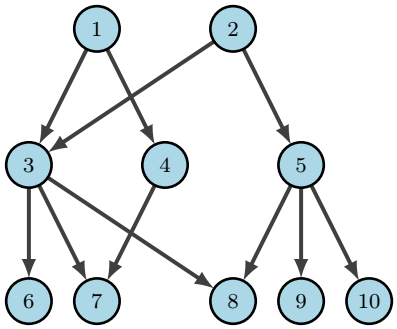
3.4.2 Traversing the set of processes

In this simulation, the full set of event times is generated for each process, one at a time, rather than generating event times for all processes in sequential order. This means that the processes need to be considered in an order which ensures all events in possible source processes exist before they are required.

The *topological ordering* or topological sorting of a graph gives a sorted list of the nodes in a graph such that if the directed edge $e_{i,j}$ from node i to node j appears, the node i appears before j. An example of a topological ordering is shown in Figure 3.2.

This ordering does not need to be unique as independent components of the graph can be traversed in an interchangeable order. As any topological ordering ensures all possible source events exist, the simulated event times are only dependent on the order chosen up to the effects of the random number generator used. To overcome this, within the implementation we use a new random number generator for each process.

The topological ordering will exist for all directed acyclic graphs, and is the reason we require the underlying coordination network to be directed and acyclic. This assumption may be limiting, as it prevents cycles of coordinating processes from being included. For example, if three processes labelled 1,2,3 coordinate, each looking to the process before, a cycle is formed, and an appropriate ordering cannot be constructed. By generating the complete set of times for each process according to the topological ordering for the directed acyclic coordination graph, we ensure that the information required to generate times for all dependent processes exists.



Some possible topological orderings include:
1,2,3,4,5,6,7,8,9,10;
1,4,2,3,6,7,5,9,10,8;
2,5,10,9,1,4,3,7,8,6...

Figure 3.2: Example of topological ordering

3.4.3 Coordinating state

Underlying each point process is a hidden state which describes if that process is currently able to coordinate or not. This hidden state is described by a HMM, with rates that can be specific to each process. This HMM is a continuous time Markov Process, with transition

rates $c_{0,1}$ and $c_{1,0}$, describing the rates which the process moves from a non-coordinating state to a coordinating state respectively (Figure 3.3). The transition times between these states are used to simulate coordination periods for each process, independently of all other processes. We denote these coordination periods $c(t)^{(j)}$ for the process j. The values of $c_{0,1}$ and $c_{1,0}$ are uniquely specified for each process j, denoted $c_{0,1}^{(j)}$ and $c_{1,0}^{(j)}$ and given in the vector $\boldsymbol{c}$.

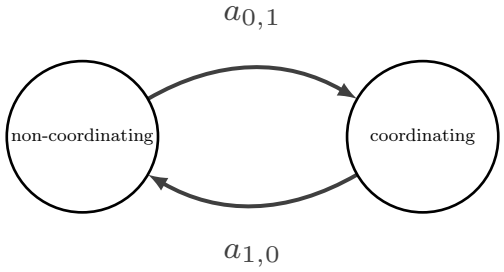


Figure 3.3: Diagram of the HMM for each process.

3.4.4 Generating event times

Once the set of source processes, S_j and the coordination periods, $c(t)^{(j)}$ have been calculated for a given process, the event times are simulated using a combination of two generating processes. These two processes produce times for both self-generated and coordinated events. We denote current time in the simulation as t_{current} and continue the simulation until the end time, T_{end}.

The general procedure to generate the event times for process j is given in Algorithm 1. Here, E_j represents the set of event times associated with process j.

Self-generated events

When not in a coordinating state, event times are generated using a Poisson process, with rate unique to each process. For process X_j, this Poisson process has rate $\lambda^{(j)}$.

Each event time is simulated by generating an exponentially distributed time delay, with the rate given by the hidden coordinating state and the parent processes, if dependencies exist. These new times are accepted if they fall within the same coordination period or are not greater than the final time. This thresholding process truncates the distribution.

Coordinated events

The process for generated coordinated event times is dependent on the source and target process. For each coordinated event, the general procedure for process j is:

Algorithm 1 Overview of data simulation procedure for process i

while $t_{\text{current}} < T_{\text{end}}$ **do**
  if $c(t_{\text{current}})^{(j)} = 0$ **then** // Process not coordinating, see (Section 3.4.4)
    Generate a self-generated event time from a Poisson process with rate $\lambda(i)$
    $t_{\text{current}} \leftarrow t_j$
    $E_j \leftarrow (E_j, t_j)$ // Store the generated event times, t_j in the list E_j
  end if
  if $c(t_{\text{current}})^{(j)} = 1$ **then** // Process coordinating, see (Section 3.4.4)
    Generate coordinated event times according to coordination procedure
    $t_{\text{current}} \leftarrow t_j$
    if Coordinated event generated **then**
      $E_j \leftarrow (E_j, t_j)$ // Store the generated event times, t_j in the list E_j
    end if
  end if
end while

When a process, $X^{(i)}$, is in a coordinating state, a new time is simulated using a coordination delay function, $t_d(i, j)$ unique to each source/target process pair.

It is possible for no coordination to occur, or the process to coordinate with itself. This represented by $w_j^\star$, which is the likelihood of no coordinated event occurring. This case is treated like the process is coordinating with itself to update the current event, but no event times are saved.

Once the parent event is chosen, an inter-event time is generated. The time for the subsequent coordinated event is given by the sum of the most recent event time from the parent process, $x_i^{(\text{source})}$ and the corresponding coordination delay function $t_d(\text{source}, i)$.

Algorithm 2 Generating coordinated events

while $c(t_{\text{current}})^{(j)} = 1$ (Coordinating) **do**
  if $w_j^\star > z \sim U(0, 1)$ **then** // Producing an event
    Draw a source process s from S_j (Details in Section 3.4.4)
    $t_j \leftarrow t_{\text{current}} + t_d(s, j)$ // Generate the coordinated event time
    $E_j \leftarrow (E_j, t_j)$ // Store the generated event time
  else // No event is produced
    $t_j \leftarrow t_{\text{current}} + t_d(j, j)$ // Generate a new time
  end if
  $t_{\text{current}} \leftarrow t_j$ // Update the current time
end while

This coordinated event is not constrained to fall before the current time. If no event time is chosen, an inter-event time corresponding to a self-coordinating event, where the parent process is the current target process, is used to update the current time so the simulation can continue. Here, E_j represents the set of event times associated with process j.

An overview of the procedure is given in Algorithm 2.

Within this simulation framework, this coordination delay function $t_d(\text{source}, i)$ can be user specified, or specified using a Beta distributed delay from the parent event. This beta distributed delay is generated using the process dependent parameters $\boldsymbol{\beta}^{(i,j)} = (\alpha^{(i,j)}, \beta^{(i,j)})$, multiplied by some scaling constant, s and is used if a custom function is not provided.

Determining the source process

Each coordinated event is tied to a source process and this source process is selected using a weighting given by the product of the strength of the direct coordination relationship and the time which has elapsed since the last event in the process. This time delay is weighted using the function $d(t)$. It is recommended that $d(t)$ is chosen to allow smaller time delays to be rewarded, although the choice of $d(t)$ is not constrained.

The procedure to determine the source process, s, given S_j, the list of all possible source processes for target process j is outlined in Algorithm 3. Note that it is possible for a process to coordinate with itself, i.e. $s = j$.

Algorithm 3 Selecting the source process

for i in S_j **do**
  $t_i \leftarrow t_k^{(i)}$ for $\max_k\{t_{\text{current}} - t_k^{(i)}\}$ // $t_k^{(i)}$ is the k^{th} event time from process i.
  $w_i \leftarrow d(t_{\text{current}} - t_i) \times a_{ji}$ // $d(\cdot)$ is the time importance decay function
  $p[i] \leftarrow w_i$
end for
$p \leftarrow p/\text{sum}(p)$ // p represents a distribution
Sample s from p

3.4.5 Additional details

Normalising coordination relationship strengths

To ensure these relationship strengths can be compared globally across the system, we constrain the column sum of A to be $1 - w_i^\star$, or $1 = \sum_j A_{i,j} + w_i^\star$. If this normalisation is not performed, the number of times a source and target pair coordinate cannot be

directly related to the strength of the direct coordination relationship in A, preventing comparison between processes.

Burn-in period

A burn-in periods is used to establish an event history for each process which allows coordinated events to be generated immediately from the beginning of the simulation.

During the burn-in period, no coordination is allowed to occur and each process only generates event times according to their non-coordinating state. This ensures that each process has a history of states to draw on when generating times. After this burn-in period, the complete set of event times for each process is calculated.

3.4.6 Algorithm

The implementation of this algorithm in PYTHON is provided at `https://github.com/bridget-smart/PointProcessSimulationEvaluaton`. This provides a number of modular functions which can be easily modified and interchanged to tailor the behaviour of the simulation algorithm. A complete outline of this implentation is given for reproducability.

The variables used in the following pseudocode are:

- N , the number of processes
- A, the $N \times N$ weighted adjacency matrix for directed acyclic coordination graph
- E_j, the list of all event times for process j.
- $\boldsymbol{c}$, the $N \times 2$ matrix, with row i containing $(c_{0,1}^{(i)}, c_{1,0}^{(i)})$.
- $\boldsymbol{\lambda}$, the vector of rates for Poisson self-excited events
- $d(t)$, the time importance decay function
- $t_d(i, j)$, the coordination delay function for source i to target j
- t_{current}, the current time
- T_{end}, the end time
- $t_k^{(i)}$ denotes the k^{th} event time from process i

Algorithm 4 Outline of overall data simulation procedure for N processes

Calculate self-excited event times for the burn-in period for Process $j = 1, ..., n$.
Calculate the coordination periods for all processes, $c(t)^{(j)}, \forall\ j\ \in\ [1, N]$.
$O \leftarrow$ reverse topological ordering of A
for j in O **do**
  while $t_{\text{current}} < T_{\text{end}}$ **do**
    if $c(t_{\text{current}})^{(j)} = 0$ **then** // Process not coordinating
      $t_j \leftarrow$ Draw a self-generated event time from a Poisson process with rate $\lambda(i)$
      $t_{\text{current}} \leftarrow t_j$
      $E_j \leftarrow (E_j, t_j)$ // Store the generated event times, t_j in the list E_j
    else if $c(t_{\text{current}})^{(j)} = 1$ **then** // Process coordinating
      if $w_j^\star > z \sim U(0, 1)$ **then** // Producing an event
        for i in S_j **do**
          $t_i \leftarrow t_k^{(i)}$ for $\max_k\{t_{\text{current}} - t_k^{(i)}\}$
          $w_i \leftarrow d(t_{\text{current}} - t_i) \times a_{ji}$
          $p[i] \leftarrow w_i$
        end for
        $p \leftarrow p/\text{sum}(p)$ // p represents a distribution
        Sample s from p
        $t_j = t_{\text{current}} + t_d(s, j)$ // Generate the coordinated event time
        $E_j \leftarrow (E_j, t_j)$ // Store the generated event time
      else // No event is produced
        $t_j \leftarrow t_{\text{current}} + t_d(j, j)$ // Generate a new time
      end if
      $t_{\text{current}} \leftarrow t_j$ // Update the current time
    end if
  end while
  Save E_j
end for

3.4.7 Parameters

To generate times for n distinct processes, the following parameters are used.

A is a $n \times n$ weighted adjacency matrix which describes the DAG containing the coordination structure for directly coordinating events. We take $a_{ij} \in [0, 1]$. This matrix represents the ground truth for directly coordinating relationships. A matrix with more edges represents a system where many processes directly coordinate, whereas the presence of paths represents processes which coordinate in sequence.

$\boldsymbol{w^\star}$ is a $1 \times n$ vector that describes the probability of each process producing a coordinated event when in a coordinating state. This is set so $1 = \sum_j A_{i,j} + w_i^\star$ for all i, or that the column sum, plus the respective $w_i^\star$ is equal to one. For parameter generation, the minimum value of $w_i^\star$ is set, and the values in A are normalised by the value $\left(\frac{\sum_j A_{i,j}}{(1-w^\star)} \mid \max_j \sum_j A_{i,j}\right)$.

$\boldsymbol{c}$ is a $N \times 2$ matrix, with row i containing $\left(\frac{1}{c_{0,1}^{(j)}}, \frac{1}{c_{1,0}^{(j)}}\right)$. $(c_{0,1}^{(j)}, c_{1,0}^{(j)})$ are the parameters which define the rates of the continuous time hidden Markov process which is used to generate the coordinating state of process j. In the hidden Markov process, transitions from a non-coordinating state to a coordinating state occur with rate $c_{0,1}$ and transitions back occur at rate $c_{1,0}$. The sequence of times generated from this hidden Markov process is denoted $c(t)^{(j)}$ and describes the state at time t for process j. If a process is always in a coordinating state, this can be represented by setting $c_{0,1} > 0$ and $c_{1,0} = 0$. If a process moves between a coordinating state and a non-coordinating state quickly, both $c_{0,1}$ and $c_{1,0}$ should be large. If a process goes through long periods of coordinating and long periods of not coordinating, both rates should be small. The expected time which the process remains in each state is given by the recriprocal of the rate, so these values can be experimentally obtained if data is available. All processes start in a non-coordinating state during the burn-in period, so if it is necessary for each process to always coordinate after this point, an additional burn-in period may be needed.

$\boldsymbol{\lambda}$ is the vector of rates which are used to generate the self-excited event times. These times are generated from a Poisson process with rate $\lambda^{(i)}$. The expected time between events is given by $\frac{1}{\lambda^{(i)}} \; \forall \; \lambda^{(i)}$. If this process produces self-excited events at a rate much higher than is possible for coordinated events, for example if the time delay for a coordinated event is very long, periods of coordination will be characterised by fewer events. It can be important to make sure these two types of events occur at similar overall rates, if periods of coordination are not easily distinguished by the event rate in real data.

$t_d(i,j)$ is the coordinated delay function used to simulate the time delay for a coordinated event in target process i using source process j. If not specified by the user, B and s are used instead.

B is a $n \times n$ matrix of tuples used to calculate the beta distributed delay between a parent and child coordinated event pair. $B_{(i,j)} = (\alpha^{(i,j)}, \beta^{(i,j)})$ which are the parameters for an event in process i which has a parent event in process j. The beta distribution was chosen as the two parameters afford a large amount of control over the shape of

the distribution. This parameter is only used if a custom coordinated event decay is not specified.

$d(t)$ For a coordinated event, the parent event is chosen using a normalised score for each possible parent process. The score for process j is calculated using the product of the time importance decay function applied to the time from the most recent event in process j and the weight of the directed coordination edge between process i and process j. This function impacts how the memory of a process impacts coordinating events. If $d(t)$ is chosen as a exponential function, more recent events are given a higher weighting, but all possible parent events are considered. If $d(t)$ is a threshold function, coordination can only occur within a specified window. If $d(t)$ is chosen to be uniform, the elapsed time does not impact the parent process.

$T_{\mathbf{start}}$ The start time. No events can occur before this time.

$T_{\mathbf{end}}$ The end time after which no events can occur.

$T_{\mathbf{burn\text{-}in}}$ The burn-in time is used to ensure all processes have generated some events which can be used as parent events. This time needs to be larger than T_{start} and smaller than T_{end}. Between T_{start} and $T_{\text{burn-in}}$, only self-excited events are permitted.

3.5 Evaluating performance

The flexibility of the data simulation framework allows for empirical evaluation of methods which produce a measure of association between two point processes. An evaluation procedure is outlined which considers performance against the underlying coordination network A, parameters of the underlying hidden Markov model which dictate coordinating periods, $\lambda^{(i)}$ which describes the self-excited event rate and the function which determines the length of a delay between coordinated event pairs.

The process follows the general procedure:

1. Parameter values are defined for the default values that remain fixed and those that are varied.
2. A number of sample data sets are generated by iterating through the range of parameter values. For each unique set of parameter values, the function to measure pairwise association is applied to the simulated event times.
3. This evaluation procedure is performed multiple times to reduce the variance of the estimated performance. Seeds are used to ensure results are reproducible.
4. The pairwise relationship values are used to evaluate performance of the target association measurement.

We outline two evaluation procedures for pairwise association measures of point processes;

(i) for directly coordinated behaviours, and

(ii) for similarity between generating processes.

Both can be used for any method which produces a single value describing the pairwise relationship, with larger values representing greater similarity or stronger relationship.

Since there may be a range of target behaviours, the two evaluation strategies proposed here may be inappropriate for some applications. However, these cover two target behaviours which frequently occur in literature, and should cover a broad range of useful cases. These can also serve as a basis from which further evaluation strategies can be developed.

3.5.1 Evaluating against directly coordinating behaviours

The pairwise association measures produced by many methods can be used to form a weighted matrix, $\hat{C}$ which can be directly compared to the underlying coordination network, C. This matrix comparison can only hold if the association measure is a global

measure, and can be compared all between pairs of nodes. If this is not the case, local comparisons, for example in-degree edge weight distributions should be considered.

The biggest challenge to meaningful comparison between estimated pairwise relationship values and the true values from C, is that these values can fall on different scales. The values produced by different methods can vary greatly, for example correlation (Schreiber et al., 2003; Palm et al., 1988) which produces a measure between -1 and 1, co-occurance counting which produces an finite non-zero integer (Palm et al., 1988; van Rossum, 2001) or inter-event time statistics which can take any finite value. Here, we propose several methods to overcome this challenge, using normalisation techniques and by looking at the proportional relationship between values.

Matrix similarity

Here, we propose and implement two methods to measure similarity between elements in the matrices $\hat{C}$ and C. These methods don't require the elements to be directly comparable. They use relative relationships and normalisation to mitigate problems caused by inconsistent scales.

1. Correlation coefficient These first method used to assess the performance of association methods uses the square of the correlation coefficient between value pairs, which is equivalent to the coefficient of determination for a linear model which takes the form $\hat{C}_{i,j} = mC_{i,j}$ for some coefficient m. This produces a similarity measure between 0 and 1, which is 1 when $\hat{C}_{i,j}$ is a multiple of $C_{i,j}$, and 0 when there is no relationship between $\hat{C}_{i,j}$ and $C_{i,j}$. As this measure considers element-wise linear relationships, is not reliant on normalisation and is meaningful in context, this is the similarity measure used in Section 3.6.

2. Matrix norms Matrix norms allow matrices to be compared. There are many different types of matrix norms, with the norm of a matrix A denoted $||A||$. The two types of norm are implemented for assessing distance between two matrices are:

> **Frobenius norm,** $||A||_F$ is given by the square root of the sum of squared elements of $\hat{C} - C$. Equivalently,
>
> $$||A||_F = \sqrt{\sum_{i,j} a_{ij}^2}.$$
>
> This norm produces larger values for a pair of matrices which have greater element-wise distances, penalising larger errors more harshly.
>
> **Max norm** $||A||_{\mathbf{max}}$, which takes the norm of the matrix to be the maximum element-wise value,
>
> $$||A||_{\max} = \max_{ij} a_{ij}.$$

This allows the strength of the strongest coordinating relationship or largest element within the matrix to be directly compared.

Any norm can be used to induce a distance metric between two matrices A and B, for example

$$d(A, B) = ||A - B||.$$

In our implentation, both matrix norm methods begin by normalising both $\hat{C}$ and C by their respective element sum. Normalising by the sum of all elements forces the elements to represent a probability. In context this is equivalent to requiring each coordination relationship to represent a proportion of the total coordination in the system. This normalisation scheme was chosen for its meaningful contextual interpretation and the preservation of relative relationships. While row or column normalisation may be a more traditional choice, these normalisation types force the in- or out- degree relationship strengths for each process to have a unit sum, which does not preserve meaningful relative relationships between processes.

Once the matrix is normalised, the similarity between the normalised matrices can be calculated using a matrix norm.

3.5.2 Evaluating against similarity of generating processes

The simulation framework of Section 3.4 is underpinned by the underlying coordination network which is directly comparable with the strength of the directly coordinating relationship between processes, it is less obvious how to evaluate performance against similarity of generating processes. A simple method is proposed here, although this should be an area of focus for future work. The assumption which underpins this technique is that the generating process can be estimated using two Poisson processes, with rates $\lambda_c^{(i)}$ and $\lambda_n^{(i)}$. These two rates represent the rate of events for process i when it is in a coordinating state, $\lambda_c^{(i)}$, and a non-coordinating state $\lambda_n^{(i)}$. When in a non-coordinating state self-excited events are generated using a Poisson distribution so, $\lambda_n^{(i)}$ is a maximum likelihood estimate for $\lambda^{(i)}$. In a coordinating state, event times are generated using a more convoluted process, so $\lambda_c^{(i)}$ is an approximation of the process behaviour.

Using the coordination periods generated using $\boldsymbol{c}$ and the set of simulated times, T for each process, the rates, $\hat{\lambda}_c^{(i)}$ and $\hat{\lambda}_n^{(i)}$, can be estimated by taking number of events which occur in a particular time period, using that for a Poisson process with rate λ, the expected time between events is given by $\dfrac{1}{\lambda}$.

This procedure gives two values for each process, $(\hat{\lambda}_c^{(i)}, \hat{\lambda}_n^{(i)})$, which are then treated as points in $\mathbb{R}^2$. The Euclidean distance between each pair of points is used to measure how dissimilar they are, with the distance normalised to give a similarity using the function

$f(d) = \frac{1}{d+1}$, so that a similarity measure between 0 and 1, where 1 is given if the two points are congruent and 0 if there is infinite distance between them.

These values are used to form the matrix B, which represents the ground truth for the similarity of generating processes in the simulation. This matrix is compared to the estimated relationship strengths, $\hat{B}$, using the methods outlined in Section 3.5.1.

3.5.3 Representing a range of behaviours

The parameters of the simulation framework are varied to represent a range of behaviours and relationship types between pairs of point processes. For each set of parameters, k datasets are simulated, to ensure the performance evaluation estimate is robust to outliers.

Since all but one parameter is fixed, the choice of default values is incredibly important. In practice, these default values can be informed by prior knowledge of relationships and data parameters. Some of the factors which should be considered include:

- Typical number of realisations and processes
- Proportion of time spent in coordinating states
- If physical constraints prohibit directly coordinating relationships between particular processes
- Expected generating process or delay function for a coordinated event
- Expected generating procedure or rate for self exciting events

The clear relationship between input parameters for the data simulation make it possible for artifical datasets which mimic real-world contexts to be generated. Here, the focus is on a completely artifical dataset designed to test the evaluation procedure and consider a range of behaviours and relationship types. To ensure this work is relevant, we will include behaviours present in real world datasets, direct coordination (Graham et al., 2020; Bauwens and Hautsch, 2009) as well as similarity between generating processes (Musmeci and Vere-Jones, 1992; Chi et al., 2007). The defaults will be chosen to produce a baseline of association behaviours which cannot be easily categorised into their association types, and the considered parameter values will range across a spectrum of realistic behaviours.

When evaluating how performance is affected by a single parameter, only iterations where that parameter is the target parameter are considered, to avoid over representation of the default values.

While this simulation framework is designed to be as flexible as possible, for illustration purposes, we describe a default range of parameter values chosen to produce complex

coordinating behaviours which produce both self generated and coordinated events at similar rates. A small system of 10 processes is used, with stochastic functions used for most parameters to encourage variance in the simulated data sets. This simulated data set includes a spectrum of direct coordination relationship strengths, representing a setting where processes experience a range of association types. Since this evaluation procedure considers one variable at a time, we aim to balance the complexity of the simulation with our ability to attribute trends by reducing randomness and clear discriminators within the default settings. To maintain interpretability of results, many values are kept constant for all processes, for example the self-generating process rate is set to be 3. The default coordination delay is chosen to be 5, so the rates between self-generated and coordinated events are similar. The simulation is run for 2000 time-steps with a burn in period of 100, leaving 1900 time steps, which produces between 300-1000 events on average per process, giving sufficient data for most estimators to converge.

n**, number of target processes,** is chosen to be 10, so all evaluation results consider a system which contains 10 distinct point processes.

A is a $n \times n$ weighted adjacency matrix which describes the underlying coordination structure for directly coordinating events. Since we use a topological ordering to populate the event times for the n processes, we require that this adjacency matrix defines a directed acyclic graph. This matrix forms the basis for all coordination within the network, and represents the ground truth for directly coordinating relationships. A matrix with more edges represents a system where many processes directly coordinate, whereas the presence of paths represents processes which coordinate in sequence. This matrix is fixed for all simulations across all iterable variables. This was generated randomly by drawing edge weights from a exponential distribution with rate 0.2, keeping each possible edge in the acyclic graph with probability 0.5.

$\min w_i^\star$ is set to 0.1. This means the minimum probability for a process to not coordinate when in a coordinating state is 0.1.

$\boldsymbol{c}$ is set to $(5, 5)$ by default, so the process spends equal time in both the coordinating and non-coordinating states, transitioning every 5 timesteps.

$\boldsymbol{\lambda}$ by default is set at $\lambda^{(i)} = 3$.

$t_d(i, j)$ by default is set to a fixed coordination delay between parent and target coordinated events of 5 units.

$d(t) = 100 \exp\left(-\frac{1}{2}t\right)$. This time importance delay places higher priority on more recent events, so processes are more likely to coordinate with processes with more recent event times. This was chosen as many applications assume that more recent events

have higher priority or relevance in the case of coordination. This can be seen in works including Graham et al. (2020); Quaglio et al. (2018).

Times. The following time markers are used: $T_{\text{start}} = 0$, $T_{\text{end}} = 2000$, $T_{\text{burn-in}} = 100$.

Alongside these default values, four iterable parameters were used to represent a range of behaviour types. These values where chosen to explore how performance is impacted by the underlying coordination structure, the effect of the event generating process and changes between coordinating and non-coordinating sequences, and the time delay between parent and coordinating events. These ranges for the parameters are chosen to represent a range of behaviours and are described below.

e_{ij}**, the weights of the adjacency matrix for the underlying coordination network** is generated by generating a directed acyclic graph uniformly at random, with edge weights drawn from an exponential distribution. The weighted adjacency matrix is normalised so that the sum of each column and the corresponding $w_i^\star$ sums to 1. The normalisation procedure sets $w^\star = \min w_i^\star = 0.1$, so each value in the adjacency matrix is normalised by $\left(\dfrac{\sum_j A_{i,j}}{(1 - w^\star)} \mid \max_j \sum_j A_{i,j} \right)$. The primary function of this normalisation is to ensure that values can be compared across the network, rather than only on a local level. This is because when coordinated events are selected, the choice of parent process is made by normalising the in-degree edge weights for the target node.

This exponential distribution allows the variance of edge weights to be controlled, varying the underlying coordination network from one where the strength of all underlying coordination relationships are similar, to one where several of these relationships are much stronger than others. This represents a situation where many processes coordinate with equal probability, to a situation where particular processes tend to coordination with only a few of their parent processes.

This parameter is varied to evaluate performance against varying underlying network structures. Rates for the distribution of the exponentially distributed edge weights are taken from the set, $\exp(-i)$ for $i \in [0, 4]$. This produces edge weights with distributions shown in Figure 3.4. While similar behaviour occurs for $i = 3$ and $i = 4$, the variance of edge weights varies between these two cases is the variable of interest, and these two values represent systems with small, but differing variances. This range of values produces a spectrum of behaviour which corresponds to a system with several highly directly coordinated pairs of processes, to a system where many processes are weakly directly coordinated.

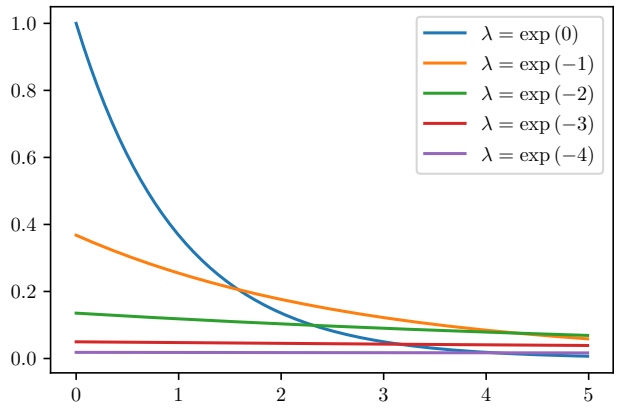


Figure 3.4: Distributions of edge weights in A before normalisation.

c_j, hidden Markov process describing the coordinating state of process i is a vector of tuples, with row j containing $\left(\frac{1}{c_{0,1}^{(j)}}, \frac{1}{c_{1,0}^{(j)}}\right)$. In the hidden Markov process, the transitions from a non-coordinating state to a coordinating state occur with rate $c_{0,1}$ and transitions back occur at rate $c_{1,0}$. The expected time which the process remains in each state is given by $\frac{1}{c_{n,m}}$, so these values can be experimentally obtained if data is available. For each iteration, these values are kept constant for all processes, and are taken to be all odd numbers between 1 and 21. This results in values $(1,1),(1,3)...,(3,1),...,(21,21)$.

$\lambda^{(i)}$, the self exciting event rate of the Poisson processes which generate the self-excited events. The expected time between events is given by $\frac{1}{\lambda^{(i)}}$. These values are unique for each process and are given by 0.2 + a realisation from a continuous uniform random variable with bounds 0 and i for $i = 1, 3, ..., 19$. This range of values produces self exciting event rates which occur with similar frequency to the default value for coordination time delay which ensures coordinating periods cannot be easily differentiated by simply looking at event counts.

$t_d(t)$ coordination time delay. This function gives the delay between the parent and coordinated event. The following functions are used to represent different types of coordination: $\delta \sim U(0, 10)$, $\delta \sim \text{Exp}(2)$, $\delta \sim \text{Exp}(5)$, $\delta = 1$, $\delta = 5$, $\delta = 10$.

3.6 Sample results

This section includes sample output from the evaluation procedure for a selection of heuristic and information-theoretic methods, denoted association measures. Complete sample results for the evaluation for each association method are given in Appendix A. Here, we will use these methods to emphasise features of the evaluation procedure and explore how it performs with respect to a variety of behaviour types.

The heuristic methods considered are co-occurrence counts, inter-event time statistics (median and variance), and an exponentially penalised time delay. These are all directed measures and are typically designed to capture directly coordinated behaviours. Generally, we expect these methods to perform better with shorter coordinated event time delays.

We also consider two information-theoretic approaches. The first, transfer entropy, is a directed measure frequently used in the literature (Shorten et al., 2021; Ver Steeg and Galstyan, 2012; Schreiber, 2000). The second, time-agnostic entropy, is an undirected measure which only uses orders of events, rather than their times.

In each example, a higher value represents better performance, or more similarity between the true and estimated values. Where these methods provide a dissimilarity rather than a similarity measure, the inverse values are considered. For these examples, dissimilarity measures a temporal distance, so it is appropriate for the reciprocal of the similarity value to be used. After this transformation is applied, the performance is evaluated as described in Section 3.5

For all association measures except for transfer entropy, these sample results use 250 iterations, although the variation in results is similar to the variation achieved with 50 iterations. This indicates that the variance which is present in these plots is due to the intentional randomness built into the simulation framework. The transfer entropy function was implemented directly from the open-source package CoTETE released alongside Shorten et al. (2021). This implementation was used to emphasise how this framework can be used to assess existing techniques, although computational limitations made it unreasonable to perform a large number of iterations due to the complexity of the calculation. Therefore, 50 iterations were used for this association measure.

3.6.1 Comparing evaluation heuristics

In this simulation framework, we propose two target behaviours of interest: directly coordinated behaviour and similarity between generating processes. These are two behaviours which are frequently targeted by researchers (Graham, 2020; Shorten et al., 2021).

The performance of measures can vary drastically when they are assessed against different

behaviour types so it is important to have a target behaviour in mind when evaluating association measures. The definitions of these target behaviours can also have an impact on results.

Performance for directly coordinating behaviours is assessed by measuring similarity with the underlying coordination network. This underlying coordination network describes the strength of the coordinated relationships. These relationship strengths are used to choose the parent process for a coordinated event, so directly correspond to more frequently coordinated events between the process pair. However, the number of directly coordinated events is also dependent on the amount of time each process spends in a coordinating state. For all iterations, the values characterising the underlying Markov process which dictates the coordination state are constant between all processes, so this factor should not affect the relationship between the number of coordinated events and the strength of the directly coordinated relationship.

Directly coordinated behaviour is characterised by well defined delays between pairs of events, and counting the number of co-occurrences is a logical choice to detect direct coordination which has a bounded delay. In Figure 3.5 (i) the best performance occurs with a fixed coordination delay of 1 unit. Performance drops for coordination delays which result in fewer directly coordinated event pairs which fall inside the co-occurrence window. Interestingly, performance is comparably low for uniformly distributed delays and fixed delays which fall outside the target window, however the low proportion of coordinated target events may explain this.

These trends do not continue when assessing similarity between generating processes (ii), with a co-occurrence window of size 1 not able to provide much discrimination between different coordinated event delays.

In general, we expect association measures which target directly coordinated behaviours to perform best when more time is spent in a coordinated state. We can see this in Figure 3.5 (iii), which shows results for an association measure using a co-occurrence window of size 1. This measure best detects directly coordinated behaviours when the most time is spend in a coordinating state, with worsening performance as a higher proportion of time is spent not coordinating.

When we consider similarity between generating processes, using a co-occurrence window may not be a natural choice of association measure. This is evident in the comparably worse performance for detecting this similarity (Figure 3.5 (ii) and (iv)). When assessing generating similarity, slightly better performance occurs when a higher proportion of time is spent in either a non-coordinating or coordinating state. Since the association measure is considering co-occurrence windows, this does not target any behaviours associated with similarity between generating processes, this behaviour is expected.

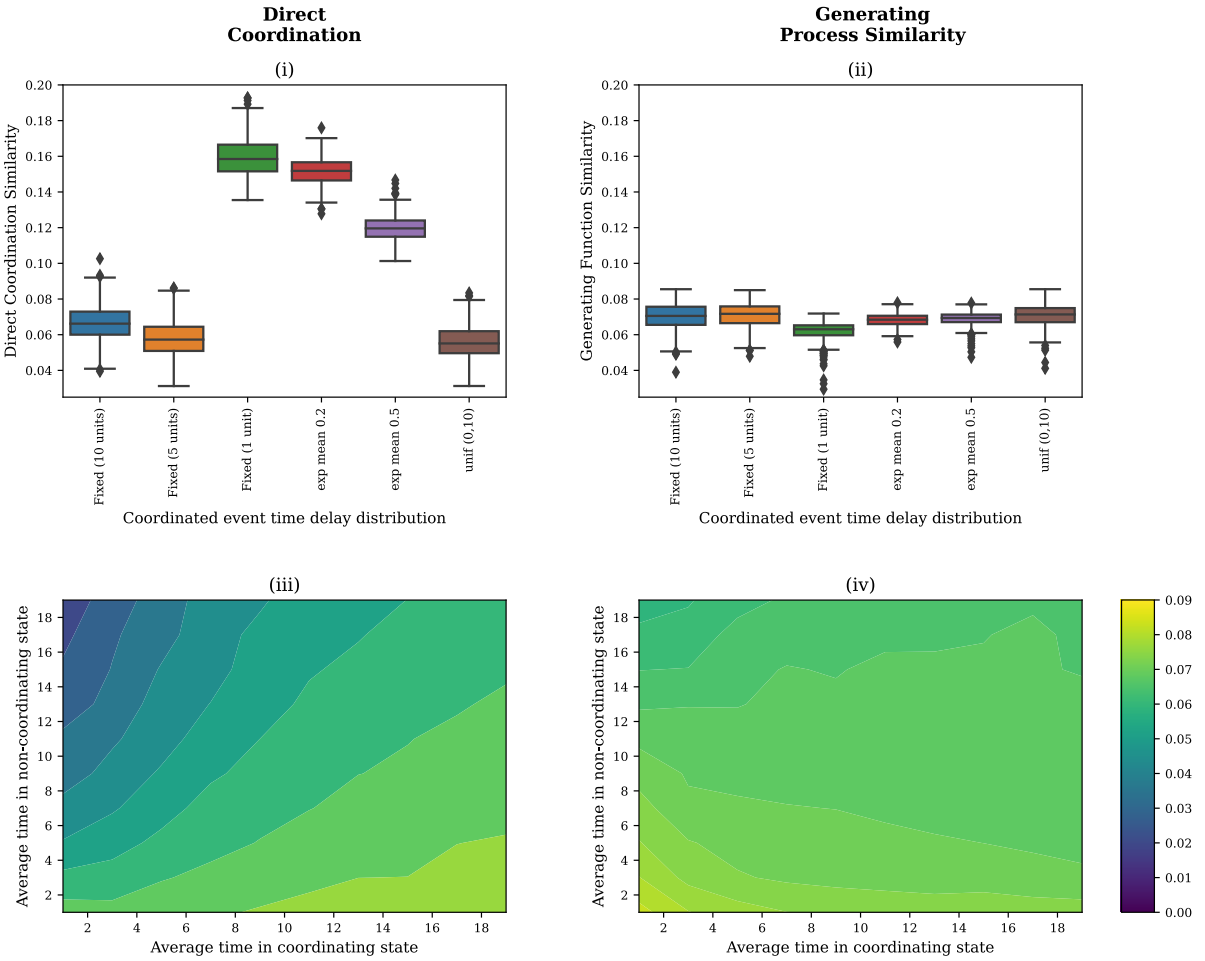


Figure 3.5: Performance for association method using a co-occurrence window of size 1 for measuring directly coordinated behaviours (i, iii) and measuring similarity between generating processes (ii, iv). The best performance occurs with a fixed coordination delay of 1 unit (I). Performance drops for coordination delays which result in fewer directly coordinated event pairs which fall outside the co-occurrence window. Interestingly, performance is comparably low for uniformly distributed delays and fixed delays which fall outside the target window, although the low proportion of coordinated target events from a uniform delay would be low. These trends do not continue when assessing similarity between generating processes (ii), with the association method using a co-occurrence window of size 1 not able to provide much discrimination between different coordinated event delays. The differences between performance for each of the target behaviours is also evident in how the methods perform when processes spend different amounts of time in coordinating states. In (iii), the association method using a co-occurrence window of size 1 is able to best detect directly coordinated behaviours when more time is spend in a coordinating state. (iv) shows performance is not affected greatly by the amount of time spent in either a non-coordinating or coordinating state, with a slight trend for better performance for a higher proportion spend in a coordinating state.

These trends continue when considering an association mesure based on the median inter-event time Figure 3.6. When considering directly coordinated behaviour, the measure performs well when processes spend a higher proportion of time in a coordinating state, but also performs well when a large amount of time is spent in a non-coordinating state. In this second case, most events will be generated independently of all other processes. However, when more events occur within a process, it is more likely that we observe an event within the coordination window. The reliability of this measure decreases as more events occur over a shorter time period, as more events are falsely attributed to being coordinated events. This will lead to a similar inter-event time for all processes due to the choice of a constant rate for these self-generated events. This can lead to cases when events occur within a short period consistently, leading to a lower median inter-event time which translates to better performance.

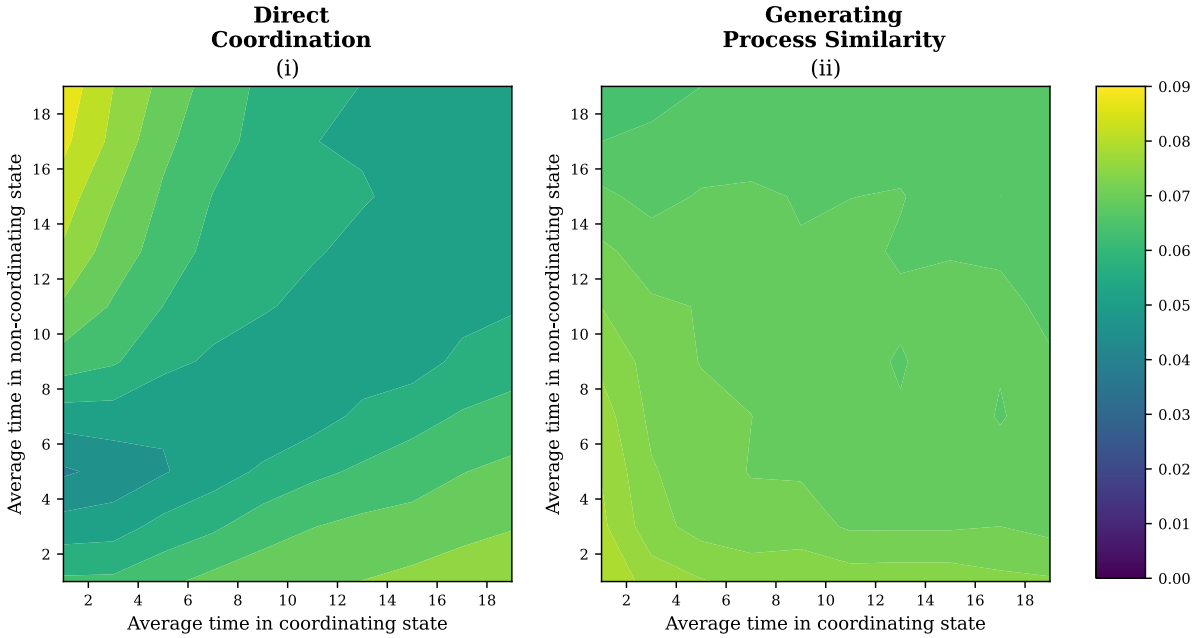


Figure 3.6: Performance for an association measure which considered the median inter-event times between pairs of processes. In (i), performance is assessed against detecting directly coordinated behaviours and in (ii) performance is considered for detecting similarity between generating functions.

When considering generating similarity (Figure 3.6 (ii)), the evaluation procedure uses a pair of rates for each process. The association measure performs well when small amounts of time are spent in each state. Performance is better when a higher proportion of time is spend in one state than the other, with slightly better performance for more time spend in the coordinating state. This may be due to behaviours between processes being more

similar when their underlying coordination states are less variable, as processes are more likely to be in the same coordination state and therefore exhibiting similar behaviour. This effect is an unintentional byproduct of the choice of technique to identify similarity between generating processes, and highlights the importance of proper validation of association measures.

Coordination delay functions

A feature of directly coordinated events is a well defined inter-event time. In this simulation framework, these inter-event times are characterised by a coordination delay function, and a variety of functions are used to explore the how different association measures perform.

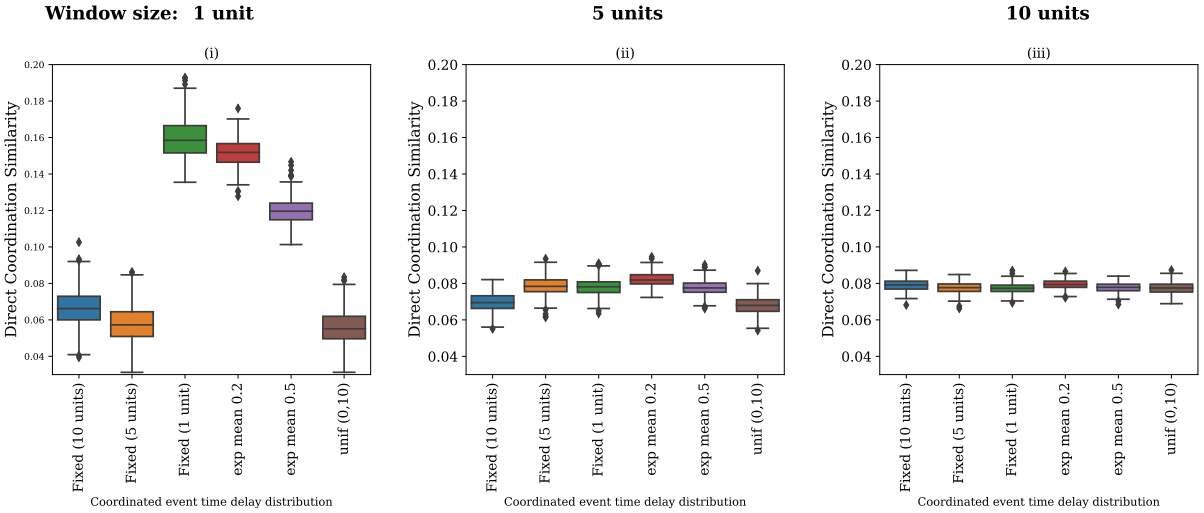


Figure 3.7: To understand how the length of coordination window impacts performance against coordinated time delay functions when detected directly coordinated behaviours, approaches with a window size of 1 unit (i), 5 units (ii) and 10 units (iii) were considered. It is expected that co-occurance counting will perform well when coordinated event delays are drawn from a distribution with a matching window size when considering directly coordinated behaviours. For a co-occurrence window of size 1, we can see best performance for a fixed delay of 1 unit and an exponential delay with lower mean delay. For the delay of 5 units, we can see this trend continues, with good performance for both exponential delays and a fixed coordination delay of 1 and 5 units. For a co-occurrence window of 10 units, each function has comparable performance, which is expected as most coordinated event delays should fall within 10 units given the considered parameter range.

We expect that measures which assign a high coordination likelihood with a condition which matches the coordination delay function will perform better at detecting these directly coordinated processes.

Co-occurrence counting is based on the idea that a window of coordination exists, where a pair of events are coordinated if they occur within some time window. This example has been chosen as it provides a straightforward comparison between the coordination delay function and the condition of the association measure. By counting the number of events in a target process that occur within the specified coordination window after an event in the source process, we aim to measure how frequently a pair of processes coordinate relative to the number of events within the target process.

To understand how the length of coordination window impacts performance, approaches with a window size of 1 unit Figure 3.6 (i), 5 units Figure 3.6 (ii) and 10 units Figure 3.6 (iii) were considered. By changing the size of this coordination window, we expect these metrics to perform best in situations where most coordinated events fall within the target window when direct coordination is assessed.

For an association measure which uses a co-occurrence window of size 1, Figure 3.6 (i), the best performance occurs with a fixed delay of 1 unit, or with a stochastic delay which has a high proportion of short coordination delays. This second case occurs for the exponentially distributed coordination delays of with mean 0.2 and 0.5. Performance is slightly better for the exponentially distributed delay with mean 0.2, which is expected as a higher proportion of delays will fall within the target window of size 1.

For a co-occurrence window of size 1, we see that co-occurance counting can perform well when coordinated event delays are drawn from a distribution with a matching window size. This is not the case for approaches with a window size of 5 (Figure 3.6 (ii)) and 10 (Figure 3.6 (iii)). In both of these cases we see less discrimination between different coordination delay functions, indicating that best performance is achieved for a closer match between the distribution of coordination delays and the association measure. When the window of the association measure is too large, we increase the likelihood of falsely labelling event pairs as coordinated, and over-counting the number of coordinated events.

3.6.2 Event counts

In these sample outputs, we consider 10 processes for 1900 time steps with the number of events within each process impacted by the input parameters. The number of events ranges from close to 100 to around 5000. With a lower number of events, there is less information available for each association measure, leading to higher variance in the final estimate. Association measures may also face convergence issues for smaller event counts. This can be a particular problem for information-theoretic measures, which use approxi-

mations based on fitting a probability density function or the spread of samples within a sequence.

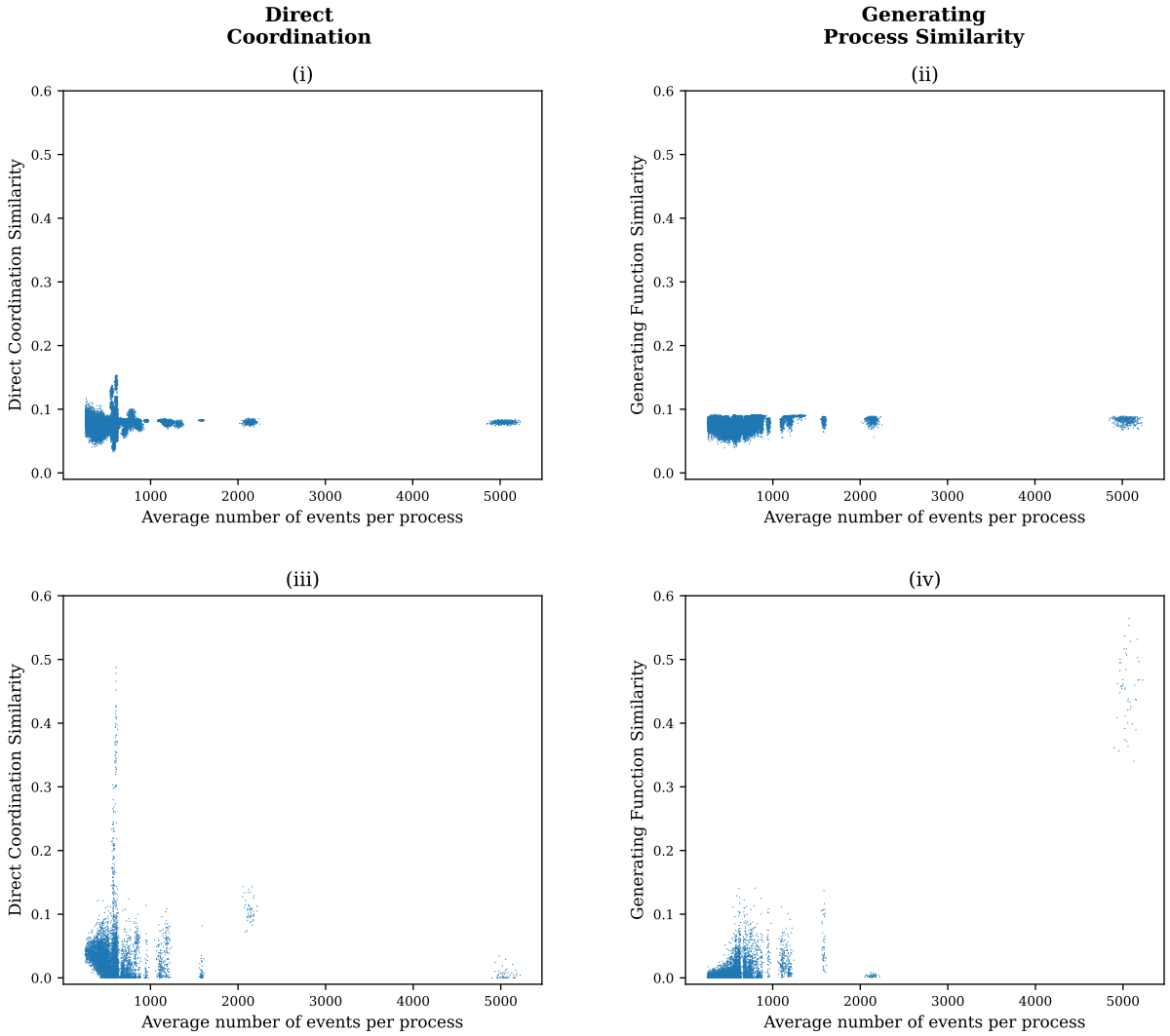


Figure 3.8: Each of these graphs show the affect of the average number of events per process on the performance of the association measure. (i) and (ii) are results for detecting direct coordination and similarity for an association measure using a co-occurance window of size 10, (iii) and (iv) show these quantities for an association measure which uses the pairwise transfer entropy between processes.

In Figure 3.8, (i) and (ii) show how performance is related to the average number of events in each process for an association measure using a co-occurance window of size 10. (iii) and (iv) show these quantities for an association measure which uses the pairwise

transfer entropy between processes. In (iii) and (iv), we can see how a trend appears to exist between performance and the average number of processes, which indicates that convergence issues may be at play. In (i) and (ii), a heuristic measure, there does not appear to be a trend in the performance, although the spread may decrease for larger event counts. Generally, more events leads to less variance in performance, as stochastic features of the simulation framework may lead to variations in simulated event times for smaller event counts.

3.6.3 Average time between self excited events

The average time between self-excited events impacts the number and rate of events which occur when a process is not in a coordinating state. This can lead to larger differences between the numbers of events which occur in non-coordinating and coordinating states. Since the default average time spend in the non-coordinating and coordinating states is 5 units, and the range of values considered for the average time between self generated ranges from 0 to 10, we expect that this may improve performance for generating process similarity for extreme values. This does not seem to affect the results, as there does not appear to be a large difference in performance for detecting generating similarity for any subset of average self generated event times Figure 3.9 (ii).

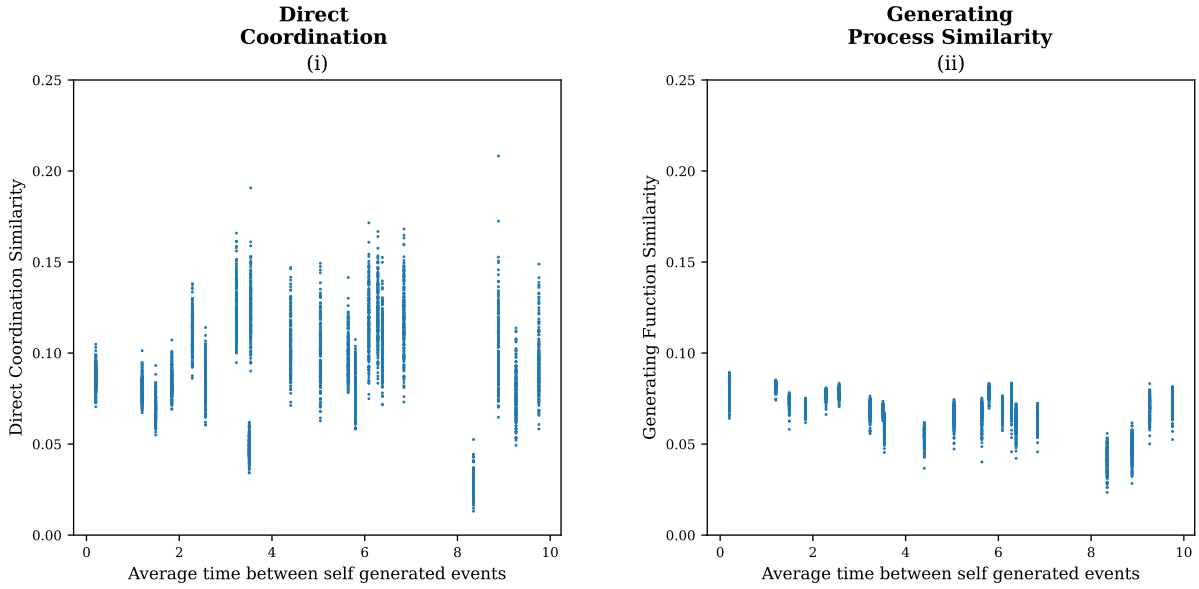


Figure 3.9: Relationship between detecting directly coordinated behaviour (i) and similarity between generating processes (ii) for an association measure which uses an exponentially penalised inter-event time. There is more variance across the range of values for performance for directly coordinated events, although neither show a strong trend.

There seems to be a slight trend for better performance against directly coordinated behaviour types for larger average times between self-excited events, which may be due to fewer self-excited events occurring (Figure 3.9 (i)). This would lead to a higher proportion of coordinated events. However the variance in performance is quite large, indicating that the choice of default parameters may have a large impact on the relationship between the exponentially penalised inter-event times and the theoretical coordinated relationship strengths.

3.6.4 Distribution of coordinated relationship strengths

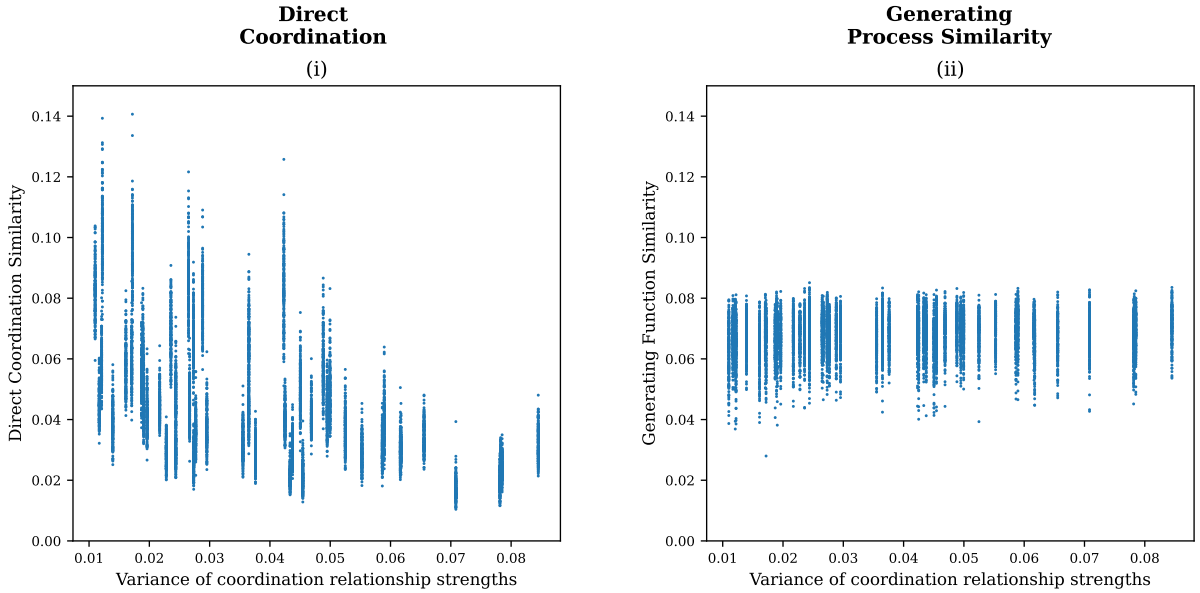


Figure 3.10: Both (i) and (ii) show the performance of a median inter-event time association measure against the variance of coordination relationship strengths. Figure (i) shows performance for directly coordinated processes and (ii) shows performance for detecting generating similarity. There is a general downward trend in (i), although the variance also seems to decrease. As the relationship strengths are drawn from an exponential distribution and normalised, as variance increases, there are fewer, but stronger relationships. This indicates that the median inter-event time is best able to detect direct coordination when there are fewer pairs of coordinating processes rather than many coordinating processes.

The structure of the underlying coordination network impacts how association measures perform. For these sample results, these relationship strengths were drawn from an exponential distribution, and these strengths are normalised by the sum of the total in degree

for each process and a likelihood of no coordinated event, so a higher variance of coordination relationship strengths indicates there are fewer coordinated relationships but the strengths of these relationships is generally higher. This leads to a simulation with a higher number of coordinated events between these pairs of strongly related processes. Inversely, for a lower variance, a greater number of weaker relationships is present, and coordinated events can occur with many different parent processes. Generally, this leads to a lower proportion of coordinated events with each individual parent process, which can make it more difficult to use these events to identify the relationship strengths.

In Figure 3.10 (i), this appears as a larger variance in performance for detecting direct coordination for lower variance in the coordination relationship strengths. When there is lower variance in the coordination relationship strengths, the best performance is occurring, indicating that the median inter-event time may be a suitable measure for these settings, potentially due to its robustness to outlying values. When considering generating process similarity Figure 3.10 (ii), there does not seem to be any trend between performance and variance in the coordination relationship strengths. This is likely due to the simulation set-up where event rates in coordinating states are fixed due to the simulation procedure and only vary if events are generated outside the current coordination window.

3.7 Conclusion

This work complements the survey given in Chapter 2, outlining a simulation procedure and evaluation framework for approaches to measure the pairwise strength of associations between point processes. The simulation procedure is robust and flexible. It is able to capture a range of behaviours from point processes and can easily be adapted to physical constraints or to accommodate prior beliefs.

Since the simulation procedure calculates all events for each process, complexity increases as the number of processes increases. All calculations are done sequentially, so the complexity of the algorithm may be a limiting factor in some applications or where large data sets are required. This efficiency trade off was made to ensure the procedure is as flexible as possible and can easily be adapted to suit the needs of researchers from a range of domains. Future work may look to optimise some of the coding procedures used to overcome these challenges.

Alongside data simulation, this framework is designed to evaluate association estimation strategies for performance against a range of features. As the simulation framework uses an underlying coordination structure, evaluation for detecting directly coordinating processes follows naturally by comparison with the underlying coordination network, although a method is also provided to consider similarity between generating processes.

This second approach approximates a processes generating structure using two Poisson processes, although information about the amount of time spent in each state is lost. Performance can be affected if a small amount of time is spend in a single state, so the sample size for rate estimation is smaller, resulting in a lower quality estimate.

This evaluation structure is tested using data simulated to imitate settings which range from several highly coordinated directly coordinating processes which do not self-generate events, to more complex settings with varied levels of direct coordination between processes and stochastic generating processes. This simulation tests six heuristic measures, co-occurance counting with window sizes of 1, 5 and 10, an exponentially penalised inter-event time, and the median and variance of the inter-event times; and two information-theoretic estimators, transfer entopy and time-agnostic entropy; against detecting directly coordinated relationships and similarity between generating processes. These sample results are used to discuss the limitations of the proposed evaluation procedures.

In practice, this procedure can be used in conjunction with prior knowledge about a particular data setting to evaluate a range of potential association measures, enabling a meaningful, robust and accurate method to be chosen. This is highly dependent on many factors: the scale of the system, what types of association occur, how events are generated and how strong signals of association can be.

Future work will extend the evaluation framework beyond pairwise relationships summarised with a single value, considering methods to measure similarity between functions and distributions to accommodate stochastic and machine learning methods.

Chapter 4

Publication: #IStandWithPutin versus #IStandWithUkraine: The interaction of bots and humans in discussion of the Russia/Ukraine war

4.1 Summary

This publication focuses on using information-theoretic and linguistic techniques to gain deeper insights into discussion on Twitter around the 2022 Russian invasion of Ukraine. The data set used in this analysis comprises of 5,203,746 Tweets collected between February 23rd 2022 00:00:00 UTC and March 8th 2022 23:59:59 UTC which contain case-insensitive versions of the hashtags:

- #(I)StandWithUkraine,
- #(I)SupportUkraine,
- #(I)StandWithZelenskyy
- #(I)StandWithRussia,
- #(I)SupportRussia, and
- #(I)StandWithPutin.

Here, (I)StandWithUkraine indicates that both #StandWithUkraine and #IStandWithUkraine are included in the set of query hashtags.

The goal of the publication is to understand and quantify the influence which bot-like accounts have online by and quantifying their impact on linguistic features of the general conversation. This was achieved by extending existing information-theoretic and linguistic analysis techniques. The technical contributions of this work will be highlighted for this thesis. To achieve these outcomes, information flow estimators were used to establish a significant influence network between groups of accounts. Existing information flow estimators were extended to consider influences between groups of accounts online. The estimator was also equipped with an empirical significance test. Once significant information flows were established within the data set, the impact of increased bot activity on sentiment and linguistic features was considered.

4.2 Mathematical background: entropy estimators in social media

In social media, it is natural to construct networks of social interactions, where a node represents a user and an edge represents a connection between the pair of users. In some cases, an edge can represent an interaction, community, url or piece of content shared between the two users. It can be insightful to construct edges representing an information flow between accounts (Bagrow et al., 2019; Pond et al., 2020; South et al., 2022).

Understanding the information flow over the network allows the structure of the underlying network to be probed and understood, as well as providing a tool to interrogate influence within the online environment. The Shannon cross entropy has been used in online social networks to describe the information flow between two accounts, by measuring the predictability between them (Bagrow et al., 2019; Pond et al., 2020; South et al., 2022).

For a discrete random process, X, which takes values from a finite alphabet, let X_a^b denote the string $(X_a, X_{a+1}, ..., X_b)$. For such a process, the Kontoyiannis temporal entropy rate estimator (Kontoyiannis et al., 1998) is given by,

$$H(X) = \lim_{n\to\infty} \frac{n \log n}{\sum_{i=1}^{n} \Lambda_i^i},$$

where Λ_i^i denotes the length of the shortest substring, l starting at index i which does not appear in the first $i + l - 1$ symbols.

This notion was extended in work by Bagrow *et. al.* to calculate the cross-entropy (Bagrow et al., 2019). The cross entropy between two sources describes the amount of information required to describe one source given the other. The Kontoyiannis estimator, which is better able to capture longer term dependencies within the data than frequency based

estimators can be extended to measure the cross entropy between two sources (Bagrow et al., 2019). To do so, it uses the history of one sequence and the future of the other to gain insight into the quantity of information shared between the two processes over time.

This time-synced cross-entropy rate estimator between a target, $\mathcal{T}$, and source, $\mathcal{S}$ at an instant i in time is given by,

$$h(\mathcal{T}||\mathcal{S}) = \frac{N_{\mathcal{T}} \log_2 N_{\mathcal{S}}}{\sum_{i=1}^{N_{\mathcal{T}}} \Lambda_i(\mathcal{T}|\mathcal{S}_{\leq t(T_i)})}, \tag{4.1}$$

where N_S and N_T are the number of symbols written by the source and target, respectively, and Λ_i^i denotes the length of the shortest substring, l starting at index i which does not appear in the first $i + l - 1$ symbols.

The cross entropy rate between two sources can be inflated by a larger vocabulary or more complex language structure used by one group. A proposed normalisation from South et al. (2022), mitigates this issue and also allows for more meaningful comparison of information flows between sources. The normalised net information flow measure is shown in 4.2.

$$\Delta(\mathcal{T}||\mathcal{S}) = \frac{\hat{h}(\mathcal{T}||\mathcal{S})}{\sum_X \hat{h}(\mathcal{T}||X)} - \frac{\hat{h}(\mathcal{S}||\mathcal{T})}{\sum_X \hat{h}(\mathcal{S}||X)}. \tag{4.2}$$

In the publication, we use this measure to estimate the influence of account groups.

4.2.1 Measuring aggregated information flow

In the given data set, we have N accounts denoted $X_1, ..., X_N$, each with a group label l_i, for $i = 1, ..., N$. These labels are derived from a number of account features and describe if an account has a national lean or appears to be bot-like. Each of these accounts is associated with some set of Tweets, which can be represented by $T^{(i)}$. To calculate information flows, the content and timing of each of these Tweets are used to establish directional information flow measures and measure the normalised cross entropy.

To estimate the information which flows between a pair of these groups containing accounts with labels l_i and l_j, we consider the set of all content associated with each of these groups, given by $\{T^{(k)}|l_k = l_i\}$ and $\{T^{(k)}|l_k = l_j\}$. This process makes the assumption that within each group, the originating account does not affect the flow of information from the group. To make an entropy estimate, we assume that the overall group activity comes from a single distribution. This first assumption may be flawed in some cases, but

this work aims to consider the overall group behaviour, and as groups are constructed using account features, this assumption likely holds.

4.2.2 An empirical significance test of information flow

Where a true null distribution is infeasible or unknown, it is possible to construct an empirical significance test. This empirical significance test of the information flow between a pair of account groups uses repeated sampling to approximate the p-value by testing if the observed value has a rank equal to or greater than the critical value. This critical value is chosen by selecting a proportion of unique values. In this case, this proportion is taken to be 0.05. A further discussion of some limitations and considerations of empirical significance tests is given in North et al. (2002).

To estimate the significance of each information flow estimate, the account labels are shuffled and the content is re-aggregated into the new groups. This process uses the null hypothesis that the distribution of each group's behaviour is not unique. Information flows are calculated on these new groups, and each pairwise value is stored to calculate the rankings required for the empirical significance test. The difference in median information flows for the two groups is used to establish significance for each considered pair. A sample of these null distributions alongside observed values (red) is given in Figure 4.1.

If the observed value was greater than or equal to the critical value, that information flow was retained. These flows were used to construct the final network of information flows (Figure 4 in publication *The interaction of bots and humans in discussion of the Russia/Ukraine war*).

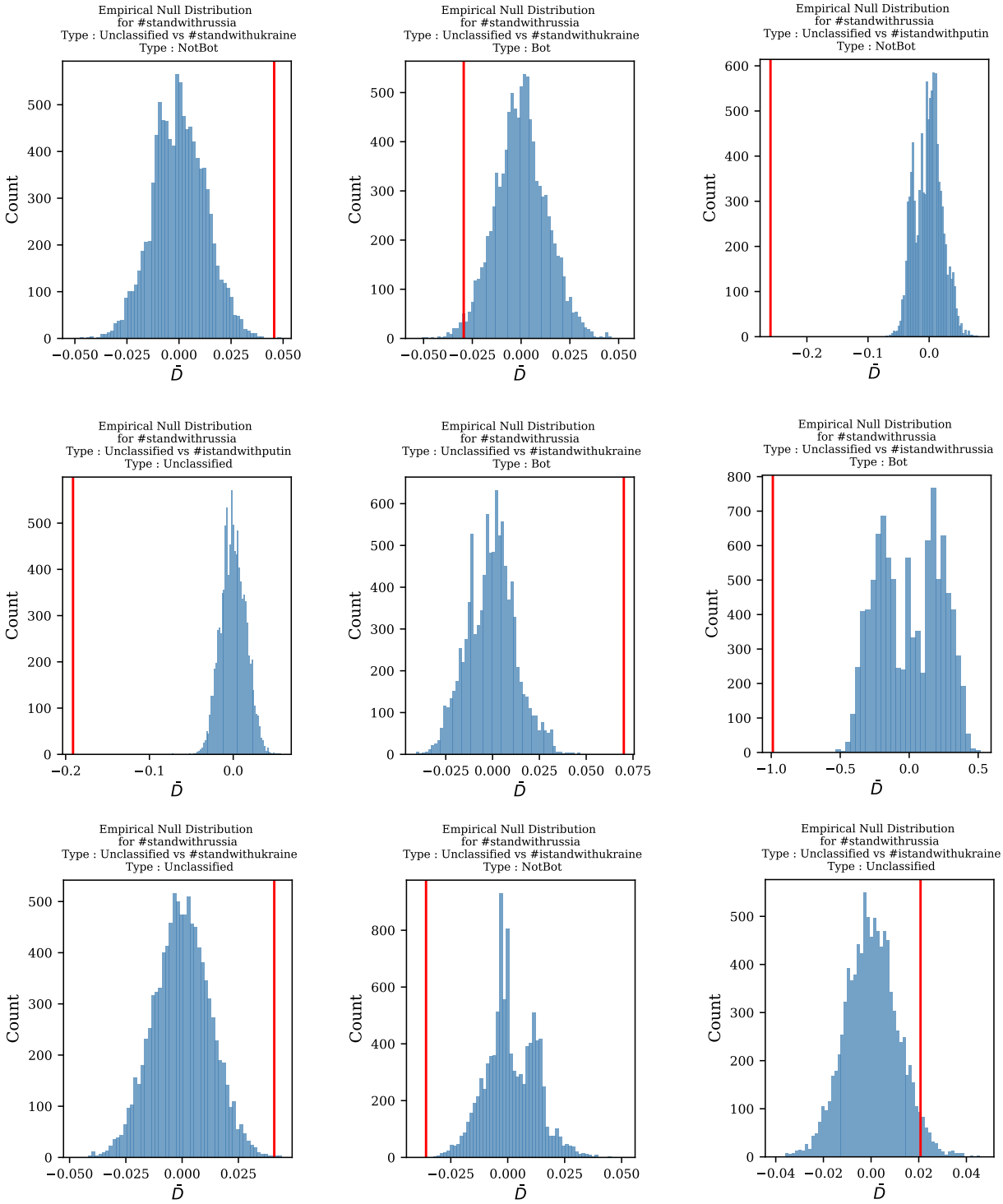


Figure 4.1: Results from the empirical significance test for the unclassified pro-Russian account group.

4.3 Statement of Authorship: #IStandWithPutin versus #IStandWithUkraine: The interaction of bots and humans in discussion of the Russia/Ukraine war

Title of Paper	#IStandWithPutin versus #IStandWithUkraine: The interaction of bots and humans in discussion of the Russia/Ukraine war
Publication Status	Published
Publication details	Smart, B., Watt, J., Benedetti, S., Mitchell, L. & Roughan, M. (2022), #IStandWithPutin Versus #IStandWithUkraine: The Interaction of Bots and Humans in Discussion of the Russia/Ukraine War, in F. Hopfgartner, K. Jaidka, P. Mayr, J. Jose & J. Breitsohl, eds, 'Social Informatics', Vol. 13618, Springer International Publishing, Cham, pp. 34–53.

Principal Author

Name of Principal Author (Candidate)	Bridget Smart		
Contribution to the Paper	This work is a joint publication, with most of the linguistic conversational analysis conducted jointly. The collection and cleaning of the data from Twitter, and exploratory data analysis were conducted jointly with Joshua Watt. I contributed to the conception and design of the work, data collection and analysis, writing and editing. For the purpose of this thesis, my individual contributions are: • Aggregating the content produced by each account group and applying the non-parametric information flow estimator to this content (Section 5.2). • Implementing an empirical significance test for the aggregated information flows by shuffling labels to establish an empirical null distribution (Section 5.2). • Exploring the relationship between bot-like activity and non-bot sentiment within the data set by constructing a time series using the product of the complement of the overall bot activity and the compound sentiment (Section 6.1). This is analysed using time series analysis techniques to explore the relationship between these two variables. • Visualisations for figures 1, 2, 3, 4 and 5.		
Overall percentage (%)	45%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	26/01/2023

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

I the candidate's stated contribution to the publication is accurate (as detailed above);

ii permission is granted for the candidate in include the publication in the thesis; and

iii the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Joshua Watt		
Contribution to the Paper	40%		
Signature		Date	22/1/2023

Name of Co-Author	Sara Benedetti		
Contribution to the Paper	5%		
Signature		Date	24/01/2023

Name of Co-Author	Lewis Mitchell		
Contribution to the Paper	5%		
Signature		Date	

Name of Co-Author	Matthew Roughan		
Contribution to the Paper	5%		
Signature		Date	

arXiv:2208.07038v2 [physics.soc-ph] 20 Aug 2022

#IStandWithPutin versus #IStandWithUkraine: The interaction of bots and humans in discussion of the Russia/Ukraine war

Bridget Smart[1 [0000−0002−0910−9470]], Joshua Watt[1 [0000−0001−7899−1244]], Sara Benedetti[1 [0000−0003−3514−797X]], Lewis Mitchell[1 [0000−0001−8191−1997]], and Matthew Roughan[1 [0000−0002−7882−7329]]

[1] The University of Adelaide
{bridget.smart, joshua.watt, sara.benedetti, lewis.mitchell, matthew.roughan} @adelaide.edu.au
set.adelaide.edu.au/mathematical-sciences

Abstract. The 2022 Russian invasion of Ukraine emphasises the role social media plays in modern-day warfare, with conflict occurring in both the physical and information environments. There is a large body of work on identifying malicious cyber-activity, but less focusing on the effect this activity has on the overall conversation, especially with regards to the Russia/Ukraine Conflict. Here, we employ a variety of techniques including information theoretic measures, sentiment and linguistic analysis, and time series techniques to understand how bot activity influences wider online discourse. By aggregating account groups we find significant information flows from bot-like accounts to non-bot accounts with behaviour differing between sides. Pro-Russian non-bot accounts are most influential overall, with information flows to a variety of other account groups. No significant outward flows exist from pro-Ukrainian non-bot accounts, with significant flows from pro-Ukrainian bot accounts into pro-Ukrainian non-bot accounts. We find that bot activity drives an increase in conversations surrounding angst (with $p = 2.450 \times 10^{-4}$) as well as those surrounding work/governance (with $p = 3.803 \times 10^{-18}$). Bot activity also shows a significant relationship with non-bot sentiment (with $p = 3.76 \times 10^{-4}$), where we find the relationship holds in both directions. This work extends and combines existing techniques to quantify how bots are influencing people in the online conversation around the Russia/Ukraine invasion. It opens up avenues for researchers to understand quantitatively how these malicious campaigns operate, and what makes them impactful.

Keywords: Bot Nets · Information Flow · Sentiment Analysis · Linguistic Analysis · Disinformation Campaigns · Influence Campaigns · Twitter.

1 Introduction

Social media is a critical tool in information warfare, playing a large role in the 2022 Russian invasion of Ukraine [6,30]. Disinformation and more generally *reflexive control* [38] have been used by Russia and other countries against their enemies and internally for many years [10]. A relative newcomer in this space – Twitter – has already been extensively used for such purposes during military conflicts, for instance in Donbass [10], but its role in conflicts is evolving and not fully understood. Both sides in the Ukrainian conflict use the online information environment to influence geopolitical dynamics and sway public opinion. Russian social media pushes narratives around their motivation, and Ukrainian social media aims to foster and maintain external support from Western countries, as well as promote their military efforts while undermining the perception of the Russian military. Examples of these narratives include allegations: that Ukraine was developing biological weapons [40], that President Volodymyr Zelenskyy had surrendered [5,16], and that there is a sustained campaign showing the apparent success of 'The Ghost of Kiev' [18]. Some of the information being pushed is genuine, and some is malicious. It is not easy to discriminate which is which.

Understanding and measuring information flows and various language features has previously allowed researchers to understand community dynamics and identify inauthentic accounts and content [36,31,1]. Here we apply and extend these techniques to understand and quantify the influence of bot-like accounts on online discussions, using Twitter data focussed on the Russian invasion of Ukraine. In essence we seek to determine whether the malicious influence campaigns work as intended.

Our dataset consists of 5,203,764 tweets, retweets, quote tweets and replies posted to Twitter between February 23rd and March 8th 2022, containing the hashtags `#(I)StandWithPutin`, `#(I)StandWithRussia`, `#(I)SupportRussia`, `#(I)StandWithUkraine`, `#(I)StandWithZelenskyy` and `#(I)SupportUkraine` [39].

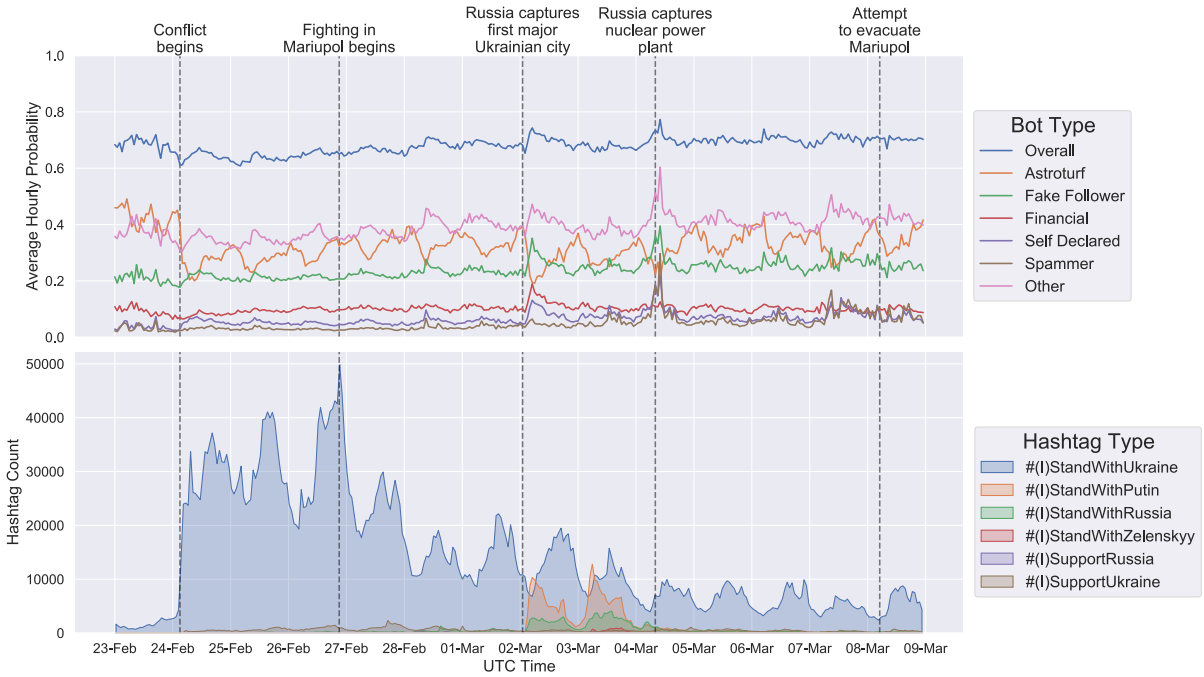


Fig. 1: Average hourly probabilities of bots tweeting query hashtags (top). Hourly frequency of the query hashtags (bottom). The time period we consider is the first fortnight after Russia's invasion of Ukraine. Both plots also include five significant events over this time period. Note that the query hashtags can be found in Section 3. We can observe a significant spike in the bot activity of several bot types on the 2nd and 4th of March. The spike in bot activity on the 2nd of March aligns with Russia's capture of Kherson, and also aligns with a significant increase in pro-Russia hashtags. This spike in activity was due to an increase in activity of pro-Russian bots – likely used by Russian authorities. The spike in bot activity on the 4th of March aligns with when the use of pro-Russia hashtags diminished, but also when Russia captured the Zaporizhzhia nuclear power plant. This spike was due to an increase in activity of pro-Russian bots (before being removed) and an increase in activity of pro-Ukrainian bots – likely by pro-Ukrainian authorities in response to Russian bots.

See Section 3 for further details. A summary plot of the data is shown in Figure 1. The figure also shows a measure of the proportion of bot traffic over the same time period, as estimated by the bot-detection tool Botometer [33].

In all time series figures, we present five significant events that provide context for our findings: when the conflict begins (24th February 2022), when the fighting in Mariupol begins (26th February 2022), when Russia captures Kherson (2nd March, 2022), when Russia captures the Zaporizhzhia nuclear power plant (4th March 2022) and when Ukrainian authorities first attempt to evacuate Mariupol (8th March 2022). These events are linked to noticeable changes in the volumes of related tweets, and in our analysis we delve deeper to understand how information is flowing. As a result, we learn how bots are influencing the online conversation by measuring what communities are talking about online, and how this discussion evolves. We use lexicon and rule based techniques to create an approach that is robust, transferable and able to be quickly applied to large volumes of data.

We employ time-series analysis techniques to understand how bot-like activity impacts the wider group of participants, by measuring linguistic content, sentiment and their lagged effect on future discussions. We use the Linguistic Inquiry and Word Count (LIWC; pronounced "Luke") [29] and Valence Aware Dictionary for Sentiment Reasoning (VADER) [13], dictionary based models to measure the linguistic features and sentiment of our dataset. To measure bot activity, we classify a random sample of 26.5% of accounts which posted at least one English language Tweet in the dataset using Botometer [33].

This work extends existing techniques to understand how bot-like accounts spread disinformation on Twitter and measure the effect of these malicious campaigns. The main contributions are:

– An extension of existing information flow techniques to examine aggregated group activity. We establish statistical significance of information flows between accounts grouped by national lean and account type. The highest information flows are out of pro-Russian non-bot accounts. Information flows into non-bot account groups are only significant for balanced and pro-Ukraine accounts, with pro-Russian non-bot accounts only exhibiting a net outward information flow of information.

– We establish a significant relationship between bot activity and overall non-bot sentiment (with $p = 0.000376$), but find this relationship is significant for both positive and negative lags, indicating there may be confounding factors.
– An analysis of the effect which bot activity has on emotions in online discussions around the Russia/Ukraine conflict. We find that bots significantly increase discussions of the LIWC categories: Angst, Friend, Motion, Time, Work and Filler. The strongest relationship is between Self Declared bot activity and words in the 'Work' category (with $p = 3.803 \times 10^{-18}$), which includes words relating to governance structures like 'president' and 'government'.
– A dataset[1] of Twitter users who participated in discussions around the Russian Invasion of Ukraine [39].

2 Related work

Many works have analysed bot-like accounts on social media [25,15,9]. Authors have shown bots are present in social networks, especially with regard to political campaigns/movements [25]. Keller and Klinger [15] showed social bot activity increased from 7.1% to 9.9% during German election campaigns, using bot probabilities before and during the election campaign. Furthermore, Stella et al. [37] showed bots increase exposure to negative and inflammatory content in online social systems. These authors used various information networks to find that 19% of overall interactions are directed from bots to humans, mainly through retweets (74%) and mentions (25%) [37]. A more socially-focused approach by Cresci et al. [9] measured Twitter's current capabilities of detecting social spambots. They assess human performance in discriminating between genuine accounts, social spambots, and traditional spambots through a crowdsourcing campaign. Notably, these works focus on analysing structural aspects of communication networks between bot and non-bot accounts, whereas we will examine information flows directly, using the full content of tweets.

Information flows in online social networks have been used to reveal underlying network dynamics, and employed to understand how individual users exert influence over one another online. Typically these flows are measured using statistical and information-theoretic measures of information flows [1,31,36], to understand if significant information flows exist between groups, particularly between bot and non-bot accounts. In social media, existing approaches only consider account-level information flows, while our work considered aggregated information flows.

The use of bots by Russian authorities has been widely observed: *e.g.,* Collins [7] found 5,000 bots were pushing protests against *Russiagate haux,* a political event concerning relations between politicians from US and Russia; and Shane [35] suggested Russia created 'Fake Americans' to influence the 2016 US election. Moreover, Purtill [32] found that Russia had a massive bot army in spreading disinformation about the Russia/Ukraine conflict. Muscat and Siebert [22] have suggested that both Ukraine and Russia are utilising bot armies in their cyber warfare. However, the extent to which these bots drive particular discussions and influence the behaviour of humans on social media during the Russia/Ukraine conflict is relatively unexplored. We aim to address this question through our analysis of information flows, sentiment, and linguistic features.

3 Data collection and preprocessing

We used the Twitter API (V2) to collect all tweets, retweets, quotes and replies containing case-insensitive versions of the hashtags `#(I)StandWithPutin, #(I)StandWithRussia, #(I)SupportRussia, #(I)StandWithUkraine, #(I)StandWithZelenskyy` and `#(I)SupportUkraine` [39]. These Tweets were posted from February 23rd 2022 00:00:00 UTC until March 8th 2022 23:59:59 UTC, the fortnight after Russia invaded Ukraine. We queried the hashtags with and without the 'I' for a total of 12 query hashtags, collecting 5,203,746 tweets. The data collected predates the beginning of the 2022 Russian invasion by one day. These hashtags were chosen as they were found to be the most trending hashtags related to the Russia/Ukraine war which could be easily identified with a particular side in the conflict.

We first extracted all of the Twitter-labelled English tweets from the dataset. Of these, we calculated the proportion of words which appear in each LIWC category for a given tweet. These proportions are what we refer to as the 'LIWC Data'. The unique accounts in this filtered data set were randomly sampled to calculate account-level Botometer labels, since Botometer uses language dependent features.

[1] Dataset available at https://figshare.com/articles/dataset/Tweet_IDs_Botometer_results/20486910.

Twitter's takedown of Russian accounts on the March 3rd may lead to bias issues within our data, as the activity of these accounts will not be present in our dataset. However, analysis showed that the content spread by these accounts persisted depsite the takedown[2].

3.1 Categorising accounts via national lean

The query hashtags from each tweet were extracted and the total number of pro-Ukrainian (ending in Ukraine or Zelenskyy) and pro-Russian (ending in Russia or Putin) hashtags were counted and used to establish the national *lean* of a tweet. If the number of pro-Ukranian query hashtags exceeded that of the pro-Russian hashtags, the tweet was labelled as 'ProUkraine', and labeled as 'ProRussia' conversely. If the counts were balanced, the tweet was labelled 'Balanced'. Where applicable, the lean of an account was taken to be the most commonly occurring national lean across all tweets from that account.

We found that 90.16% of accounts fell into the 'ProUkraine' category, while only 6.80% fell into the 'ProRussia' category. The balanced category contained 3.04% of accounts, showing that accounts exhibiting mixed behaviour are present in the dataset.

We explored other methods for categorising accounts, e.g., labelling accounts as 'ProUkraine' or 'ProRussia' if they use only those types of hashtag. However, as we were primarily concerned with aggregated activity, we elected to prioritise labelling each account by their 'usual' behaviour.

3.2 Bot Classifications

We use Botometer [41] to quantify the extent of bot activity in the dataset by assigning scores to a random sample of accounts. Note that we used Botometer's 'English' scores throughout this paper – these scores utilise both language dependent and language independent features during classification [41]. Botometer provides an 'overall' bot score, referred to as the complete automation probability (CAP) and scores corresponding to six distinct sub-types: AstroTurf, Fake Follower, Financial, Self Declared, Spammer and Other.

The rate limits allowed us to randomly sample 26.5% of unique accounts in our dataset which posted at least one English Tweet. This random sample leads to an approximately uniform frequency of Tweets from accounts with Botometer labels across the time frame we considered.

Due to rate limit constraints, the Botometer scores were calculated post-collection, so a small number of accounts may have been removed or scores may be calculated using activity after our collection period.

While it is more appropriate to use Botometer's CAP score as a measure of how bot-like an account is, rather than as a classification tool, it was necessary to label accounts to establish and understand information flows between account groups. Using the recommended cutoff of 0.43, we categorised each labelled account into one of the six Botometer categories or as 'NotBot' [33]. Where an account was not queried, it was labelled as 'FailedToClassify'.

The process for each account is as follows:

1. If the maximum Botometer score is greater than 0.43 then the corresponding category label is assigned to that account.
2. Else if the maximum score is smaller than 0.43, the account is categorised as'NotBot'.
3. Otherwise the account is labelled as 'FailedToClassify'.

The results of classification were 1,347,082 'FailedToClassify', 218,382 'NotBot', 192,633 'Other', 29,627 'Fake Follower', 29,622 'AstroTurf', 1,976 'Spammer', 1,723 'Self Declared' and 662 'Financial' accounts.

4 The role of bots in the overall discussion

Figure 1 shows the average hourly bot probability for different bot types (top), and the hourly frequency of query hashtags (bottom). There is an initial spike in the `#(I)StandWithUkraine` tweets, which is also most dominant overall. Interestingly, the `#(I)StandWithPutin` and `#(I)SupportPutin` hashtags spike on 2nd-3rd March, just after Russia captured its first Ukranian city (Kherson). We believe these spikes in support

[2] https://twitter.com/timothyjgraham/status/1500101414072520704

of Putin are predominately due to the presence of bots, as indicated by the increase in overall bot activity around this time. This observation was independently made by researcher Timothy Graham around this time [32]. On March 4th, Twitter removed over 100 users who pushed the `#(I)StandWithPutin` campaign for violating its platform manipulation and spam policy [8]. This may lead us to underestimate the impact of pro-Russian media after this date, as information may be spreading from alternative sources or shifting to different hashtags.

In Figure 1 we can see the daily cycles in activity. Figure 2 enhances that view by showing the daily cycle based on the hour of day (centred around the mean). Note that the 'AstroTurf' cycle is opposite to that of all other types. Astroturfing accounts are active at opposite times to the other bot types. There are two potential explanations: either the Astroturfing accounts are from a different timezone to a majority of the accounts, or, Botometer uses timezone to determine whether an account is Astroturfing.

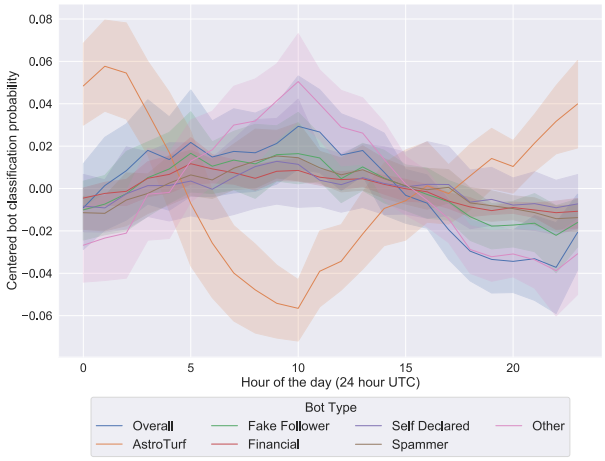


Fig. 2: Average hourly Botometer results showing the daily cycle. The time series observed in Figure 1 (top) is averaged based on the hour of the day (UTC time).

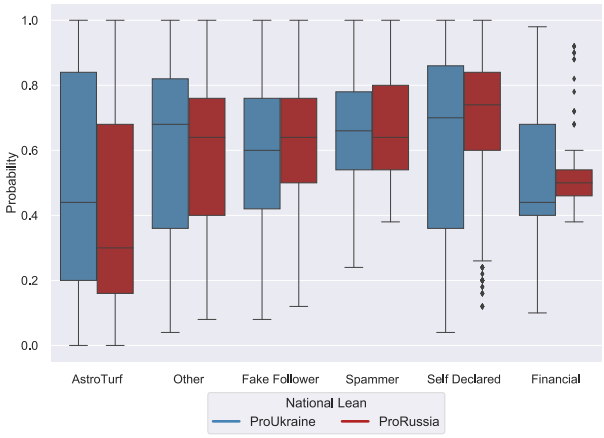


Fig. 3: Probabilities of bot types based on national lean and bot classification. National lean and bot classification are described in Sections 3.1 and 3.2.

Figure 1 (top), also shows a spike in bots on March 2nd and 4th. The first spike aligns with Russia capturing Kherson, but also when the `#(I)StandWithPutin` and `#(I)StandWithRussia` hashtags were trending. We observed the mean overall Botometer score of active pro-Russia accounts increased from 0.535 (1st March) to 0.593 (2nd March), whereas the mean overall Botometer score of active pro-Ukrainian accounts decreased from 0.585 (1st March) to 0.562 (2nd March). Hence, this further suggests that bots were responsible for making the pro-Russia hashtags trend on these dates. The second spike in bot activity on March 4th is more difficult to explain. On this date Russia captured the Zaporizhzhia nuclear power plant, but also a handful of pro-Russian accounts were removed by Twitter. The mean overall Botometer score of active pro-Russia accounts significantly increased from 0.535 (3rd March) to 0.613 (4th March) and the mean overall Botometer score of active pro-Ukrainian accounts slightly increased from 0.573 (3rd March) to 0.603 (4th March). As a result, this spike in bot activity is due to the presence of pro-Russian bots (before they were removed) and the presence of pro-Ukrainian bots advocating against the pro-Russian accounts. Nonetheless, there is an obvious presence of bots over the duration of the first fortnight after Russia's invasion of Ukraine.

The time of day effects are most pronounced for the AstroTurf and Other bots, whereas the activity of Fake Follower, Financial, Self Declared and Spammer bots are less impacted by the time of day. This may be because AstroTurf and Other bots are pushing campaigns specific to certain countries, and hence sharing content aligned with those timezones. The spike in Other bots occurs at 10:00 UTC which corresponds to 1:00pm Ukrainian time. Matthews [21] suggested that noon to 1:00pm is the most popular time to tweet in any timezone. Hence, the Other bots are likely to be increasing their engagement in Ukraine by being most active around this time.

Figure 3 shows pairwise box plots of the Botometer type probabilities based on whether the accounts are pro-Ukraine or pro-Russia. The most commonly-used bot type for both campaigns is the Self-Declared bots, suggesting that authorities have identified these bots to be most useful in a information warfare campaign. Furthermore, we observe a fairly consistent spread of bot types for both campaigns. Pro-Russian accounts have a mean CAP score of 0.42, while pro-Ukrainian accounts have a mean score of

0.43, with medians 0.36 and 0.34 respectively. However, the median probability of an account being an AstroTurf bot is slightly higher for pro-Ukrainian accounts than pro-Russian accounts. Additionally, the median probability of a Self-Declared bot is slightly higher for pro-Russian accounts compared to pro-Ukrainian accounts. This highlights that pro-Ukrainian accounts may be utilising more Astroturfing in their information warfare, whereas pro-Russian accounts may be utilising more Self-Declared bots.

5 Information flows between bots and human accounts

5.1 Information-flow estimation methods

We measure the influence of accounts on overall online discussion using the following symmetric net information flow measure from the time-stamped writings of a source $\mathcal{S}$ to target $\mathcal{T}$ [36]:

$$\Delta(\mathcal{T}||\mathcal{S}) = \frac{\hat{h}(\mathcal{T}||\mathcal{S})}{\sum_X \hat{h}(\mathcal{T}||X)} - \frac{\hat{h}(\mathcal{S}||\mathcal{T})}{\sum_X \hat{h}(\mathcal{S}||X)}. \tag{1}$$

Here $\hat{h}(\mathcal{T}||\mathcal{S})$ is the non-parametric cross entropy rate estimator [1,17]:

$$\hat{h}(\mathcal{T}||\mathcal{S}) = \frac{N_{\mathcal{T}} \log_2 N_{\mathcal{S}}}{\sum_{i=1}^{N_{\mathcal{T}}} \Lambda_i(\mathcal{T}|\mathcal{S}_{\leq t(T_i)})}, \tag{2}$$

where N_S and N_T are the number of symbols written by the source and target, respectively, and Λ_i^i denotes the length of the shortest substring, l starting at index i which does not appear in the first $i + l - 1$ symbols. See [2] for an example of Λ_i^i estimation. We aggregate content by account type rather than on an account level to measure the information flows between account types and establish their significance.

We use the language analysis tools Valence Aware Dictionary and Sentiment Reasoner (VADER) [13] for sentiment analysis, as well as the Linguistic Inquiry and Word Count (LIWC) [29] to establish relationships between conversation features and bot-activity. We then use the Granger causality test to determine whether one time series X is useful in forecasting another time series Y with some time lag p.

We do this by fitting two linear models. The first model we include only the lagged values of Y:

$$Y_t = \alpha_{1,0} + \alpha_{1,1} Y_{t-1} + \cdots + \alpha_{1,p} Y_{t-p} + \epsilon_{1,t}, \tag{3}$$

where we define $\epsilon_{i,t}$ as the error term of model i at time t and $\alpha_{i,j}$ as the parameter of model i at lag j. Next, we augment the model to also include the lagged values of X:

$$Y_t = \alpha_{2,0} + \alpha_{2,1} Y_{t-1} + \cdots + \alpha_{2,p} Y_{t-p} + \beta_1 X_{t-1} + \cdots + \beta_p X_{t-p} + \epsilon_{2,t}. \tag{4}$$

The null hypothesis, that X does not Granger-cause Y, is accepted via an F-test if and only if no lagged values of X are retained in the regression model observed in Eq. 4.

5.2 Aggregated information flows

We apply information-flow measures to content aggregated by account type, to understand inter-community information flows. Rather than using an aggregate statistic on individual information flows, the proposed aggregated flow approach allows the symmetric and normalisation properties of the net information flow measure [36] to be preserved. This process improves the quality of the entropy estimate for the group behaviour, by increasing the available sequence length and mitigating the effect of slow convergence of the estimator. In this section we also develop a significance test for net information flow.

Each account is labeled by both bot classification and national lean, and the content within these account groups is aggregated. The cross entropy between each of these groups is calculated pairwise, and these values are normalised according to Eq. 1.

These pairwise cross entropy estimations produce a fully connected network. We then perform a statistical test for whether aggregated net information flows are significant between groups, allowing a network of significant information flows to be constructed (Figure 4).

To approximate the null distribution for the difference between median outgoing flow rates between each group, we randomly shuffle group labels for each tweet, reconstruct aggregated sequences, then calculate

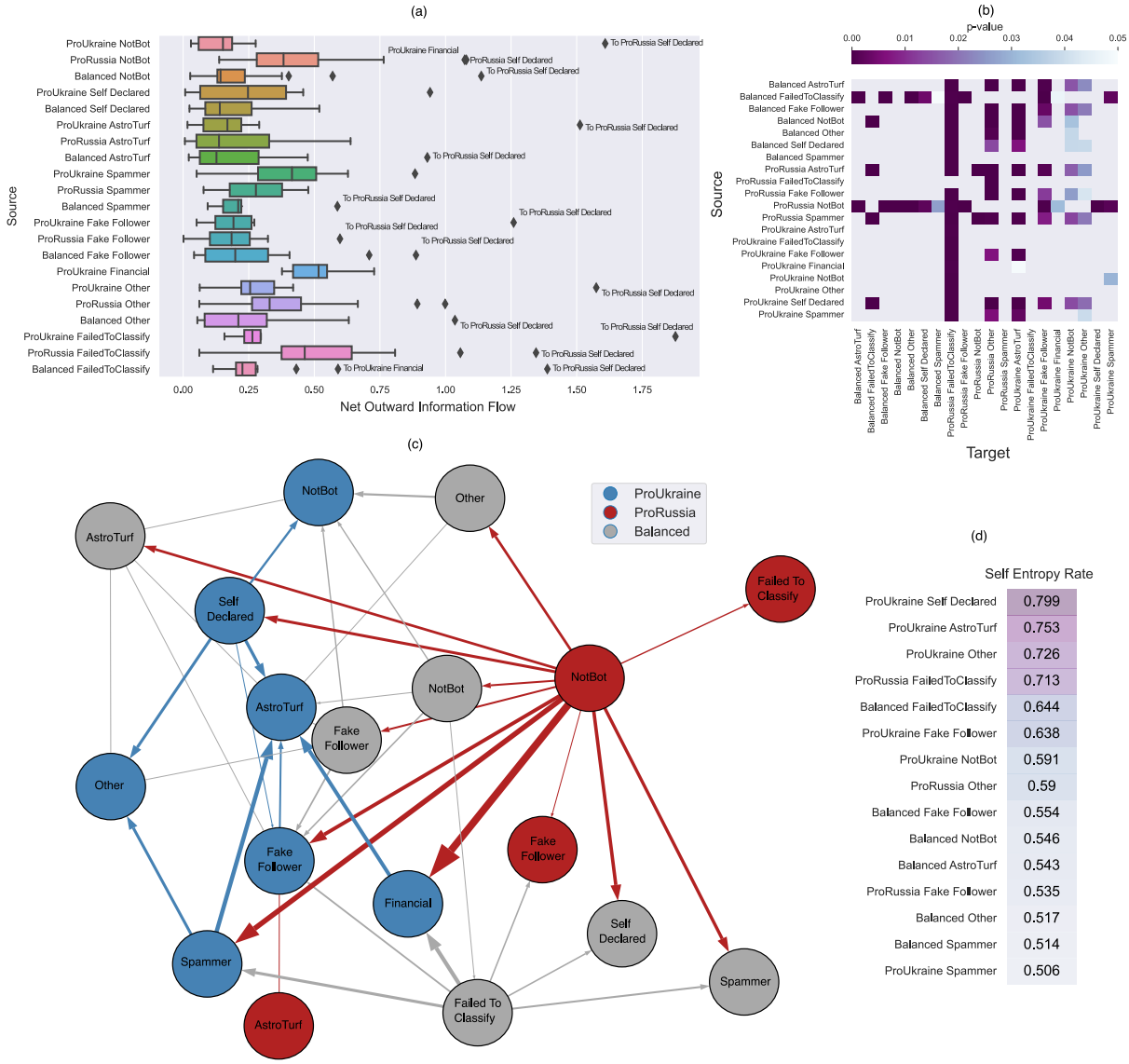


Fig. 4: (a) Aggregated net outward information flows by account type and national lean. The outward flows for each aggregated group predominantly fall in the $(0, 0.5)$ range, however 'ProRussia NonBot', 'ProUkraine Financial', 'ProUkraine Spammer' and 'ProRussia FailedToClassify' net flows tend to be greater. Values above the 80th percentile are labelled – the majority of these represent flows into the 'ProRussia Self Declared' group. (b) Heatmap of empirical p-values for each intergroup net information flow, showing significance of the difference in median outbound flows between groups. This test is used to form the network shown in (c). (c) Significant net information flows between groups. Not pictured are aggregated groups with inadequate sample size. Each aggregated group is coloured by national lean; edges are weighted by magnitude of net information flow. Significant flows out of 'ProRussia NotBot' accounts indicate that information flows from these accounts to other groups, most commonly groups with a balanced or 'ProUkraine' lean. Most intergroup flows for groups with a 'ProUkraine' lean are to other groups with 'ProUkraine' lean, with no significant flows from 'ProUkraine' account groups to 'ProRussia' account groups. (d) Lists the 15 highest self entropy rate values for the aggregated groups, with the top three groups all having a 'ProUkraine' lean.

net information flows. These aggregated net flows are used to calculate the differences in group median out-flows and construct the empirical null distribution, which is used to calculate an empirical p-value for the observed values. These aggregated net information flows reveal that generally information flows out of the pro-Russian accounts, with the exception of the pro-Russia FailedToClassify and pro-Russian Fake Follower account groups (Figure 4). This indicates that these account groups may be predominately interacting with other accounts within the same group, rather than accounts with other leans or types. The 'ProRussia NotBot' account group has the largest outward information flows and significant flows to a range of other groups, having a positive information flow into both 'ProUkraine' and 'Balanced' account groups.

This indicates that these Russian non-bot accounts influence a variety of user groups with the greatest between group information flows. This may indicate that human-controlled accounts, or accounts which appear less bot-like, have more influence in our social network, potentially due to their behaviour or perception. While the 'NotBot' label is derived from the Botometer score, this label does not mean these accounts are not malicious or automated.

Most of the significant information flows between 'ProUkraine' account groups is between groups with the same lean. This may indicating that more information flows between the accounts within each of these groups rather than to accounts in other groups. 'ProUkraine' groups have the highest self entropy rates, meaning that these groups do not just aggregate information from other account groups, but influence other accounts within the same group (Figure 4 (d)).

The Balanced account groups show information flows to all other national-lean types, and connect otherwise disjoint parts of the information flow network. These accounts may act as a bridge for information to move between 'ProRussia' and 'ProUkraine' accounts. Most of these groups have small but significant information flows to other groups, with information tending to flow out of these groups.

Notably, the few significant information flows into non-bot account groups indicate some influence from 'bot-like' accounts on non-bot accounts. However, these account groups have stronger outward net information flows then inward flows, suggesting that while they tend to have influence on the content of other 'bot-like' accounts, they do not influence non-'bot-like' users generally.

When account-level flows are considered rather than the aggregated flows presented here, several similar significant flows exist between 'bot-like' and non-'bot-like' accounts.

6 How bot accounts influence linguistic features of the conversation

Having characterised bot activity and identified significant information flows, we now aim to explore the content of these relationships. We first consider relationships between bot activity and sentiment, with a focus on understanding if bot-like accounts have a significant impact on the compound sentiment of non-bot accounts, measured using the CAP Botometer score and weighted average compound sentiment. The linguistic impact is then quantified by using LIWC to develop a statistical framework for understanding the relationship between bots activity and emotional/linguistic content.

6.1 Bot activity and overall sentiment

To understand how bots drive non-bot sentiment, we begin by cleaning and preparing two time series. The first is the mean CAP Botometer score, which acts as a proxy for the total proportion of bot-like activity on the network. The second is the CAP-weighted mean compound sentiment. Weighting the VADER compound sentiment by the complement of the Botometer CAP score provides a measure of non-bot sentiment without making account labelling assumptions. It is robust to threshold choices, and provides a meaningful measure of the overall sentiment of the dataset.

Each time series is aggregated hourly. The first 50 hours are removed from both time series since there is a comparatively small tweet volume over that period.The mean CAP Botometer score has a linear trend, which is removed via a linear regression. Both time series are standardised to have mean zero to ensure they comply with the assumptions to perform Granger causality analysis. We also removed the daily periodic cycle (Figure 2) from each time series.

The cross correlations are then calculated for various lags to understand the effect of mean CAP Botometer score on the CAP-weighted mean compound sentiment. A maximum lag of 12 hours is considered. A positive relationship exists between the cleaned time series, indicating there is a correlation between the activity of 'bot-like' accounts and the compound sentiment of the non-'bot-like' accounts. There is a significant relationship between the two series, with $p = 3.76 \times 10^{-4}$.

Since effects cannot occur simultaneously, we consider the lagged effect of bot activity on non-bot compound sentiment, finding a positive cross correlation for both positive and negative lags (Figure 5). This shows that bot activity increases when sentiment increases, but also that sentiment increases with increases in bot activity. Figure 5 indicates that the relationship between sentiment and bot activity is complicated, with marked events driving spikes in the compound sentiment of not-'bot-like' accounts. There are also spikes in the mean compound sentiment of other account lean types, which may be due to events which we did not consider in our analysis. This indicates that there may not be an overall

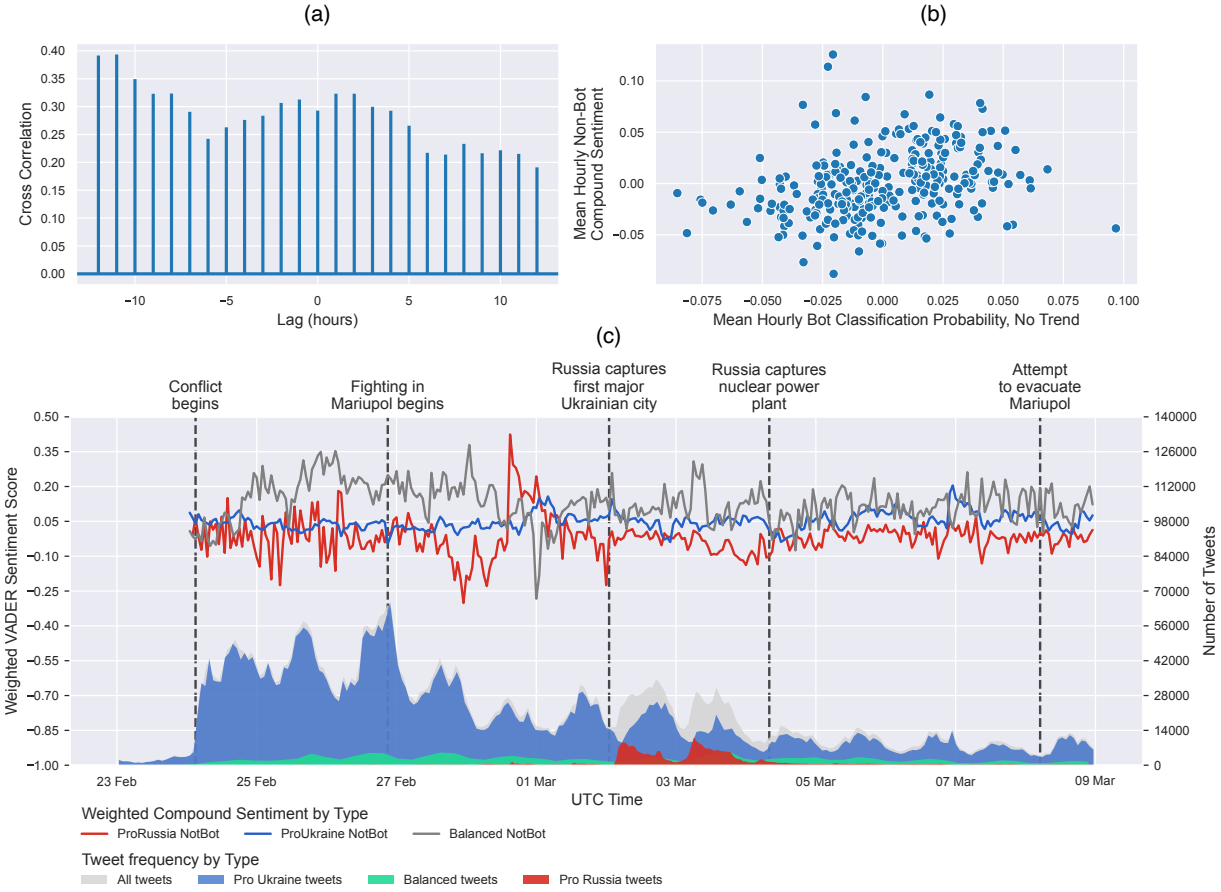


Fig. 5: (a) Shows the lagged cross correlations between the mean hourly Botometer Overall CAP score and the CAP weighted mean compound sentiment. Each lag represents an offset of the CAP weighted mean compound sentiment in hours. These correlations are significant in both lag directions. (b) Considering a scatterplot of the two timeseries with trend and burn-in samples removed, reveals a positive linear relationship between them. (c) The timeseries represent the CAP weighted mean compound sentiment grouped by national lean, with 'ProRussia', 'ProUkraine' and 'Balanced' accounts considered separately. To aid interpretation of these timeseries, the tweet frequency of each type (from all accounts) and some significant event markers are given. Before the 2nd of March there was minimal activity from 'ProRussia' accounts, so the CAP weighted mean compound sentiment estimate has high variance and is of low quality. After March 3rd, there is a spike in Balanced CAP weighted mean compound sentiment, from Balanced accounts, suggesting that these accounts were producing more positive tweets overall, potentially in response to humanitarian corridors opening.

effect on the non-bot compound sentiment due to bot activity, although this relationship may exist on an individual account level.

6.2 Bot activity and linguistic discussion features

Using LIWC, we explore how different types of bots drive emotions and discussions around the Russia/Ukraine conflict. We produce hourly averages for overall LIWC proportions and the Botometer probabilities. This results in a set of time series, all over 336 hours. We utilise the Granger Causality Test (Section 5.1) on these time series to determine whether the activity of certain bots Granger-cause more or less discussion around particular LIWC categories.

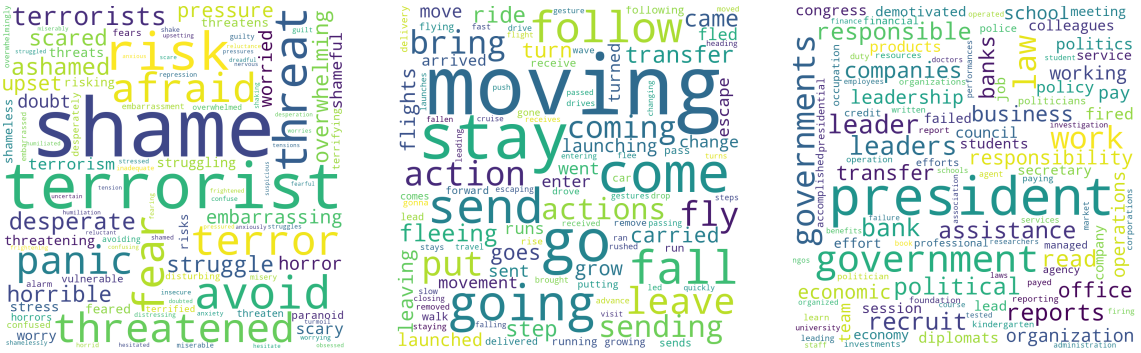
We apply pairwise Granger Causality Tests between each Botometer timeseries, X, and each LIWC timeseries, Y, for $p = 12$ lags/hours (see Eq. 3 and 4 in Section 5.1). This time window is chosen as it is reasonable to assume a majority of the effects from bots will occur over this time frame. The validity of this assumption is explored below.

We use the F-score from the Granger Causality Test as a measure of how 'influential' a type of bot is on each LIWC category. To get a sense of direction for these relationships, we use the sign of the largest β coefficient from Eq. 4 in Section 5.1. We multiply the sign of this coefficient by the F-score from the Granger Causality Test to obtain a measure of strength and direction, referring to this as the Bot Effect Strength and Direction. Moreover, we use the lag of the largest β coefficient from Eq. 4 in Section 5.1

to represent the most prolific lag in the relationship. We use the p-value from the Granger Causality Test to determine whether the effects are significant, and perform a Bonferroni Adjustment to adjust for multiple hypothesis tests. The results are displayed in Figure 6, where we have only included the significant relationships. The number in the centre of each square represents the most prolific lag – we interpret this as the number of hours until the effects of the bot activity are most pronounced. Figure 6 shows that bots

Fig. 6: A series of pairwise Granger Causality Tests are performed to examine whether the activity of bot types is Granger-causing changes in discussions of the LIWC categories. The heat maps colour describes the bot effect strength and direction from the Granger Causality Test (over 12 hours/lags) between the time series of hourly bot proportions and the time series of hourly LIWC category proportions. The number in the centre describes the most prolific lag in the Granger Causality Test. We calculate the bot effect strength using the F-score from an F-test on the Granger Causality linear models. Moreover, we calculate the bot effect direction and most prolific lag using the sign and lag (respectively) of the largest β coefficient from Eq. 4 in Section 5.1. We perform a Bonferroni Adjustment on the p-values from the Granger Causality Tests and only show the Bot Types and LIWC Categories with a significant adjusted p-value (< 0.05).

do have a significant impact on discussions of certain LIWC categories. To better understand what each of these LIWC categories represent, we generated word clouds of the words from each LIWC category that appeared in the dataset. The size of the words represent their relative frequency in the data – the larger the word the more frequently it occurs. The word clouds for Angst, Motion and Work are shown in Figures 7a, 7b and 7c, respectively. For a full discussion of the words associated with various LIWC categories, see Pennebaker and Francis [28]. In Figure 6, the self-declared bots have greatest amount of influence on



(a) Angst Category (b) Motion Category (c) Work Category

Fig. 7: Word Clouds which demonstrate the frequency of words in particular LIWC categories.

a number of discussions. In particular, the self-declared bots increase discussions around angst, friends, motion, time, work and the usage of filler words, but decrease the usage of function words. Moreover, it is apparent that these bots most strongly influence discussion of the work category (with a most prolific lag of five hours). Figure 7c shows that most of the discussion around work is involved with governing

bodies, with 'president' and 'governments' being the most commonly used words. While it is difficult to assert exactly why these bots are driving more discussions of work, we gain further understanding by also observing that self declared bots Granger-cause more angst (with a most prolific lag after 7 hours). Combining these two observations suggests that self-declared bots drive more angst about governing bodies. From a pro-Russian perspective, this may be to cause more disruption in the West, and from a pro-Ukrainian perspective, this may be to cause more disruption in Russia. Figure 3 shows a fairly even probability of pro-Russia and pro-Ukraine accounts being self-declared bots. Although the exact origin of self-declared accounts is unknown, it is worth noting that we considered predominately English accounts. It is therefore more likely that the intention of these accounts was to drive more disruption in English-speaking countries.

Observe that Fake Follower, Spammer and Other bots also increase angst (all with the most prolific lag after 7 hours). Figure 7a shows that a majority of angst-related words are surrounding fear and worry. Hence, we argue that self-declared, fake follower, spammer and other forms of automated account types combine to increase fear in the overall discussion of the Russia/Ukraine war. This observation has been hypothesised by many authors [24,26], but a detailed analysis has been lacking and may be of concern for many governments and defence organisations.

Figure 6 further shows that fake follower, self-declared, spammer and other bot types also increase online discussion around motion. In Figure 7b, we see a number of motion related words that are potentially associated with staying or fleeing the country. Combining this with increases in Angst suggests that bots could be influencing people's decisions surrounding whether to flee their homes or not. Druziuk [11] noted that bots have allowed "Ukrainians to report to the government when they spot Russian troops", but the usage of bots to influence people on staying/leaving the country is something not observed before. However, it is difficult to denote whether this is being done in support of Ukraine, Russia or both.

In Figure 6 the most prolific lag is mostly consistent for a given LIWC category, but varies greatly for bot type. Hence, the time which bots effect a given discussion on the war depends mainly on the topic of discussion and not on the type of bot. For instance, we observe that Fake Follower, Self-Declared, Spammer and Other bots all most prolifically effect discussions of work after five hours. To further examine the effects of the lag on discussion of different LIWC categories, we plot cross correlations in Figure 8. These plots represent the cross correlation between Self-Declared bot proportions and a number of significant LIWC categories (in Figure 6) over 48 hours.

The direction of the effect for each LIWC variable in Figure 8 is consistent with Figure 6, further validating our results. This direction is consistent for all significant lags, justifying our decision to choose the largest

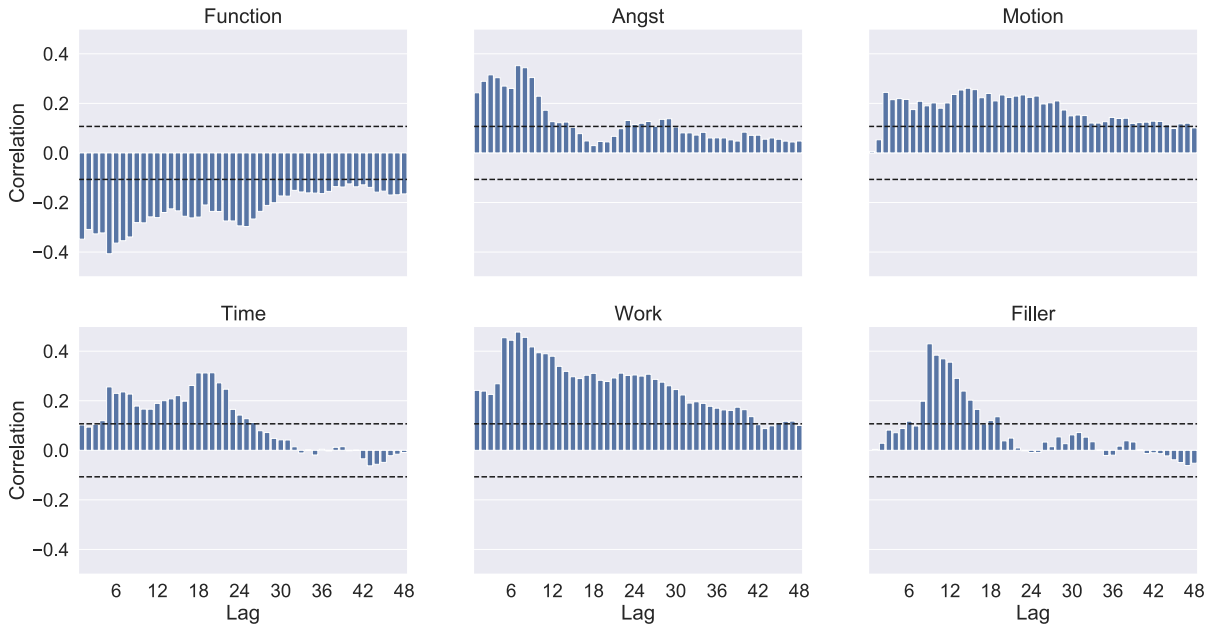


Fig. 8: Lagged cross correlations between the hourly Self Declared bot proportions and the significant hourly LIWC Category proportions (significance is determined from the results in Figure 6). We consider 48 hours/lags for each of these plots and represent the significance threshold using a horizontal dotted line. This shows the extent to which the bots drive changes in online discussion and how long these effects can persist for.

parameter in the regression model as an indication of direction in Figure 6. For some LIWC categories the effects of Self-Declared bots linger over many lags but for others the effects diminish relatively quickly. For instance, the effects on the work category and the function category are significant for lags almost up to 48 hours, whereas the effects on the angst and filler categories diminish within 24 hours.

7 Conclusion

This work investigates if and how bot-like accounts influence the online conversation around the Russian invasion of Ukraine during February 2022. We showed which account groups have measurable influence by aggregating accounts using their national lean and account bot-type label. The patterns of information flows between bot and non-bot account vary based on national lean: Pro-Russian non-bot accounts are most influential overall, with information flows to a variety of other account groups. No significant outward flows exist from pro-Ukrainian non-bot accounts, with significant flows from pro-Ukrainian bot accounts into pro-Ukrainian non-bot accounts. Pro-Russian account groups are seemingly isolated, with smaller self-entropy rates and less significant between group net information flows. However, there exists significant information flows out of pro-Russia non-bot and AstroTurf account groups, with the largest net flows originating in the pro-Russian non-bot account groups. Contrastingly, pro-Ukrainian account groups tend to have more information flows between pro-Ukrainian and balanced account groups. Pro-Ukrainian aggregated groups also tend to have higher self-entropy rates.

To understand how bot-like accounts influence non-bot like accounts across all national lean types, we consider non-bot sentiment, measured using a weighted compound sentiment score. By weighting this by the overall bot probability, this compound sentiment reflects the overall sentiment of non-bot-like accounts across the network. The relationship between this sentiment and the bot activity is significant but occurs in both directions, with sentiment and bot activity both impacting each other.

Finally, we identify the effect of bot-like accounts on LIWC linguistic discussion features. Self-declared bots have the largest impact, showing significant relationships with word in the Function, Angst, Friend, Motion, Time, Work and Filler categories. To find the direction and significance of these relationships we use pairwise Granger causality tests. We find bots generally increase word usage in these categories with a 3-10 hour lag. Self-declared bots show the strongest relationship with Work discussions, i.e., words pertaining to governing bodies ("president", "government" and "leadership"). We also see bot accounts increase the use of words in the angst category which contains words related to fear and worry ("shame", "terrorist", "threat", "panic").

In future work, we will explore information contained on the network of interactions between users recorded in the tweets using a network science approach [3,23,4]. We will also further explore diverse ways to classify the national lean of authors based on their published Twitter content. Preliminary results indicate heavy-tailed distributions in timing lags that differ between account types, suggesting differences in coordinated activity signatures [20]. We will examine coordination campaigns and coordination networks [34,12,27,14,19] to quantify the impact of coordinated activity in the social network structure and to further investigate its influence on social media users.

Using a number of approaches we describe a framework to understand the impact of bots on the network, explore how malicious campaigns operate and measure their effect on online discussion. Our approach is applicable to any social media content presenting polarisation between distinct groups and can be applied to other data to understand how malicious campaigns operate.

8 Acknowledgements

B.S. would like to acknowledge the support of a Westpac Future Leaders Scholarship. L.M. and M.R. are supported by the Australian Government through the Australian Research Council's Discovery Projects funding scheme (project DP210103700). L.M. also acknowledges support from the Australian Defence Science and Technology Group ORNet scheme.

References

1. Bagrow, J.P., Liu, X., Mitchell, L.: Information flow reveals prediction limits in online social activity. Nature Human Behaviour **3**(2), 122–128 (Feb 2019). https://doi.org/10.1038/s41562-018-0510-5
2. Bagrow, J.P., Mitchell, L.: The quoter model: A paradigmatic model of the social flow of written information. Chaos: An Interdisciplinary Journal of Nonlinear Science **28**(7), 075304 (2018)
3. Barabási, A.L.: Network science. Cambridge university press (2016)
4. Caldarelli, G.: Scale-free networks: complex webs in nature and technology. Oxford University Press (2007)
5. Champion, M., Krasnolutska, D.: Ukraine's TV comedian President Volodymyr Zelenskyy finds his role as wartime leader (Feb 2022), https://www.japantimes.co.jp/news/2022/02/26/world/volodymyr-zelenskyy-wartime-president/
6. Chen, E., Ferrara, E.: Tweets in time of conflict: A public dataset tracking the twitter discourse on the war between Ukraine and Russia. arXiv preprint arXiv:2203.07488 (2022)
7. Collins, B.: After Mueller report, Twitter bots pushed 'Russiagate hoax' narrative (2019), https://www.nbcnews.com/tech/tech-news/after-mueller-report-twitter-bots-pushed-russiagate-hoax-narrative-n997441, Accessed 19/6/2022
8. Collins, B., Korecki, N.: Twitter bans over 100 accounts that pushed #IStandWithPutin. https://www.nbcnews.com/tech/internet/twitter-bans-100-accounts-pushed-istandwithputin-rcna18655 (Mar 2022)
9. Cresci, S., Di Pietro, R., Petrocchi, M., Spognardi, A., Tesconi, M.: The Paradigm-Shift of Social Spambots: Evidence, Theories, and Tools for the Arms Race. In: Proceedings of the 26th International Conference on World Wide Web Companion. pp. 963–972. WWW '17 Companion, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE (Apr 2017). https://doi.org/10.1145/3041021.3055135
10. Doroshenko, L., Lukito, J.: Trollfare: Russia's disinformation campaign during military conflict in Ukraine. International Journal of Communication **15**, 4662–4689 (2021)
11. Druziuk, Y.: A citizen-like chatbot allows Ukrainians to report to the government when they spot Russian troops - here's how it works. Business Insider (Apr 2022), https://www.businessinsider.com/ukraine-military-e-enemy-telegram-app-2022-4, Accessed on 19/6/2022
12. Giglietto, F., Righetti, N., Rossi, L., Marino, G.: It takes a village to manipulate the media: coordinated link sharing behavior during 2018 and 2019 italian elections. Information, Communication & Society **23**(6), 867–891 (2020). https://doi.org/10.1080/1369118X.2020.1739732
13. Hutto, C., Gilbert, E.: Vader: A parsimonious rule-based model for sentiment analysis of social media text. In: Proceedings of the international AAAI conference on web and social media. vol. 8, pp. 216–225 (2014)
14. Keller, F.B., Schoch, D., Stier, S., Yang, J.: Political astroturfing on twitter: How to coordinate a disinformation campaign. Political Communication **37**(2), 256–280 (2020)
15. Keller, T.R., Klinger, U.: Social Bots in Election Campaigns: Theoretical, Empirical, and Methodological Implications. Political Communication **36**(1), 171–189 (Jan 2019). https://doi.org/10.1080/10584609.2018.1526238
16. Klepper, D.: Russian propaganda 'outgunned' by social media rebuttals. AP NEWS (Mar 2022), https://tinyurl.com/3x9anuta, section: Russia-Ukraine war
17. Kontoyiannis, I., Algoet, P.H., Suhov, Y.M., Wyner, A.J.: Nonparametric entropy estimation for stationary processes and random fields, with applications to english text. IEEE Transactions on Information Theory **44**(3), 1319–1327 (1998)
18. Laurence, P.: How Ukraine's 'Ghost of Kyiv' legendary pilot was born. BBC News (May 2022), https://www.bbc.com/news/world-europe-61285833, Accessed 19/6/2022
19. Lukito, J.: Coordinating a multi-platform disinformation campaign: Internet research agency activity on three us social media platforms, 2015 to 2017. Political Communication **37**(2), 238–255 (2020). https://doi.org/10.1080/10584609.2019.1661889
20. Mathews, P., Mitchell, L., Nguyen, G., Bean, N.: The nature and origin of heavy tails in retweet activity. In: Proceedings of the 26th International Conference on World Wide Web Companion. pp. 1493–1498 (2017)
21. Matthews, B.: Best time to tweet for clicks, retweets and engagement. Empower Agency (Jun 2015), https://empower.agency/best-time-to-tweet-clicks-retweets-engagement/, Accessed 19/6/2022
22. Muscat, S., Siebert, Z.: Laptop generals and bot armies: The digital front of Russia's Ukraine war (Mar 2022), https://eu.boell.org/en/2022/03/01/laptop-generals-and-bot-armies-digital-front-russias-ukraine-war, Accessed 19/6/2022
23. Newman, M.: Networks. Oxford university press (2018)
24. Nguyen, K.: How Putin's propaganda is sowing seeds of doubt to deny sympathy for Ukraine. ABC News (Apr 2022)
25. Orabi, M., Mouheb, D., Al Aghbari, Z., Kamel, I.: Detection of Bots in Social Media: A Systematic Review. Information Processing & Management **57**(4), 102250 (Jul 2020). https://doi.org/10.1016/j.ipm.2020.102250
26. Osborne, C.: Ukraine destroys five bot farms that were spreading 'panic' among citizens. https://www.zdnet.com/article/ukraine-takes-out-five-bot-farms-spreading-panic-among-citizens/ (Mar 2022)
27. Pacheco, D., Hui, P.M., Torres-Lugo, C., Truong, B.T., Flammini, A., Menczer, F.: Uncovering coordinated networks on social media: methods and case studies. In: Proceedings of the AAAI international conference on web and social media (ICWSM). pp. 455–466 (2021)

28. Pennebaker, J.W., Francis, M.E.: Cognitive, Emotional, and Language Processes in Disclosure. Cognition and Emotion **10**(6), 601–626 (Nov 1996). https://doi.org/10.1080/026999396380079
29. Pennebaker, J.W., Francis, M.E., Booth, R.J.: Linguistic inquiry and word count: Liwc 2001. Mahway: Lawrence Erlbaum Associates **71**(2001), 2001 (2001)
30. Polyzos, E.S.: Escalating tension and the war in Ukraine: Evidence using impulse response functions on economic indicators and twitter sentiment. Available at SSRN 4058364 (2022)
31. Pond, T., Magsarjav, S., South, T., Mitchell, L., Bagrow, J.P.: Complex contagion features without social reinforcement in a model of social information flow. Entropy **22**(3), 265 (Mar 2020). https://doi.org/10.3390/e22030265
32. Purtill, J.: When it comes to spreading disinformation online, Russia has a massive bot army on its side. ABC News (Mar 2022)
33. Sayyadiharikandeh, M., Varol, O., Yang, K.C., Flammini, A., Menczer, F.: Detection of Novel Social Bots by Ensembles of Specialized Classifiers. In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management. pp. 2725–2732 (Oct 2020). https://doi.org/10.1145/3340531.3412698
34. Schoch, D., Keller, F.B., Stier, S., Yang, J.: Coordination patterns reveal online political astroturfing across the world. Scientific reports **12**(1), 4572 (2022). https://doi.org/10.1038/s41598-022-08404-9
35. Shane, S.: The fake americans Russia created to influence the election. The New York Times (Sep 2017), https://www.nytimes.com/2017/09/07/us/politics/russia-facebook-twitter-election.html, Accessed 19/6/2022
36. South, T., Smart, B., Roughan, M., Mitchell, L.: Information flow estimation: a study of news on Twitter. Tech. Rep. arXiv:2205.06029, arXiv (May 2022). https://doi.org/10.48550/arXiv.2205.06029, http://arxiv.org/abs/2205.06029, arXiv:2205.06029 [physics] type: article
37. Stella, M., Ferrara, E., Domenico, M.D.: Bots increase exposure to negative and inflammatory content in online social systems. Proceedings of the National Academy of Sciences **115**(49), 12435–12440 (2018). https://doi.org/10.1073/pnas.1803470115, https://www.pnas.org/doi/abs/10.1073/pnas.1803470115
38. Thomas, T.: Russia's reflexive control theory and the military. Journal of Slavic Military Studies **17**(2), 237–256 (2004). https://doi.org/10.1080/13518040490450529
39. Watt, J., Smart, B.: Tweets discussing the Russia/Ukraine War (August 2022). https://doi.org/10.6084/m9.figshare.20486910.v4, https://figshare.com/articles/dataset/Tweet_IDs_Botometer_results/20486910
40. Wong, E.: U.S. fights bioweapons disinformation pushed by Russia and China. The New York Times (Mar 2022), https://www.nytimes.com/2022/03/10/us/politics/russia-ukraine-china-bioweapons.html, Accessed on 19/6/2022
41. Yang, K.C., Ferrara, E., Menczer, F.: Botometer 101: Social bot practicum for computational social scientists (Jan 2022). https://doi.org/10.48550/arXiv.2201.01608

Chapter 5

Publication: The entropy rate of Linear Additive Markov Processes

5.1 Summary

With the sequence lengths encountered in real-world data sets, the non-parametric entropy estimator applied in Chapter 4 does not reach the true entropy estimate due to a slow convergence rate. The rate of convergence is not an issue for relative comparisons between estimates formed using sequences of approximately equal length. Hence, this finding does not invalidate existing work which considers relative relationships between information flows as in Chapter 4.

To investigate and quantify the size of this error, it is useful to have a model with known ground truth. This is a current avenue of future work, and the rate of convergence has been considered for independent and identically distributed sequences. This has led to the development of a theoretical bias correction which will be described in future work. To extend this correction to cases with dependency structures within the sequence, it is necessary to have a validation distribution with a known entropy rate. While the true entropy rate is known for independently and identically distributed variables, these sequences are dissimilar to those which appear in real-world contexts where long-term dependencies may be present. This publication identifies a suitable model able to produce sequences with long-term dependency structures, derives the theoretical entropy rate and validates this process on four publically available datasets.

The Linear Additive Markov Process (LAMP) model Kumar et al. (2017) was identified as an expressive model with comparatively small parameter space which allows the long-term dependency structure to be explicitly defined. This makes it a suitable model for validating and exploring the behaviour of non-parametric entropy estimators. As a new

model, the theoretical entropy rate had not been derived, and the following publication derives a theoretical value for this rate, finding it to be the same as the underlying first-order Markov chain.

This work goes on to explore the applicability of the LAMP model for estimating entropy rates, fitting the model to four publically available data sets. To calculate an entropy estimate, it is necessary for the transition matrix of the underlying Markov process to be ergodic, and this work also explores two methods of forcing ergodicity and their relative effect on the final entropy estimate.

5.2 Statement of Authorship: The entropy rate of Linear Additive Markov Processes

Title of Paper	The entropy rate of Linear Additive Markov Processes
Publication Status	Submitted for publication
Publication Details	Submitted for publication at PLOS One. Preprint available on arVix at `https://arxiv.org/abs/2211.05350`.

Principal Author

Name of Principal Author (Candidate)	Bridget Smart
Contribution to the Paper	Under the supervision of Professor Matthew Roughan and Lewis Mitchell I contributed to the conception of the work, collected the data, processed the data, wrote all code and undertook all data analysis. I derived the theoretical entropy of the LAMP model and wrote the manuscript, with corrections and edits from both co-authors.
Overall percentage (%)	80 %
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.
Signature	Date: 26/01/2023

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

I the candidate's stated contribution to the publication is accurate (as detailed above);

ii permission is granted for the candidate in include the publication in the thesis; and

iii the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Matthew Roughan		
Contribution to the Paper	15 %		
Signature		Date	

Name of Co-Author	Lewis Mitchell		
Contribution to the Paper	5%		
Signature		Date	20/1/2023

The entropy rate of Linear Additive Markov Processes

Bridget Smart, Matthew Roughan and Lewis Mitchell

I. Abstract

This work derives a theoretical value for the entropy of a Linear Additive Markov Process (LAMP), an expressive model able to generate sequences with a given autocorrelation structure. While a first-order Markov Chain model generates new values by conditioning on the current state, the LAMP model takes the transition state from the sequence's history according to some distribution which does not have to be bounded. The LAMP model captures complex relationships and long-range dependencies in data with similar expressibility to a higher-order Markov process. While a higher-order Markov process has a polynomial parameter space, a LAMP model is characterised only by a probability distribution and the transition matrix of an underlying first-order Markov Chain. We prove that the theoretical entropy rate of a LAMP is equivalent to the theoretical entropy rate of the underlying first-order Markov Chain. This surprising result is explained by the randomness introduced by the random process which selects the LAMP transitioning state, and provides a tool to model complex dependencies in data while retaining useful theoretical results. We use the LAMP model to estimate the entropy rate of the LastFM, BrightKite, Wikispeedia and Reuters-21578 datasets. We compare estimates calculated using frequency probability estimates, a first-order Markov model and the LAMP model, and consider two approaches to ensuring the transition matrix is irreducible. In most cases the LAMP entropy rates are lower than those of the alternatives, suggesting that LAMP model is better at accommodating structural dependencies in the processes.

***Index Terms*—LAMP processes, entropy, Markov processes, long-range dependency, stochastic processes.**

II. Introduction

Markov processes are a simple model with wide-ranging applications. They rely on the Markov property, where the next state is only dependent on the current state, and is conditionally independent of the history of the process. The model's simplicity allows theoretical results to be easily derived and understood, and the flexibility of the model's framework makes it a useful tool which can be easily extended, and has consequently been used in a vast number of applications.

B. Smart, M. Roughan and L.Mitchell are with the University of Adelaide. Email {bridget.smart, matthew.roughan, lewis.mitchell} @adelaide.edu.au

B. Smart would like to acknowledge the support of a Westpac Future Leaders Scholarship. M. Roughan and L. Mitchell are supported by the Australian Government through the Australian Research Council's Discovery Projects funding scheme (project DP210103700).

However, many real sequences have correlations with a longer or more complex history than the simple Markov property allows [LTWW94], [Ber94]. It is not uncommon to find correlations in real data that are so strong that the process is considered to have long-range dependence, which is defined by the tail of the autocorrelation function decaying so slowly that sums over the tail diverge. Other processes with only short-range correlations may still exhibit complex structure of interest.

Markov processes do have a correlation structure, and this can even be long-range dependent for infinite-state Markov chains [CD07], but the autocorrelation structure of a Markov process is difficult to tune, reducing their usefulness in applications that require control over the autocorrelation structure.

Higher-order Markov processes allow the next transition to depend on the previous m states, where m describes the order of the process, and thus allow longer correlations to be built into the process. For a mth-order Markov Process with n possible states, modelling these transitions requires $(n-1)n^m$ parameters, so higher-order models are prone to overfitting as m increases.

The Linear Additive Markov Process (LAMP), proposed by Kumar *et al.* [KMST17], overcomes these challenges. The LAMP can fit a measured autocorrelation structure even for sequences that exhibit long-range dependence. An expressive model, the LAMP model has been shown to have comparable performance to deep sequential models despite their relatively small parameter space [KMST17]. While LAMPs are not as expressive as higher-order Markov processes, their straightforward theoretical results and explainability make them a good model for time-series data where the measured correlation structure is of interest.

This work proves that the entropy rate of a LAMP is equal to that of the underlying first-order Markov Chain. The result is surprising because the structure introduced by the LAMP plays no part in entropy rate, despite having a strong impact on the structure of the process, for instance, through the autocorrelation. A consequence of this result is that the LAMP is able to distinguish between sequences that might look very different due to the presence of long-range dependency (LRD), yet have the same entropy rate – providing a further useful application of the model.

In summary, the contributions of the paper are:

1) A closed form solution giving the entropy rate of a LAMP process, and showing that the entropy rate is exactly that of the underlying Markov chain regardless of the autocorrelation.

2) Use of this result to construct an estimator which is applied to four publicly available datasets to obtain entropy estimates, close to or lower than values obtained using a first-order Markov model. Estimated entropy values are 4.88 bits/symbol for the Lastfm dataset, 2.49 bits/symbol for the BrightKite dataset, 3.18 bits/symbol for the Wikispeedia dataset and 3.61 bits/symbol for the Reuters dataset.
3) An exploration of the impacts of forcing ergodicity in a transition matrix on entropy estimates. This work considers two approaches, using only the largest connected component and adding an artificial fully connected node, with entropy estimates similar between each approach. Analysis into the stability of the estimate against the artificial transition probability was undertaken, and a robust value of $p_{\text{artificial}} = 2^{-15}$ is recommended.

III. Background

We start by defining a standard discrete-time, time-homogeneous Markov chain X_t, a process which underlies the LAMP model. The definitions and notation are standard, but we include them to ensure there is no confusion.

Let S be a state space, which contains n states. Without loss of generality, we label the states $S = [n] = \{1, 2, \ldots, n\}$. We take P to be the $n \times n$ stochastic matrix which defines the transitions between states for the underlying first-order Markov process. It follows that the row sums of P are all equal to 1, and each element is non-negative.

In a first-order Markov process, the probability of transitioning from state x_{t-1} to x_t is given by,

$$\begin{aligned}&\Pr(X_n = x_n | X_0 = x_0, ..., X_{n-1} = x_{n-1})\\ &= \Pr(X_n = x_n | X_{n-1} = x_{n-1})\\ &= P_{x_{n-1}, x_n},\end{aligned}$$

by the Markov property. We also have that the stationary distribution of the Markov process is denoted by the vector π, which is the left eigenvector of P corresponding to the eigenvalue of 1. That is, π satisfies the equation

$$\pi P = \pi.$$

If P defines an ergodic Markov process, this stationary distribution is unique.

The above results can be generalised to countable state processes, however, a finite-state Markov chain cannot reproduce long-range dependency structures. LAMPs afford control over the correlations within the process without requiring an infinite state space.

A. Linear Additive Markov Process (LAMP)

A Linear Additive Markov Process (LAMP) is defined using a first-order Markov process and a probability distribution w on the positive integers that we refer to as the LAMP kernel. Assuming we are at time $n-1$ and wish to calculate the next state X_n we adopt the following two stage process:

1) Select a past state x_{n-q} with probability w_q. This state is also referred to as the *transition state*, X_{T_n}.
2) Determine the state $X_n = x_n$ according to the probabilities P_{x_{n-q}, x_n}.

That is, we use the same state transition probabilities as in the underlying Markov chain, but the historical state on which the transition is based is randomly chosen according to w.

In practice, there are some extra details: (i) the distribution w is typically assumed to have finite support $[k] = \{1, 2, \ldots, k\}$ but it need not be finite, and (ii) the state that is chosen must be from the existing history, which does not extend k steps initially, and so the state chosen as the jumping off point is masked according to $x_{\max\{0, n-q\}}$. The value of k gives the *order* of the LAMP.

An alternative equivalent definition is provided in [KMST17] as follows:

Definition III.1 (LAMP Transitions). Given a stochastic matrix P and a distribution w on $[k]$, the kth-order LAMP evolves according to the following transition probabilities:

$$Pr(X_n = x_n | x_0, ..., x_{n-1}) = \sum_{q=1}^{k} w_q P_{x_{\max\{0, n-q\}}, x_n}$$

In their defining paper Kumar *et al.* [KMST17] provide a number of important results:

1) There exists a LAMP of order k that cannot be approximated with any constant factor by a higher order Markov process of order $m-1$. So the model has more expressive power than a conventional Markov chain, with an exponentially smaller number of parameters.
2) Under mild regularity conditions (*i.e.,* Ergodicity of the underlying Markov chain) LAMPs possess a limiting equilibrium distribution which is the same as that of the underlying simple Markov chain, *i.e.,* the stationary distribution π.
3) The choice of the kernel w determines how the history is included into the next state, and so can incorporate correlations extending as far back as desired, including long-range correlations (for $k \to \infty$). It is not stated, but it is plausible that this is closely related to the convolution of the conventional autocorrelation of the underlying Markov chain and the LAMP kernel w. We show this in action in some later examples.
4) Kumar *et al.* [KMST17] provide a measure of the speed of convergence to the equilibrium distribution, though interestingly these results only apply where w has a finite fourth moment, and so these convergence estimates do not apply in the case of a long-range dependent process.

The authors also provide an approximate Maximum Likelihood Estimator (MLE) to learn a LAMP from data using alternating estimation of w and P.

Thus LAMPs are a simple and expressive model, characterised by a (first-order Markov) transition matrix and a discrete probability distribution on the non-negative integers. Their efficient parameterisation and simple theoretical results mean they can be easily learned and fit to data.

Kumar *et al.* [KMST17] provide many useful results for their model but do not provide an explicit formula for the entropy rate of the process, which is the topic of this paper.

B. Entropy and Entropy Rate

The entropy rate of a sequence is a quantity that describes the self-information the sequence contains. It also represents an asymptotic lower bound on the lossless compression ratio of the sequence and can be used to measure how predictable a sequence is [CT91].

For a discrete random variable, X, the Shannon entropy is denoted H, and is defined to be

$$H(X) = -\sum_x p(x) \log p(x),$$

where $p(\cdot)$ is the probability distribution of X, and with the convention that $0 \log 0 = 0$. The joint entropy of n random variables X_i is the obvious extension:

$$\begin{aligned} &H(X_1, \ldots, X_n) \\ &= -\sum_{x_1, \ldots, x_n} p(x_1, \ldots, x_n) \log p(x_1, \ldots, x_n). \end{aligned}$$

The *entropy rate* for a process $\mathcal{X}$ is just the limiting average of the joint entropy,

$$H(\mathcal{X}) = \lim_{n \to \infty} \frac{1}{n} H(X_1, \ldots, X_n).$$

A standard result [CT91, Thm 4.2.1] links the entropy rate to an equivalent defined in terms of conditional probabilities,

$$H'(\mathcal{X}) = \lim_{n \to \infty} H(X_n | X_1, \ldots, X_{n-1}).$$

For a stationary process the limits H and H' exist and are equal [CT91, Thm 4.2.1], *i.e.,* $H(\mathcal{X}) = H'(\mathcal{X})$. That is, for a stationary stochastic process, the conditional entropy rate and per symbol entropy rate of n random variables are equal in the limit.

The result is convenient, particularly for computing the entropy rate of a stationary Markov chain, which is given by [CT91, Thm 4.2.4] as

$$H(\mathcal{X}) = -\sum_{ij} \pi_i P_{ij} \log P_{ij},$$

where P is the probability transition matrix, and π the stationary distribution. The primary result of this paper is that this formula also provides the entropy rate of a LAMP.

IV. The Entropy Rate of a LAMP

Theorem IV.1 (Entropy rate of a LAMP). For a LAMP defined by an underlying stationary, first-order Markov Chain with transition matrix P, and kernel distribution w on $[k]$, the entropy rate of the LAMP is

$$H(\mathcal{X}) = -\sum_{ij} \pi_i P_{ij} \log P_{ij}, \tag{1}$$

where π is the stationary distribution of the Markov chain.

Proof. We begin by considering the entropy rate of n realisations from a LAMP, using known results from information theory about conditional entropy and the entropy rate of a first-order Markov Chain. As noted above, for a stationary Markov chain

$$H(\mathcal{X}) = H'(\mathcal{X}).$$

Assume that at time $n-1$ we choose $X_{T_n} = X_{n-q}$ to be the transition state according to distribution w. The state at time n depends only on X_{n-q} and so the conditional entropy

$$H(X_n | X_{n-1}, ..., X_0, q) = H(X_n | X_{n-q}, q),$$

where we include q explicitly in the conditioned term for notational convenience. A transition from state $X_{n-q} \to X_n$ in the LAMP is governed by the same probability transition matrix P as the transition $X'_{n-1} \to X'_n$ in the underlying Markov chain (we use X' here to denote the underlying Markov chain). Furthermore [KMST17] shows that if the Markov chain corresponding to P is ergodic, then its stationary distribution π' is also the stationary distribution of the LAMP, so we denote both by π. Hence, the conditional entropy is the entropy rate of the Markov chain, *i.e.,*

$$\begin{aligned} H(X_n | X_{n-q}, q) &= H(X'_n | X'_{n-1}) \\ &= -\sum_{ij} \pi_i P_{ij} \log P_{ij}. \end{aligned}$$

by [CT91, Thm 4.2.4]. Note that this decouples the entropy rate from the choice of q (assuming n is large enough that $n - q \geq 0$ for all choices of q, or alternatively presuming an infinite history of the stationary process).

Now, we can remove the conditioning on q by noting that the overall conditional entropy is a probabilistically weighted sum over the choices of q, *i.e.,*

$$\begin{aligned} &H(X_n | X_{n-1}, ..., X_0) \\ &= \sum_k P(T_n = n - q) H(X_n | X_{n-q}, q) \\ &= -\sum_{ij} \pi_i P_{ij} \log P_{ij} \sum_q w_q. \end{aligned}$$

Noting that w is a proper probability distribution, its sum is 1, and hence the conditional entropy is given by the constant term, and hence the limit is trivial and the result follows.

$$H(\chi_{LAMP}) = H(\chi),$$

where $H(\chi)$ is the entropy of the underlying first-order Markov Chain. □

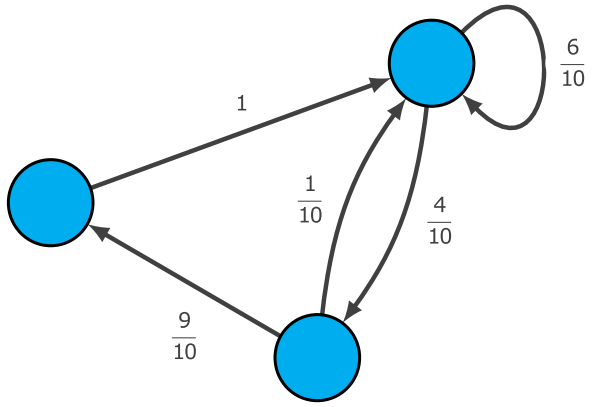
The most noteworthy feature of this result is that LAMP kernel w plays no part in the formula. This is surprising because the kernel clearly affects the autocorrelation structure of the process. Naive intuition suggests that a process with stronger autocorrelations should present less information at each time step, and hence have a smaller entropy rate.

While at first surprising, this result is intuitive as once the conditioning state, X_{T_n} is selected, the process transitions in the same manner as a first-order Markov Chain. The randomness introduced by the choice of q, balances with the increased predictability which correlation with the history of the process provides.

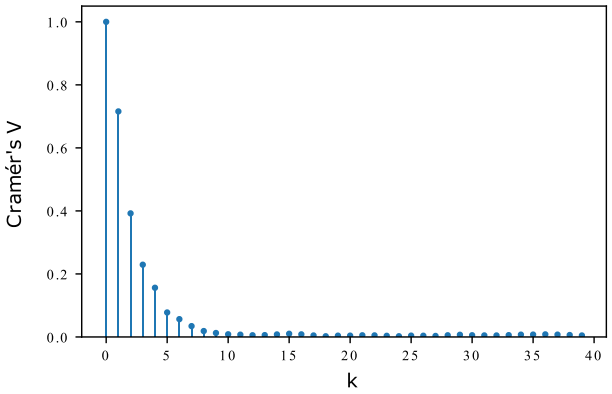
Alternatively, we might think of this as increasing the correlations in the process, while also increasing the variance per time step, maintaining a constant entropy rate. However, we work here in the context of a finite alphabet of (not necessarily ordinal) states, so the common notions of variance have been subtly generalised by the LAMP model.

Shannon entropy rate for all models : 0.45627 bits/symbol.

(i) Underlying Markov Model

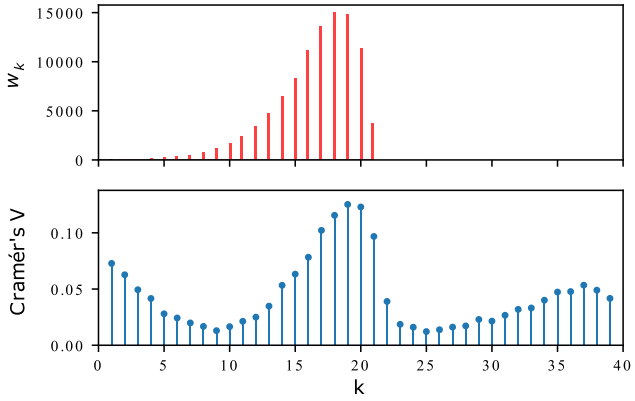


(ii) First Order Markov Model



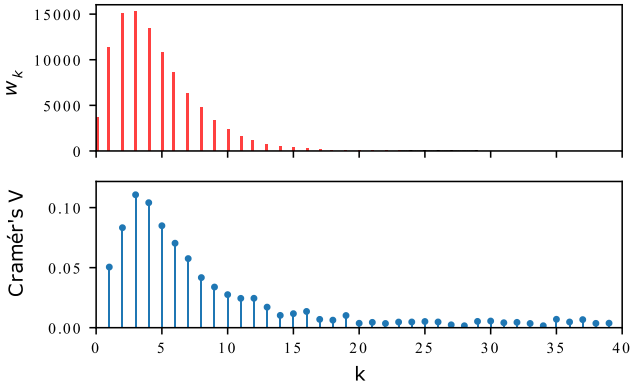
Sample path: [2,3,1,2,2,3,1,2,3,1,2,3,1,2,2,2,2,2,2,3 1,2,2,3,1,2,2,3,1,2,2,3,2,3,1,2,2,3,1,2]

(iii) Negatively Skewed Function



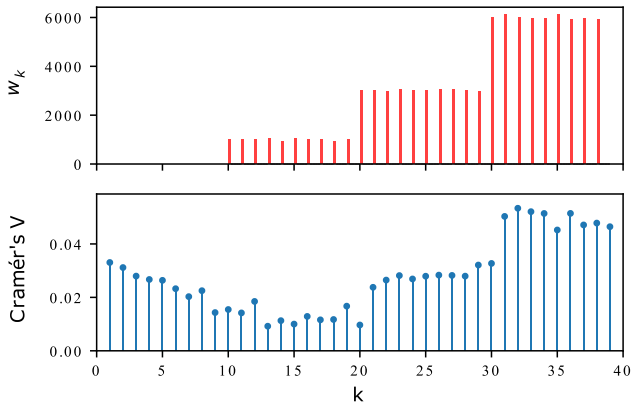
Sample path: [1, 2, 2, 2, 3, 2, 2, 2, 3, 2, 2, 2, 3, 1, 2, 3, 1, 2, 2, 2 1, 2, 1, 1, 1, 1, 3, 2, 3, 2, 2, 2, 2, 1, 1, 2, 2, 2, 2, 2]

(iv) Positively Skewed Function



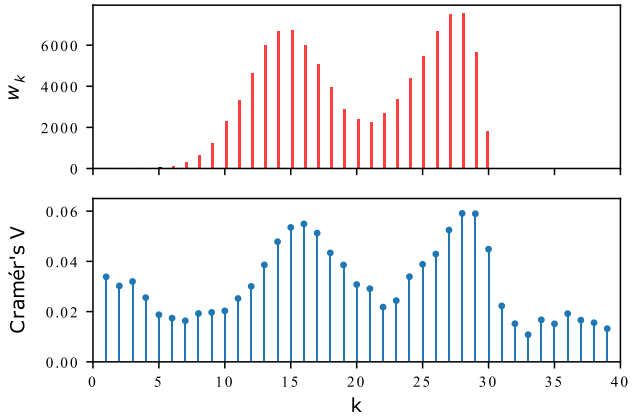
Sample path: [2, 1, 1, 2, 2, 2, 2, 2, 3, 3, 2, 2, 2, 2, 1, 1, 2, 2, 2, 3 2, 3, 2, 2, 3, 2, 2, 2, 2, 2, 2, 2, 3, 2, 3, 3, 1, 2, 1, 1]

(v) Step Function



Sample path: [2, 2, 2, 3, 2, 2, 2, 2, 2, 1, 2, 2, 3, 2, 1, 3, 2, 3, 3, 3 3, 3, 2, 1, 3, 2, 3, 1, 2, 2, 1, 2, 2, 2, 2, 3, 2, 3, 1, 2]

(vi) Multimodal Function



Sample path: [2, 2, 2, 1, 2, 1, 2, 2, 1, 2, 2, 2, 1, 2, 2, 2, 2, 3, 2, 3 3, 2, 2, 2, 2, 2, 2, 2, 2, 2, 3, 2, 2, 2, 2, 2, 3, 2, 3]

Figure 1. (i) A diagram of the underlying transition matrix for each of the models shown in (ii) - (vi). While these models all produce sample paths with differing dependency structures, they all have the same entropy rate, as proven in Theorem 4.1. In blue, the dependency structure of each model is visualised using Cramér's V for lags up to 40. This provides a measure of correlation for symbolic variables and is analogous to considering a lagged-cross correlation function. In (ii), the dependency structure is shown for a first-order Markov model. In (iii) - (vi) we see the frequency of w_k, or the distribution of the size of the backward step directly influences the shape of the dependency structure. While the Cramér's V is calculated for a lag of 0 for the first-order Markov Model, it is excluded from plots (iii)-(vi) since the value is much larger due to the Markov Chain's memoryless property. It was excluded to allow the similarity between the frequency distribution of step size and dependency structure to be more easily seen.

This useful result provides a model able to capture complex dependency structures in data but maintain consistent randomness on a symbol by symbol level.

V. Model Comparisons

To demonstrate the significance of the LAMP model, we will consider the dependency structure of 5 comparable models: a first-order Markov model, and four alternative LAMP models with differing w_k (Figure 1). Each of these models share the same transition matrix illustrated in Figure 1 (i). Since they share a transition matrix, these sequences all share the same stationary distribution and entropy rate of 0.45627 bits per symbol, so each model is superficially similar.

The dependency structure is measured by calculating Cramér's V statistic for various lags. Based on Pearson's Chi-Squared test, the Cramér's V statistic measures the association between two discrete, nominal variables. To measure the self association for a sequence with a given lag l, pairs of values from the sequence at index i and $i + l$ are used to construct a contingency table, which is used to calculate Cramér's V statistic. The statistic ranges from 0 to 1, where 0 indicates no correlation, and 1 indicates a perfect relationship between the pairs of variables. Calculating this value for various values of l produces a measure of dependency structure comparable to a lagged cross-correlation plot, but without assigning ordinal value to the state labels.

To illustrate the autocorrelation structure for a standard first-order Markov model, it is shown in plot Figure 1 (ii) along with a short sample path. Figure 1 (iii)-(vi) show the Cramér's V Statistic for the LAMP models in comparison to the choice of w. Note that lag = 0 is not plotted in these cases, as this value will always be 1.

It is evident that the dependency structure mirrors the shape of w_k, demonstrating that the LAMP framework provides a useful model for generating sequences with a desired dependency structure, although it is important to note this structure will repeat and diminish for integer multiples of the original lag, as well as including structure from the original Markov model, so the two are not identical. However, there is an evident ability to tune the autocorrelation of a LAMP, while preserving other properties such as the stationary distribution and entropy rate.

VI. Empirical Evaluation

Following the same approach as the original LAMP paper [KMST17], we fit and estimate Shannon Entropy rates on four publicly available datasets, LastFM, BrightKite, Wikispeedia and Reuters. Each dataset is comprised of a number of distinct sequences which represent the activity of an individual or a single process. We compare the entropy rate estimates of (1) an empirical estimator, (2) the estimator obtained by fitting a Markov chain, and (3) the entropy estimate obtained by fitting a LAMP.

Our code is available at github.com/bridget-smart/LAMPEntropyEstimates and the links to each dataset are provided in the text below.

The LastFM dataset contains the user activity from the music streaming service last.fm for 992 users. Users can choose to listen to stations based on a genre or artist, a particular song, and can share their listening activity. The dataset contains the user, timestamp, artist and song, but here we construct sequences using only the user and artist. Each sequence contains data for a single user, with items representing different artists. Containing 19M items, the data is available at http://ocelma.net/MusicRecommendationDataset/lastfm-1K.html.

The BrightKite dataset contains check-in data from the location-based social networking service BrightKite, where users share their checked-in locations. Each sequence represents a single user, with items representing the locations which the user has checked into. 772,967 locations and 51,406 users are represented in the dataset. The data is available at https://snap.stanford.edu/data/loc-brightkite.html.

The Wikispeedia navigation path dataset is comprised of sequences representing a path along Wikipedia links from a condensed version of Wikipedia. These sequences are collected using the online game Wikispeedia where users race to navigate between articles. This dataset contains 51,318 complete paths. The dataset is available at https://snap.stanford.edu/data/wikispeedia.html.

For the Reuters dataset we use the Reuters-21578, Distribution 1.0 benchmark corpus, containing newswire articles, with each sequence representing a single article and items representing distinct words. This dataset is available at www.nltk.org/nltk_data.

Following the same process as in [KMST17], consecutive repeated values within each path repeated values are removed to prevent self-loops from appearing in the transition matrices. Consecutive repeated values are also not meaningful in our datasets, as they do not contain temporal data. We also replace values which appear less than 10 times, or less than 50 times for the LastFM dataset with a unique token, representing a less frequently visited state. This preprocessing ensured our model fit was consistent with the original paper, and across both the LAMP and first-order Markov models.

LAMP models are fit to each dataset using the original LAMP code available at github.com/google-research/google-research/tree/master/lamp. Modified code used to obtain the LAMP transition matrices for this paper is available at github.com/bridget-smart/modified_lamp. The code used to obtain the entropy estimates is available at github.com/bridget-smart/LAMPEntropyEstimates.

While a modified version of the author's original code was used to fit the LAMP model, care was taken to exactly reproduce the preprocessing steps outlined in Kumar *et al.* [KMST17] when fitting the remaining models. The code from Kumar *et al.* [KMST17] fits the LAMP model using a subset of the data and a cross-validation technique to overcome computational and time constraints. Therefore, the estimate using the LAMP model was fit on a smaller subset of the data than other estimates. Discrepancies in the transition matrix size between the first-order Markov and LAMP models indicate the differences caused by this.

Stationary distribution derivations are valid where the transition matrix is irreducible, but the data is imperfect, and this condition is not true for all transition matrices estimated from these datasets. We overcome this in two ways: (1) by only considering the largest strongly-connected component of the graph generated by considering the transition matrix as a directed network; and (2) by artificially connecting each distinct communicating class to the first state with transition weight $p_{\text{artificial}}$.

The first approach assumes that the largest connected component of the transition matrix is representative of the chain's behaviour. To assess this we consider how many nodes are excluded from that component. For the first-order Markov models, the number of nodes ignored when only considering the largest fully connected component were 1, 1255, 68, 2 for each of the four datasets, LASTFM, BRIGHTKITE, WIKISPEEDIA and REUTERS respectively. The total number of nodes in each of these models were 23870, 43596, 3425 and 8465 respectively. The BRIGHTKITE dataset has the largest proportion of nodes removed (0.02879), which suggests the component can be representative.

In the LAMP models 237 (of 17766 for LASTFM), 2591 (of 25348 for BRIGHTKITE), 2775 (of 3504 for WIKISPEEDIA) and 3094 (of 6822 for REUTERS) nodes were ignored while only considering the largest fully connected component to obtain the entropy estimates for the LAMP models. These proportions are generally higher than those for the first-order Markov model, ranging from 0.7920 to 0.01334, which may result in more instability within the estimated entropy values.

The second approach forces irreducibility by ensuring that the chain mixes, and all states can be visited. It, and in particular a discussion of how to choose $p_{\text{artificial}}$ is given in Appendix A, but a brief description is given here. This approach uses the assumption that it is possible to communicate between each possible pair of states, an assumption which is reasonable for most of our datasets, as, for instance, it is possible for users to listen to any pair of artists, but rare transitions are not likely to be observed, and so including a small transition probability to connect the network is reasonable. Artificially inducing transitions between states which are otherwise disconnected, modifies the behaviour of the chain. It follows that these artificial links should have a small weight since these transitions have not been observed within the data sample suggesting they are rare, but the weight of these links needs to be large enough to ensure mixing can occur, stable estimates can be calculated and errors introduced by computer precision are avoided. The choice of $p = 2^{-15}$ and $p = 2^{-10}$ is discussed in Appendix A.

The results are shown in Table I. The estimators referred to in the table are defined in detail below. At a high level:

- *Shannon Empirical* estimators use
$$H(\chi) = -\sum_x \hat{p}(x) \log \hat{p}(x),$$
where $\hat{p}(x)$ is an estimated probability distribution. We consider three cases, defined below.
- Markov estimates use the standard Markov-chain entropy given in Equation 1, with empirical estimates of the transition and stationary probabilities. They do so either on the Largest Connected Component (CC) or on a chain with artificially induced irreducibility as described earlier.
- LAMP estimates use the LAMP entropy, which is also given in Equation 1, with empirical estimates of the transition and stationary probabilities, but note that the transition matrices for LAMP models will differ from those of the Markov case. These estimators also use either on the Largest Connected Component (CC) or on a chain with artificially induced irreducibility as described earlier.

Shannon estimators are applied to a probability distribution, but the structure of the data provides several alternative distributions to consider.

The data used contains multiple sample paths, *i.e.,* we have a set of n subsequences $\{y_i\}_{i=1}^n$, each composed of m_i items, $\{s_{ij}\}_{j=1}^{m_i}$ for a total of N realisations across all subsequences.

The *Sequence Level* estimator uses the frequency of each item across all subsequences to estimate its probability of occurrence, *i.e.,*
$$\hat{p}(x) = \frac{1}{N} \sum_{i=1}^{n} \sum_{j=1}^{m_i} \mathbb{I}(x = s_{ij}),$$
where $\mathbb{I}(x = s_{ij})$ is an indicator variable which takes the value 1 if the $x = y_i$.

The *Path Level* estimator uses the frequency of each symbol in a sequence to estimate a simple Shannon entropy for each sequence, and then averages over these, *i.e.,*
$$\hat{p}_i(x) = \frac{1}{m_i} \sum_{j=1}^{m_i} \mathbb{I}(x = s_{ij}),$$
where $\mathbb{I}(x = s_{ij})$ is an indicator variable which takes the value 1 if $x = s_{ij}$, and
$$\hat{H}_i(y_i) = \sum_x \hat{p}_i(x) \log \hat{p}_i(x),$$
and
$$\hat{H}(\chi) = \frac{1}{n} \sum_{i=1}^{n} \hat{H}_i(y_i),$$

The *Stationary Distribution* estimator uses as its probability distribution the stationary distribution as estimated through one of the Markov models.

The results shown in Table I provide several insights:

1) The Shannon sequence-level estimator grossly overestimates entropy values for all cases. The most likely scenario is that there just are not enough sampled sequences to properly represent the underlying distribution, so the calculation is essentially computing the entropy of a distribution close to the uniform distribution, *i.e.,* near the maximum possible value.

[1]Values calculated using an artificial edge weight of 2^{-15}, except for the value for the BRIGHTKITE dataset.

[2]This value was calculated using an articifical edge weight of 2^{-10}, rather than the value of 2^{-15} used for all the other estimates. For more details see Appendix A.

[3]Values calculated using an artificial edge weight of 2^{-15}.

Estimate Type	Subtype	Data Set LASTFM	BRIGHTKITE	WIKISPEEDIA	REUTERS
Shannon Empirical	Sequence Level	11.83825	9.83813	9.97541	9.35027
Shannon Empirical	Path Level	6.77366	0.88488	2.53676	5.13913
Shannon Empirical	Stationary Distribution	8.20559	6.72001	6.67657	6.47187
First-order Markov	Largest CC	3.69304	1.67316	3.14187	4.09229
First-order Markov	Induced Irreducibility[1]	3.69334	1.58479[2]	3.93053	4.50134
LAMP	Largest CC	4.88141	2.49536	3.18030	3.61305
LAMP	Induced Irreducibility[3]	4.88092	2.53492	3.18052	3.61328

Table I

ENTROPY ESTIMATES FOR EACH OF THE FOUR DATASETS USING SEVEN ESTIMATION METHODS. TWO MAIN APPROACHES ARE USED TO ENSURE TRANSITION MATRICES ARE ERGODIC: ONLY CONSIDERING THE LARGEST CONNECTED COMPONENT; AND ADDING AN ARTIFICIAL STATE WITH TRANSITION WEIGHT $p_{\text{ARTIFICIAL}}$. BOTH OF THESE APPROACHES GIVE SIMILAR RESULTS FOR EACH OF THE FOUR DATASETS.

2) The Shannon entropy of the stationary distribution is also a fair over-estimate indicating that the transition structure is important in these sequences.
3) The Shannon path-level estimator produces some overestimates, but also a very low value for the `BrightKite` data. This anomalous result may arise here for several reasons, but it is noteworthy that the Shannon estimator can be biased on short data, and as it is being applied here to each subsequence and averages, we should expect bias in the overall estimate. At best it appears to be a sensitive and hard to interpret measure.
4) There is close agreement for entropy estimators using the largest connected component, and induced irreducibility. This similarity is most pronounced for the LAMP models. That suggests that the manner in which we corrected for reducible processes does not matter, but that the LAMP models are also more robust to details such as this.
5) Lower estimated entropies are often indicative of a better model (for the purpose of estimating entropy) that captures more of the predictability of the sequence. The estimates generated using the LAMP models are the lowest estimates for the WIKISPEEDIA and REUTERS datasets. The lower entropy estimate values imply that more information about the sequence is captured by the LAMP model than the underlying first-order Markov model. This implies that the model captures more information about the behaviour of the system, so the generated sequence is more predictable. By better capturing the long-term dependency structures in the data, we achieve a more meaningful model. The estimates obtained using the LAMP model appear appropriate for all datasets. This supports the claims of [KMST17], that the LAMP model provides a better, succinct, representation of these datasets, but our claim is slightly stronger in the context of entropy because they compare against k-order and adaptive Markov chains.
6) For the LASTFM and BRIGHTKITE datasets, the estimates obtained using the first-order Markov models are lower than those obtained with the LAMP model. These two datasets represent artist sequences on a music streaming service and checked in user locations and it is possible that these sequences can be accurately captured without the LAMP structure. This is somewhat in contrast with [KMST17], but note that this finding is only in respect to the entropy estimates and there are other facets of a model that are also important.

VII. CONCLUSION

By deriving a theoretical result for the entropy rate of a linear additive Markov Process, we extend the versatility of LAMPs to information-centic applications and derive a surprising equality between the entropy rate of a LAMP and the underlying first-order Markov Chain. A consequence of this is that realisations of processes having different long-range dependency structures – and sample paths which are therefore quite different from each other – might nonetheless be indistinguishable based on entropy rates. This is because the entropy rates depend only on the underlying transition matrix structure, and not the long-range dependence structure. The LAMP however provides a means to distinguish between such processes, by visualisation of the long-range dependency structure. We demonstrate this via numerical simulations and use of the Cramer's V statistic. While LAMPs exhibit long-term dependency structures, their symbol-by-symbol dependency is low due to the randomness introduced by the distribution over the choice of conditioning state. Linear Additive Markov Processes present an intuitive yet useful extension of traditional Markov chain models to incorporate long-range dependence. This work demonstrates that LAMPs can be understood within a traditional information-theoretic framework, and many further results are possible in this context. Future work will explore the application of LAMPs to understanding long-range phenomena such as in social media communications [MMNB17], to provide enhancements to network traffic measurements such as [NR13], and to derive online estimators for entropy such as in [RVA98].

REFERENCES

[Ber94] J. Beran. *Statistics for Long-Memory Processes*. Chapman and Hall, New York, 1994.

[CD07] K. Carpio and D. Daley. Long-range dependence of Markov chains in discrete time on countable state space. *Journal of Applied Probability*, 44(4):1047–1055, 2007.

[CT91] T. Cover and J. Thomas. *Elements of information theory*. Wiley series in telecommunications. Wiley, New York, 1991.

[KMST17] R. Kumar, M.Raghu, T. Sarlós, and A. Tomkins. Linear additive Markov processes. In *Proceedings of the 26th International Conference on World Wide Web*, WWW '17, page 411–419, Republic and Canton of Geneva, CHE, 2017. International World Wide Web Conferences Steering Committee.

[LTWW94] W. Leland, M. Taqqu, W. Willinger, and D. Wilson. On the self-similar nature of Ethernet traffic (extended version). *IEEE/ACM Transactions on Networking*, 2(1):1–15, Feb 1994.

[MMNB17] P. Mathews, L. Mitchell, G. Nguyen, and N. Bean. The nature and origin of heavy tails in retweet activity. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 1493–1498, 2017.

[Nor98] J. Norris. *Markov chains*. Number 2. Cambridge University Press, 1998.

[NR13] H. Nguyen and M. Roughan. Rigorous statistical analysis of Internet loss measurements. *IEEE/ACM Transactions on Networking*, 21(3):734–745, 2013.

[RVA98] M. Roughan, D. Veitch, and P. Abry. On-line estimation of the parameters of long-range dependence. In *IEEE GLOBECOM*, volume 6, pages 3716–3721, 1998.

APPENDIX A
ON CONNECTING A MARKOV CHAIN: CHOOSING ARTIFICIAL STATES AND CONNECTION PROBABILITIES

For a unique stationary distribution of a Markov Chain to exist, it is necessary for the Markov Chain to be ergodic [Nor98]. By adding an artificial state to the chain, which is connected to all other states with some transition probability $p_{\text{artificial}}$, we can guarantee the chain is ergodic, and that a unique stationary distribution exists. The choice of hyper-parameter $p_{\text{artificial}}$ is critical to ensuring this artificial state does not dramatically alter the behaviour of the Markov chain and instead acts as a link between otherwise disconnected communicating classes.

The parameter $p_{\text{artificial}}$ was chosen for each of the four datasets for both the LAMP and first-order Markov Chain approaches by calculating entropy estimates for $p_{\text{artificial}} = 2^{-i}$ for $i = 1, ..., 25$ for the first-order Markov estimates and $i = 1, ..., 50$ for the LAMP estimates. Since we are selecting for small values of $p_{\text{artificial}}$, it is also important to ensure we avoid numerical precision errors. It is important that the estimated value is robust to small changes in $p_{\text{artificial}}$. By selecting values of $p_{\text{artificial}}$ for which the estimate is stable and appears to have converged, we can estimate the entropy of the process. The true entropy value will differ between models and datasets, so we are not concerned with the value which the estimate converges to, only the stability of the estimate.

Generally, these estimates approach a stable value for larger values of i, although it is not known if this trend would continue, due to precision limitations. The largest value for which the estimate is stable was taken to perform each of the estimates was used.

The results of these simulations are shown in Figure 2. For all dataset model combinations except for the entropy estimates obtained using the first-order Markov model for the WIKISPEEDIA and the REUTERS datasets, the entropy values approach a limiting value from above, with most estimates converging to a stable value. The only estimate which does not appear to converge is the estimate obtained with the first-order model for the BRIGHTKITE dataset, which begins to flatten between -7 to -10, but then approaches zero. This may be due to numerical limitations, but this behaviour will be explored in future work.

In Figure 2, the entropy estimates are normalised to have a minimum value of 0 and a maximum value of 1, to enable the shape of the curves to be easily compared. This highlights the shape of the curve and emphasises the direction of convergence. These normalised convergence curves for the WIKISPEEDIA and REUTERS datasets overlap for the first-order Markov model, so the values for the REUTERS dataset are offset by +0.04 for visualisation. Similarly, for the LAMP model convergence curves, a small offset is added to REUTERS and BRIGHTKITE datasets in the LAMP model visualisation (+0.01 and -0.005 respectively).

The choice of $p_{\text{artificial}} = 2^{-15}$ appears appropriate for all datasets where the estimates appear to converge, while remaining large enough to avoid precision errors. This artificial transition probability is equivalent to observing a single transition amongst 32 768, which is sufficiently small to have a minimal impact on the scales we are considering. It is important that this choice is evaluated for different datasets, where the size of the data may make a different value more suitable.

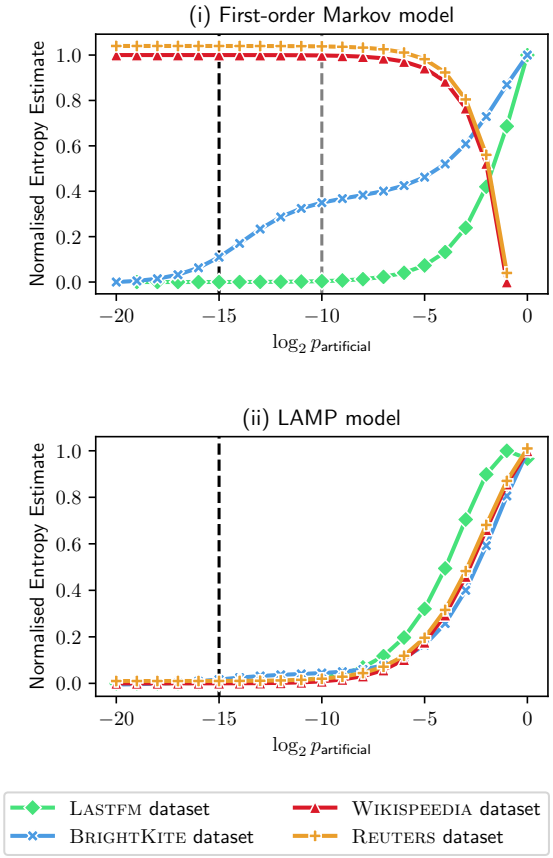


Figure 2. Plots to show convergence for the normalised entropy estimate value against $\log_2 p_{\text{artificial}}$, which was used to ensure ergodicity. (i) Shows hyperparameter sensitivity for the first-order Markov models and (ii) for the LAMP models. Each plot shows the effect of the weight of this artificial link on both the first-order Markov model estimate and the estimate obtained using the LAMP model for each dataset. We aim to find a region where the estimates are insensitive to the choice of hyperparameter. A dashed black line on each plot indicates the value when the artificial link weight is 2^{-15}. This value was chosen as a global value, since it is a reasonable choice for all dataset model combinations, apart from the BrightKite dataset first-order Markov model, when a value of 2^{-10} was used to obtain the final estimate. This alternative value is indicated by a grey dashed line. The convergence curve for the WIKISPEEDIA and REUTERS datasets overlapped for the first-order Markov model, so the values for the REUTERS dataset was offset by +0.04 for visualisation. Small vertical offset was also added to the REUTERS and BRIGHTKITE datasets in the LAMP model visualisation (+0.01 and -0.005 respectively).

Chapter 6

Conclusion

This thesis builds upon and improves the robustness of existing techniques to measure pairwise relationships and information flows in real-world networks, ultimately improving our ability to model complex systems accurately.

In the first part of this thesis, we consider the association between point processes, conducting a survey of existing measures of association in Chapter 2. This survey aims to improve the accessibility of approaches between fields, so unified notation and definitions are presented. Four classes of techniques are considered; heuristic, information-theoretic, stochastic and machine learning based, with the general limitations and applications of each class described. This is followed by a discussion of practical considerations and accompanied by a conceptual framework designed to unify existing work. While this survey increases the accessibility of techniques between fields, empirical evaluation is an important step to ensure approaches are robust and contextually appropriate.

In Chapter 3, a simulation and evaluation framework is described and applied to simulate realistic data from a network of point processes and evaluate measures of association. When the evaluation procedure is applied to heuristic and information-theoretic estimators, it reveals how sensitive each measure is to particular features of association. By improving the quality and accessibility of simulated data in combination with a versatile framework, this simulation further improves the dissemination potential of association measures for point processes. The evaluation framework is equipped with methods for measuring direct coordination and generating process similarity, which will allow approaches to be validated against a range of meaningful artificial data sets.

The focus of the survey and evaluation framework was on pairwise associations, where data is used to construct edges between processes corresponding to an underlying network structure. Techniques to directly infer the structure of the network from higher-order effects were not considered, and a focus on these global methods is an avenue for future work.

Measuring global network dynamics is a growing area of interest, so universal theoretical frameworks and generalised notations have potential to shape new methods, to measure dynamics of networks from point processes, and prevent the divergence of domain-specific techniques. As matrix similarity is used, the proposed evaluation framework is currently only able to consider approaches that summarise the pairwise relationship with a single value. This approach cannot be easily extended to techniques which summarise pairwise relationships with more complex objects. By revising how the performance of association measures is evaluated, future work can extend this evaluation procedure to stochastic and machine learning based approaches, which often require comparison between distributions.

Non-parametric entropy estimators are the focus of the second half of this work, and a procedure for empirical significance testing is described. In Chapter 4, this significance test, alongside an aggregated information flow measure, is described and applied to a Twitter data set containing Tweets related to the 2022 Russian invasion of Ukraine. These results reveal insights into how information flows between bot and non-bot accounts. We found that the largest information flows originate in the pro-Russian non-bot account group, and information flows into non-bot account groups were only significant for balanced and pro-Ukrainian accounts. Additional analysis was undertaken to enhance the insights from the information flow network, identifying a relationship between bot activity and overall non-bot sentiment (with $p = 0.000376$).

Unlike the first part of the thesis, temporal data was utilised alongside content and account details. With the rich information available on social media, there is opportunity to measure the differences between models fit using only temporal data and models fit using more complete information. Understanding how the performance of temporal models are affected by missing information can contribute to a better understanding of how reliable the insights from temporal models are when predicting account behaviour on online social networks. This could be achieved by estimating a network structure using the association measures explored in Chapter 2, selected through the proposed validation procedure, and comparing this structure to the network estimated using information flows.

Applying this estimator to real data revealed issues with the convergence rate of the estimator, and further work was done to calculate the theoretical entropy rate of a Linear Additive Markov Process (LAMP) in Chapter 5. The LAMP model was identified as a useful tool for exploring the entropy estimator, as the ability to specify the long-term dependency structure allows the complexity of natural language to be reproduced, providing a more meaningful test for an entropy estimator designed for natural language data. This opens up an avenue for future work; using the LAMP model to explore convergence issues with the original entropy estimator, to propose corrections for these estimates, and to understand the effects of long-term dependency on the convergence rate. As an expressive Markov model with capacity for long-term dependency structures, the LAMP model has

potential to be used to simulate more complex processes, leading to more realistic and robust method validation tools.

Aside from the individual avenues of future work, this thesis aims to highlight the importance of evaluating techniques in meaningful data settings, and aims to improve the accessibility of techniques to researchers tackling complex problems across many domains.

Appendix A

Complete sample results for evaluation of association measures

This appendix includes sample output from the evaluation procedure outlined in Chapter 3. Each plot shows the complete evaluation results for a single association measure. The top row of each plot shows performance for detecting directly coordinated behaviours and the bottom row for similarity between generating processes. Each column concerns a different feature of the data; the variance of coordination relationship strengths, the average number of events per process, the average time between self generated events, the coordination delay function, and the average time spend in either coordinating and non-coordinating states.

A.0.1 Metrics

Coordination windows

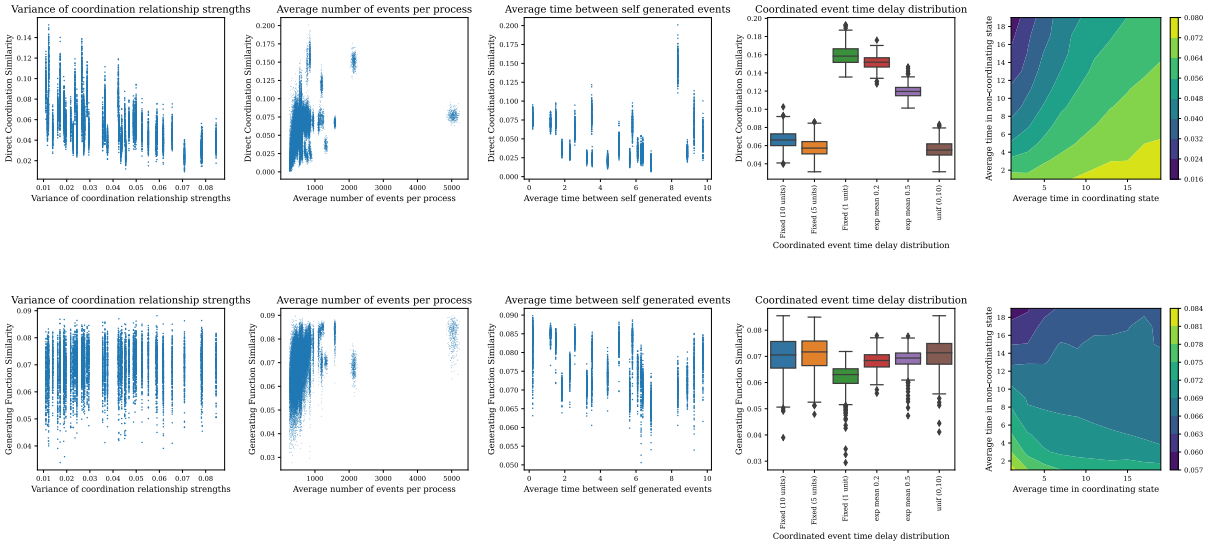


Figure A.1: Co-occurrence counts, the coordination window size is 1 unit

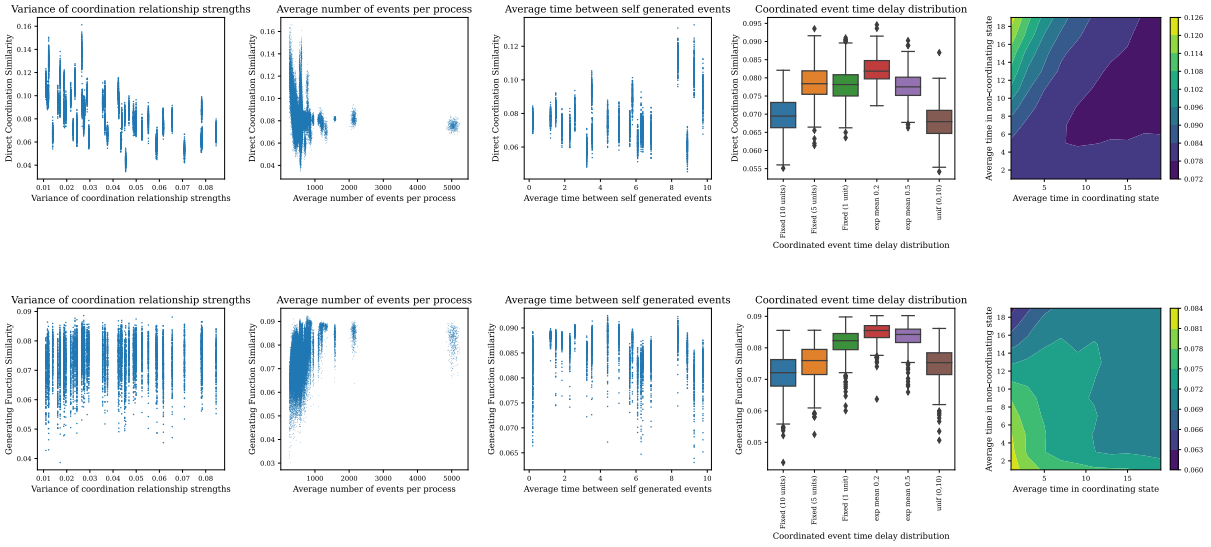


Figure A.2: Co-occurrence counts, the coordination window size is 5 units

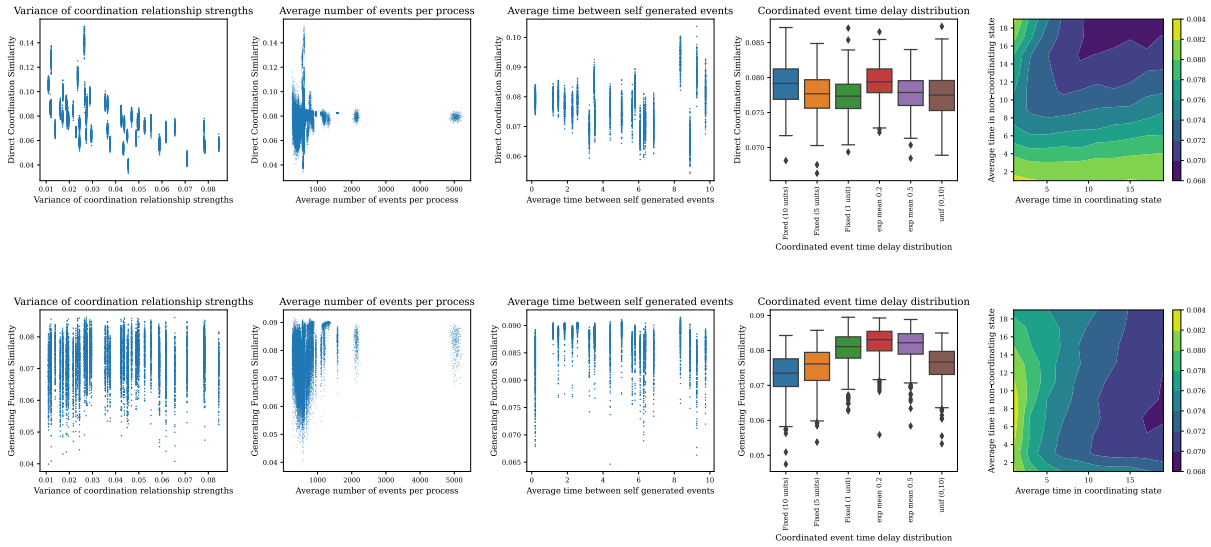


Figure A.3: Co-occurrence counts, the coordination window size is 10 units

Inter-event time statistics

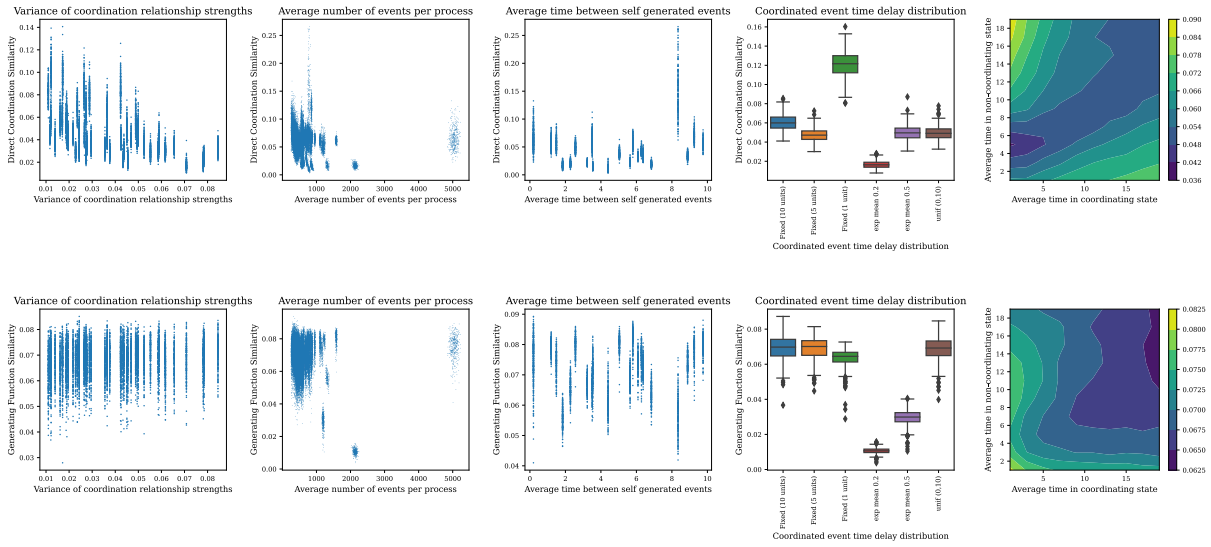


Figure A.4: Median inter-event time

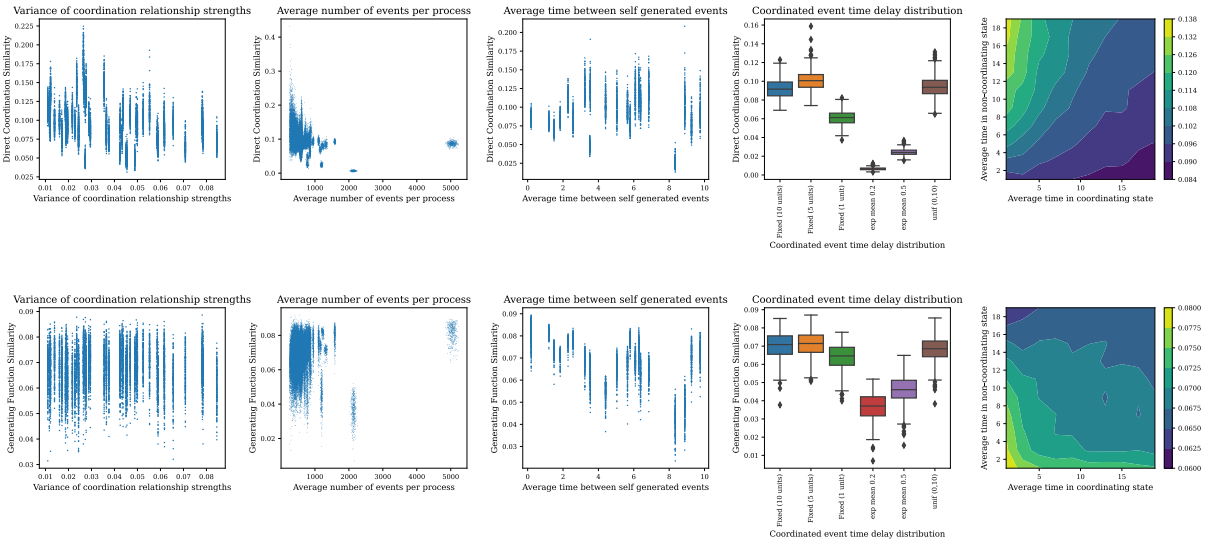


Figure A.5: Exponentially penalised inter-event time

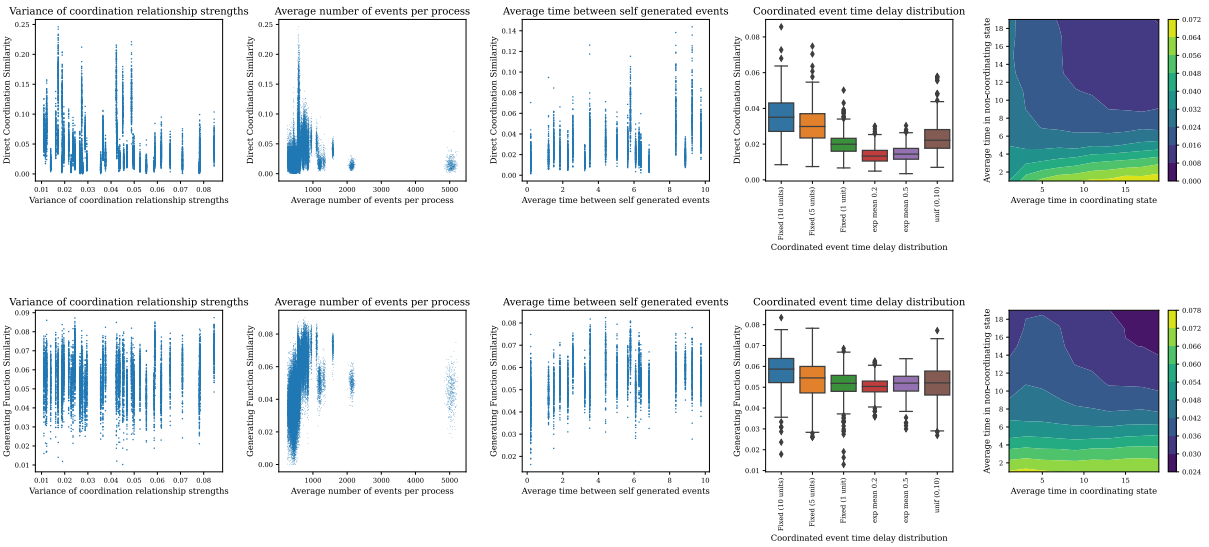


Figure A.6: Variance of inter-event times

A.0.2 Infomation-theoretic

Transfer entropy

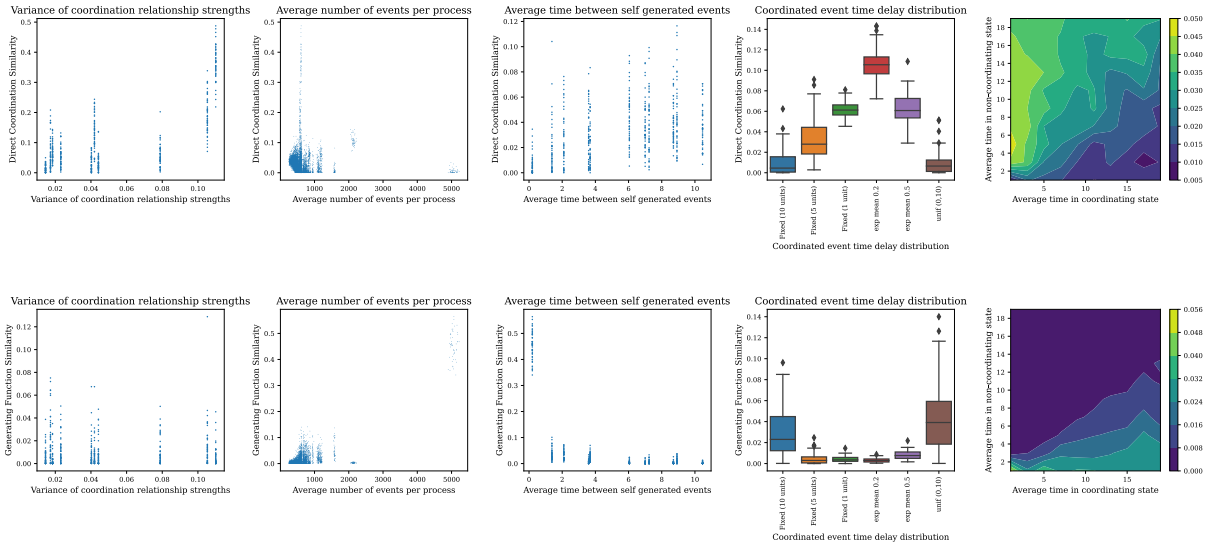


Figure A.7: Transfer Entropy

Time agnostic entropy

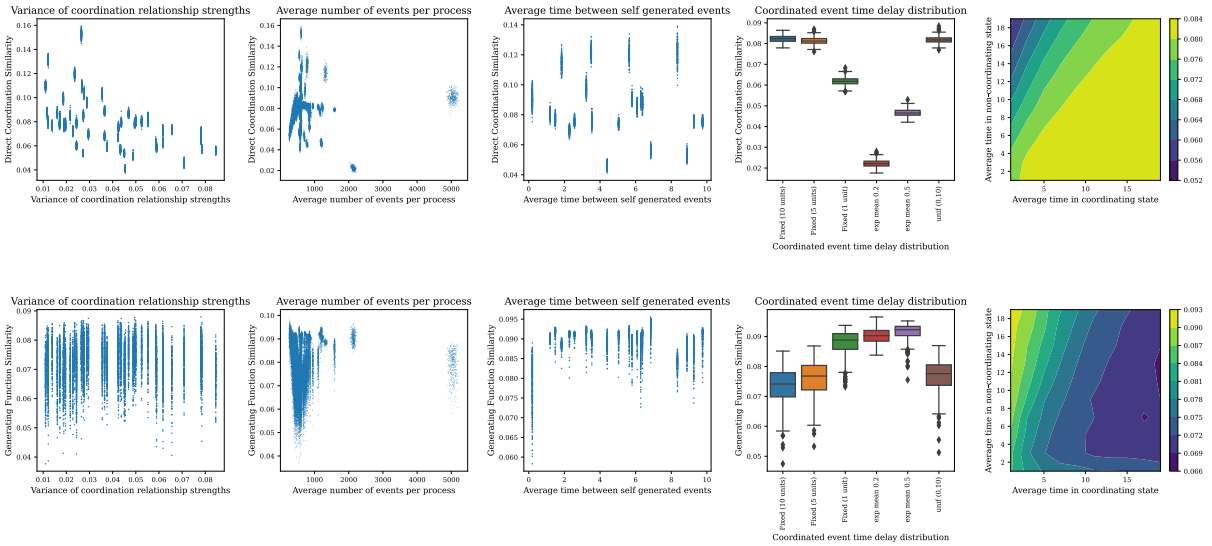


Figure A.8: Time agnostic entropy

Bibliography

Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74(1), 47–97, January 2002. doi:10.1103/RevModPhys.74.47.

Meysam Alizadeh, Jacob N. Shapiro, Cody Buntain, and Joshua A. Tucker. Content-based features predict social media influence operations. *Science Advances*, 6(30), article eabb5824, July 2020. doi:10.1126/sciadv.abb5824.

Eiman Alothali, Nazar Zaki, Elfadil A. Mohamed, and Hany Alashwal. Detecting Social Bots on Twitter: A Literature Review. In *2018 International Conference on Innovations in Information Technology (IIT)*, pp. 175–180, November 2018. doi:10.1109/INNOVATIONS.2018.8605995.

Corrado Ameli and Roberto Sassi. Parametric Estimation of Entropy Using High Order Markov Chains for Heart Rate Variability Analysis. In *2018 Computing in Cardiology Conference (CinC)*, volume 45, pp. 1–4, September 2018. doi:10.22489/CinC.2018.190.

Emmanuel Bacry and Jean-François Muzy. First- and second-order statistics characterization of hawkes processes and non-parametric estimation. *IEEE Transactions on Information Theory*, 62(4), 2184–2202, 2016. doi:10.1109/TIT.2016.2533397.

James P. Bagrow and Lewis Mitchell. The quoter model: A paradigmatic model of the social flow of written information. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 28(7), article 075304, July 2018. doi:10.1063/1.5011403.

James P. Bagrow, Xipei Liu, and Lewis Mitchell. Information flow reveals prediction limits in online social activity. *Nature Human Behaviour*, 3(2), 122–128, February 2019. doi:10.1038/s41562-018-0510-5.

Albert-László Barabási and Zoltán N. Oltvai. Network biology: Understanding the cell's functional organization. *Nature Reviews Genetics*, 5(2), 101–113, February 2004. doi:10.1038/nrg1272.

Marco Bastos and Johan Farkas. "Donald Trump Is My President!": The Internet Research Agency Propaganda Machine. *Social Media + Society*, 5(3), article 2056305119865466, April 2019. doi:10.1177/2056305119865466.

Luc Bauwens and Nikolaus Hautsch. Modelling Financial High Frequency Data Using Point Processes. In Thomas Mikosch, Jens-Peter Kreiß, Richard A. Davis, and Torben Gustav Andersen, editors, *Handbook of Financial Time Series*, pp. 953–979. Springer, Berlin, Heidelberg, 2009. doi:10.1007/978-3-540-71297-8_41.

Petro Bodnar. *Detecting Patterns of Coordinated News Article Dissemination*. PhD thesis, Ukrainian Catholic University, 2021.

M. I. Borrajo, C. Comas, Sergi Costafreda-Aumedes, and Jorge Mateu. Stochastic smoothing of point processes for wildlife-vehicle collisions on road networks. *Stochastic Environmental Research and Risk Assessment*, 36(6), 1563–1577, June 2022. doi:10.1007/s00477-021-02072-3.

Steven L. Bressler and J. A. Scott Kelso. Coordination Dynamics in Cognitive Neuroscience. *Frontiers in Neuroscience*, 10, article 397, 2016. doi:https://doi.org/10.3389/fnins.2016.00397.

Ruichu Cai, Siyu Wu, Jie Qiao, Zhifeng Hao, Keli Zhang, and Xi Zhang. THPs: Topological Hawkes Processes for Learning Causal Structure on Event Sequences. *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15, 2022. doi:10.1109/TNNLS.2022.3175622.

Álvaro Cartea and Sebastian Jaimungal. Modelling Asset Prices for Algorithmic and High-Frequency Trading. *Applied Mathematical Finance*, 20(6), 512–547, December 2013. doi:10.1080/1350486X.2013.771515.

Lars-Erik Cederman. Endogenizing geopolitical boundaries with agent-based modeling. *Proceedings of the National Academy of Sciences*, 99(suppl_3), 7296–7303, May 2002. doi:10.1073/pnas.082081099.

Zhiyi Chi and Daniel Margoliash. Temporal precision and temporal drift in brain and behavior of zebra finch song. *Neuron*, 32(5), 899–910, 2001. doi:https://doi.org/10.1016/S0896-6273(01)00524-4.

Zhiyi Chi, Wei Wu, Zach Haga, Nicholas G. Hatsopoulos, and Daniel Margoliash. Template-based spike pattern identification with linear convolution and dynamic time warping. *Journal of Neurophysiology*, 97(2), 1221–1235, February 2007. doi:10.1152/jn.00448.2006.

Oliver M. Cliff, Joseph T. Lizier, Naotsugu Tsuchiya, and Ben D. Fulcher. Unifying Pairwise Interactions in Complex Dynamics, January 2022. doi:10.48550/ARXIV.2201.11941.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley Series in Telecommunications. Wiley, New York, 1991.

Larissa Doroshenko and Josephine Lukito. Trollfare: Russia's Disinformation Campaign During Military Conflict in Ukraine. *International Journal of Communication*, 15 (0), 4662–4689, October 2021.

Lilia R. Duskaeva and Yulia M. Konyaeva. Trolling in Russian Media. *Journal of Organizational Culture, Communications and Conflict*, 20(4), 57–72, 2016.

Abdulrahman M El-Sayed, Peter Scarborough, Lars Seemann, and Sandro Galea. Social network analysis and agent-based modeling in social epidemiology. *Epidemiologic Perspectives & Innovations*, 9(1), article 1, December 2012. doi:10.1186/1742-5573-9-1.

Daisuke Endo, Ryota Kobayashi, Ramon Bartolo, Bruno B. Averbeck, Yasuko Sugase-Miyamoto, Kazuko Hayashi, Kenji Kawano, Barry J. Richmond, and Shigeru Shinomoto. A convolutional neural network for estimating synaptic connectivity from spike trains. *Scientific Reports*, 11(1), article 12087, June 2021. doi:10.1038/s41598-021-91244-w.

Andrew Feutrill and Matthew Roughan. A Review of Shannon and Differential Entropy Rate Estimation. *Entropy*, 23(8), article 1046, August 2021. doi:10.3390/e23081046.

V. Filimonov and D. Sornette. Apparent criticality and calibration issues in the Hawkes self-excited point process model: Application to high-frequency financial data. *Quantitative Finance*, 15(8), 1293–1314, November 2015. doi:10.1080/14697688.2015.1032544.

Wolfgang Fischer and Kathleen Meier-Hellstern. The Markov-modulated Poisson process (MMPP) cookbook. *Performance Evaluation*, 18(2), 149–171, September 1993. doi:10.1016/0166-5316(93)90035-S.

Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5), 75–174, February 2010. doi:10.1016/j.physrep.2009.11.002.

Santo Fortunato and Darko Hric. Community detection in networks: A user guide. *Physics Reports*, 659, 1–44, November 2016. doi:10.1016/j.physrep.2016.09.002.

Kai S. Gansel and Wolf Singer. Detecting Multineuronal Temporal Patterns in Parallel Spike Trains. *Frontiers in Neuroinformatics*, 6, article 18, 2012. doi:10.3389/fninf.2012.00018.

Martha Garcia-Garcia, Cesar Marquez Chin, and Milos Popovic. Operant conditioning of motor cortex neurons reveals neuron-subtype-specific responses in a brain-machine interface task. *Scientific Reports*, 10, article 19992, November 2020. doi:10.1038/s41598-020-77090-2.

Fabio Giglietto, Nicola Righetti, Luca Rossi, and Giada Marino. It takes a village to manipulate the media: Coordinated link sharing behavior during 2018 and 2019 Italian elections. *Information, Communication & Society*, 23(6), 867–891, March 2020. doi:10.1080/1369118X.2020.1739732.

Timothy Graham. Coordination Network Toolkit. Queensland University of Technology Digital Observatory, 2020. doi:10.25912/RDF_1632782596538. `https://pypi.org/project/coordination-network-toolkit/`.

Timothy Graham, Axel Bruns, Guangnan Zhu, and Rod Campbell. *Like a Virus: The Coordinated Spread of Coronavirus Disinformation.* The Australia Institute, Canberra, A.C.T, May 2020.

Timothy Graham, Axel Bruns, Daniel Angus, Edward Hurcombe, and Sam Hames. #ISTandWithDan versus #DictatorDan: The polarised dynamics of Twitter discussions about Victoria's COVID-19 restrictions. *Media International Australia*, 179(1), 127–148, May 2021. doi:10.1177/1329878X20981780.

Sonja Grün, Markus Diesmann, and Ad Aertsen. Unitary Events in Multiple Single-Neuron Spiking Activity: I. Detection and Significance. *Neural Computation*, 14(1), 43–80, January 2002. doi:10.1162/089976602753284455.

David J Hand and Veronica Vinciotti. Local Versus Global Models for Classification Problems: Fitting Models Where it Matters. *The American Statistician*, 57(2), 124–131, May 2003. doi:10.1198/0003130031423.

Alan G. Hawkes. Spectra of Some Self-Exciting and Mutually Exciting Point Processes. *Biometrika*, 58(1), 83–90, 1971. doi:10.2307/2334319.

Saike He, Xiaolong Zheng, Daniel Zeng, Kainan Cui, Zhu Zhang, and Chuan Luo. Identifying Peer Influence in Online Social Networks Using Transfer Entropy. In G. Alan Wang, Xiaolong Zheng, Michael Chau, and Hsinchun Chen, editors, *Intelligence and Security Informatics*, Lecture Notes in Computer Science, pp. 47–61, Berlin, Heidelberg, 2013. Springer. doi:10.1007/978-3-642-39693-9_6.

Andreas Henelius, Kai Puolamäki, Henrik Boström, and Panagiotis Papapetrou. Clustering with Confidence: Finding Clusters with Statistical Guarantees, December 2016. doi:https://doi.org/10.48550/arXiv.1612.08714.

Hideitsu Hino, Ken Takano, and Noboru Murata. mmpp: A Package for Calculating Similarity and Distance Metrics for Simple and Marked Temporal Point Processes. *The R Journal*, 7(2), 237–248, 2015. doi:10.32614/RJ-2015-033.

Amelia M. Jamison, David A. Broniatowski, and Sandra Crouse Quinn. Malicious Actors on Twitter: A Guide for Public Health Researchers. *American Journal of Public Health*, 109(5), 688–692, May 2019. doi:10.2105/AJPH.2019.304969.

Ovidiu F. Jurjuţ, Medorian Gheorghiu, Wolf Singer, Danko Nikolić, and Raul C. Mureşan. Hold Your Methods! How Multineuronal Firing Ensembles Can Be Studied Using Classical Spike-Train Analysis Techniques. *Frontiers in Systems Neuroscience*, 13, article 21, 2019. doi:https://doi.org/10.3389/fnsys.2019.00021.

Robert E. Kass and Valérie Ventura. A Spike-Train Probability Model. *Neural Computation*, 13(8), 1713–1720, August 2001. doi:10.1162/08997660152469314.

Franziska B. Keller, David Schoch, Sebastian Stier, and JungHwan Yang. Political Astroturfing on Twitter: How to Coordinate a Disinformation Campaign. *Political Communication*, 37(2), 256–280, October 2020. doi:10.1080/10584609.2019.1661888.

Arik Kershenbaum, Vlad Demartsev, David E. Gammon, Eli Geffen, Morgan L. Gustison, Amiyaal Ilany, and Adriano R. Lameira. Shannon entropy as a robust estimator of Zipf's Law in animal vocal communication repertoires. *Methods in Ecology and Evolution*, 12 (3), 553–564, mar 2021. doi:10.1111/2041-210X.13536.

Dongwoo Kim, Timothy Graham, Zimin Wan, and Marian-Andrei Rizoiu. Analysing user identity via time-sensitive semantic edit distance (t-SED): A case study of Russian trolls on Twitter. *Journal of Computational Social Science*, 2(2), 331–351, July 2019. doi:10.1007/s42001-019-00051-x.

Matthias Kirchner. An estimation procedure for the Hawkes process. *Quantitative Finance*, 17(4), 571–595, September 2016. doi:10.1080/14697688.2016.1211312.

Andrei A. Klishin and Igor Chilingarian. Explaining the stellar initial mass function with the theory of spatial networks. *The Astrophysical Journal*, 824(1), article 17, June 2016. doi:10.3847/0004-637X/824/1/17.

Ryota Kobayashi and Renaud Lambiotte. TiDeH: Time-Dependent Hawkes Process for Predicting Retweet Dynamics. In *Tenth International AAAI Conference on Web and Social Media*, March 2016.

I. Kontoyiannis, P. H. Algoet, Yu. M. Suhov, and A. J. Wyner. Nonparametric entropy estimation for stationary processes and random fields, with applications to English text. *IEEE Transactions on Information Theory*, 44(3), 1319–1327, May 1998. doi:10.1109/18.669425.

Ravi Kumar, Maithra Raghu, Tamás Sarlós, and Andrew Tomkins. Linear Additive Markov Processes. In *Proceedings of the 26th International Conference on World Wide Web*, pp. 411–419, Perth Australia, April 2017. International World Wide Web Conferences Steering Committee. doi:10.1145/3038912.3052644.

Hyeong-Joon Kwon, Tae-Hoon Lee, Jung-Hyun Kim, and Kwang-Seok Hong. Improving Prediction Accuracy Using Entropy Weighting in Collaborative Filtering. In *2009 Symposia and Workshops on Ubiquitous, Autonomic and Trusted Computing*, pp. 40–45, Brisbane, Australia, 2009. IEEE. doi:10.1109/UIC-ATC.2009.50.

M. Laghate and D. Cabric. Learning Wireless Networks' Topologies Using Asymmetric Granger Causality. *IEEE Journal of Selected Topics in Signal Processing*, 12(1), 233–247, February 2018. doi:10.1109/JSTSP.2017.2787478.

Pantelis Linardatos, Vasilis Papastefanopoulos, and Sotiris Kotsiantis. Explainable AI: A Review of Machine Learning Interpretability Methods. *Entropy*, 23(1), article 18, December 2020. doi:10.3390/e23010018.

Fangxin Liu, Wenbo Zhao, Yongbiao Chen, Zongwu Wang, and Li Jiang. SpikeConverter: An Efficient Conversion Framework Zipping the Gap between Artificial Neural Networks and Spiking Neural Networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 1692–1701, June 2022. doi:10.1609/aaai.v36i2.20061.

Luca Luceri, Silvia Giordano, and Emilio Ferrara. Detecting Troll Behavior via Inverse Reinforcement Learning: A Case Study of Russian Trolls in the 2016 US Election. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 14, pp. 417–427, May 2020. doi:10.1609/icwsm.v14i1.7311.

S. S. Melnik and O. V. Usatenko. Entropy and long-range memory in random symbolic additive Markov chains. *Physical Review E*, 93(6), article 062144, June 2016. doi:10.1103/PhysRevE.93.062144.

Michael Merry, Pat Riddle, and Jim Warren. A mental models approach for defining explainable artificial intelligence. *BMC Medical Informatics and Decision Making*, 21 (1), article 344, December 2021. doi:10.1186/s12911-021-01703-7.

Gorana Mijatovic, Yuri Antonacci, Tatjana Loncar-Turukalo, Ludovico Minati, and Luca Faes. An Information-Theoretic Framework to Measure the Dynamic Interaction Between Neural Spike Trains. *IEEE Transactions on Biomedical Engineering*, 68 (12), 3471–3481, December 2021. doi:10.1109/TBME.2021.3073833.

F. Musmeci and D. Vere-Jones. A space-time clustering model for historical earthquakes. *Annals of the Institute of Statistical Mathematics*, 44(1), 1–11, March 1992. doi:10.1007/BF00048666.

M. E. J. Newman. *Networks*. Oxford University Press, Oxford, second edition, 2018.

Lynnette Hui Xian Ng and Kathleen M. Carley. Online Coordination: Methods and Comparative Case Studies of Coordinated Groups across Four Events in the United States. In *14th ACM Web Science Conference 2022*, pp. 12–21, New York, NY, USA, June 2022. Association for Computing Machinery. doi:10.1145/3501247.3531542.

Nicolas Koumchatzky and Anton Andryeyev. Using Deep Learning at Scale in Twitter's Timelines. https://blog.twitter.com/engineering/en_us/topics/insights/2017/using-deep-learning-at-scale-in-twitters-timelines.html, May 2017.

B. V. North, D. Curtis, and P. C. Sham. A Note on the Calculation of Empirical P Values from Monte Carlo Procedures. *American Journal of Human Genetics*, 71(2), 439–441, August 2002. doi:10.1086/341527.

Leonardo Novelli, Patricia Wollstadt, Pedro Mediano, Michael Wibral, and Joseph T. Lizier. Large-scale directed network inference with multivariate transfer entropy and hierarchical statistical testing. *Network Neuroscience*, 3(3), 827–847, July 2019. doi:10.1162/netn_a_00092.

Yosihiko Ogata. Space-Time Point-Process Models for Earthquake Occurrences. *Annals of the Institute of Statistical Mathematics*, 50(2), 379–402, June 1998. doi:10.1023/A:1003403601725.

Leslie C. Osborne, Stephanie E. Palmer, Stephen G. Lisberger, and William Bialek. The Neural Basis for Combinatorial Coding in a Cortical Population Response. *Journal of Neuroscience*, 28(50), 13522–13531, December 2008. doi:10.1523/JNEUROSCI.4390-08.2008.

António R. C. Paiva, Il Park, and José C. Príncipe. A Reproducing Kernel Hilbert Space Framework for Spike Train Signal Processing. *Neural Computation*, 21(2), 424–449, February 2009. doi:10.1162/neco.2008.09-07-614.

G. Palm, A. M. H. J. Aertsen, and G. L. Gerstein. On the significance of correlations among neuronal spike trains. *Biological Cybernetics*, 59(1), 1–11, June 1988. doi:10.1007/BF00336885.

Il Memming Park, Sohan Seth, Antonio R. C. Paiva, Lin Li, and Jose C. Principe. Kernel methods on spike train space for neuroscience: A tutorial. *IEEE Signal Processing Magazine*, 30(4), 149–160, July 2013. doi:10.1109/MSP.2013.2251072.

Junhyung Park, Frederic Paik Schoenberg, Andrea L. Bertozzi, and P. Jeffrey Brantingham. Investigating Clustering and Violence Interruption in Gang-Related Violent Crime Data Using Spatial–Temporal Point Processes With Covariates. *Jour-*

nal of the American Statistical Association, 116(536), 1674–1687, April 2021. doi:10.1080/01621459.2021.1898408.

Karl Pearson. Notes on the History of Correlation. *Biometrika*, 13(1), 25–45, 1920. doi:10.2307/2331722.

Alessio Perinelli, Michele Castelluzzo, Davide Tabarelli, Veronica Mazza, and Leonardo Ricci. Relationship between mutual information and cross-correlation time scale of observability as measures of connectivity strength. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 31(7), article 073106, July 2021. doi:10.1063/5.0053857.

Donald H. Perkel, George L. Gerstein, and George P. Moore. Neuronal Spike Trains and Stochastic Point Processes: II. Simultaneous spike trains. *Biophysical Journal*, 7 (4), 419–440, July 1967a. doi:10.1016/S0006-3495(67)86597-4.

Donald H. Perkel, George L. Gerstein, and George P. Moore. Neuronal Spike Trains and Stochastic Point Processes: I. The single spike train. *Biophysical Journal*, 7(4), 391–418, July 1967b. doi:10.1016/S0006-3495(67)86596-2.

Tyson Pond, Saranzaya Magsarjav, Tobin South, Lewis Mitchell, and James P. Bagrow. Complex Contagion Features without Social Reinforcement in a Model of Social Information Flow. *Entropy*, 22(3), article 265, February 2020. doi:10.3390/e22030265.

Pietro Quaglio, Vahid Rostami, Emiliano Torre, and Sonja Grün. Methods for identification of spike patterns in massively parallel spike trains. *Biological Cybernetics*, 112 (1-2), 57–80, April 2018. doi:10.1007/s00422-018-0755-0.

R. Quian Quiroga, T. Kreuz, and P. Grassberger. Event synchronization: A simple and fast method to measure synchronicity and time delay patterns. *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics*, 66(4), article 041904, October 2002. doi:10.1103/PhysRevE.66.041904.

Alex Reinhart. A Review of Self-Exciting Spatio-Temporal Point Processes and Their Applications. *Statistical Science*, 33(3), 299–318, August 2018. doi:10.1214/17-STS629.

Marian-Andrei Rizoiu, Young Lee, Swapnil Mishra, and Lexing Xie. Hawkes processes for events in social media. In Shih-Fu Chang, editor, *Frontiers of Multimedia Research*, pp. 191–218. Association for Computing Machinery and Morgan & Claypool, New York, NY, USA, December 2017. doi:10.1145/3122865.3122874.

Bleema Rosenfeld, Osvaldo Simeone, and Bipin Rajendran. Spiking Generative Adversarial Networks With a Neural Network Discriminator: Local Training, Bayesian Models, and Continual Meta-Learning. *IEEE Transactions on Computers*, 71(11), 2778–2791, November 2022. doi:10.1109/TC.2022.3191738.

Walter Rudin. *Functional Analysis*. International Series in Pure and Applied Mathematics. McGraw-Hill, New York, second edition, 1991.

Alessio Sancetta. Estimation for the prediction of point processes with many covariates. *Econometric Theory*, 34(3), 598–627, April 2017. doi:10.1017/S0266466617000172.

Mohsen Sayyadiharikandeh, Onur Varol, Kai-Cheng Yang, Alessandro Flammini, and Filippo Menczer. Detection of Novel Social Bots by Ensembles of Specialized Classifiers. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pp. 2725–2732, Virtual Event Ireland, October 2020. ACM. doi:10.1145/3340531.3412698.

Fabian Schäfer, Stefan Evert, and Philipp Heinrich. Japan's 2014 General Election: Political Bots, Right-Wing Internet Activism, and Prime Minister Shinzō Abe's Hidden Nationalist Agenda. *Big Data*, 5(4), 294–309, December 2017. doi:10.1089/big.2017.0049.

Elad Schneidman, William Bialek, and Michael J. Berry. Synergy, Redundancy, and Independence in Population Codes. *Journal of Neuroscience*, 23(37), 11539–11553, December 2003. doi:10.1523/JNEUROSCI.23-37-11539.2003.

S. Schreiber, J. M. Fellous, D. Whitmer, P. Tiesinga, and T. J. Sejnowski. A new correlation-based measure of spike timing reliability. *Neurocomputing*, 52–54, 925–931, June 2003. doi:10.1016/S0925-2312(02)00838-X.

Thomas Schreiber. Measuring Information Transfer. *Physical Review Letters*, 85(2), 461–464, July 2000. doi:10.1103/PhysRevLett.85.461.

Steven Scott and Padhraic Smyth. The Markov Modulated Poisson Process and Markov Poisson Cascade With Applications to Web Traffic Modelling. *Bayesian Statistics*, 7, 1–10, January 2003.

M. Ángeles Serrano, Marián Boguñá, and Alessandro Vespignani. Extracting the multiscale backbone of complex weighted networks. *Proceedings of the National Academy of Sciences*, 106(16), 6483–6488, April 2009. doi:10.1073/pnas.0808904106.

C. E. Shannon. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27(3), 379–423, July 1948. doi:10.1002/j.1538-7305.1948.tb01338.x.

Karishma Sharma, Yizhou Zhang, Emilio Ferrara, and Yan Liu. Identifying Coordinated Accounts on Social Media through Hidden Influence and Group Behaviours. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 1441–1451, Virtual Event Singapore, August 2021. ACM. doi:10.1145/3447548.3467391.

David P. Shorten, Richard E. Spinney, and Joseph T. Lizier. Estimating Transfer Entropy in Continuous Time Between Neural Spike Trains or Other Event-Based Data. *PLOS Computational Biology*, 17(4), article e1008054, April 2021. doi:10.1371/journal.pcbi.1008054.

Auste Simkute, Ewa Luger, Bronwyn Jones, Michael Evans, and Rhianne Jones. Explainability for experts: A design framework for making algorithms supporting expert decisions more explainable. *Journal of Responsible Technology*, 7–8, article 100017, October 2021. doi:10.1016/j.jrt.2021.100017.

Bridget Smart, Joshua Watt, Sara Benedetti, Lewis Mitchell, and Matthew Roughan. #IStandWithPutin Versus #IStandWithUkraine: The Interaction of Bots and Humans in Discussion of the Russia/Ukraine War. In Frank Hopfgartner, Kokil Jaidka, Philipp Mayr, Joemon Jose, and Jan Breitsohl, editors, *Social Informatics*, volume 13618, pp. 34–53. Springer International Publishing, Cham, 2022. doi:10.1007/978-3-031-19097-1_3.

Tobin South, Bridget Smart, Matthew Roughan, and Lewis Mitchell. Information flow estimation: A study of news on Twitter. *Online Social Networks and Media*, 31, article 100231, September 2022. doi:10.1016/j.osnem.2022.100231.

C. Spearman. The Proof and Measurement of Association between Two Things. *The American Journal of Psychology*, 15(1), 72–101, 1904. doi:10.2307/1412159.

Richard E. Spinney, Mikhail Prokopenko, and Joseph T. Lizier. Transfer entropy in continuous time, with applications to jump and neural spiking processes. *Physical Review E*, 95(3), article 032319, March 2017. doi:10.1103/PhysRevE.95.032319.

Satoshi Suzuki, Yoshito Hirata, and Kazuyuki Aihara. Definition of distance for marked point process data and its application to recurrance plot-based analysis of exchange tick data of foreign currencies. *International Journal of Bifurcation and Chaos*, 20 (11), 3699–3708, November 2010. doi:10.1142/S0218127410027970.

Emmanuelle Tognoli, Mengsen Zhang, Armin Fuchs, Christopher Beetle, and J. A. Scott Kelso. Coordination Dynamics: A Foundation for Understanding Social Behavior. *Frontiers in Human Neuroscience*, 14, article 317, 2020. doi:https://doi.org/10.3389/fnhum.2020.00317.

William Turin. Fitting probabilistic automata via the EM algorithm. *Communications in Statistics. Stochastic Models*, 12(3), 405–424, January 1996. doi:10.1080/15326349608807392.

M. C. van Rossum. A novel spike distance. *Neural Computation*, 13(4), 751–763, April 2001. doi:10.1162/089976601300014321.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Greg Ver Steeg and Aram Galstyan. Information transfer in social media. In *Proceedings of the 21st International Conference on World Wide Web*, pp. 509–518, Lyon France, April 2012. ACM. doi:10.1145/2187836.2187906.

Jonathan D Victor. Spike train metrics. *Current opinion in neurobiology*, 15(5), 585–592, October 2005. doi:10.1016/j.conb.2005.08.002.

Jonathan D Victor and Keith P Purpura. Metric-space analysis of spike trains: Theory, algorithms and application. *Network: Computation in Neural Systems*, 8(2), 127–164, January 1997. doi:10.1088/0954-898X_8_2_003.

Giulia Vilone and Luca Longo. Notions of explainability and evaluation approaches for explainable artificial intelligence. *Information Fusion*, 76, 89–106, December 2021. doi:10.1016/j.inffus.2021.05.009.

Isaac Waller and Ashton Anderson. Quantifying social organization and political polarization in online platforms. *Nature*, 600(7888), 264–268, December 2021. doi:10.1038/s41586-021-04167-x.

Jiqian Wang, Feng Ma, M.i.m. Wahab, and Dengshi Huang. Forecasting China's Crude Oil Futures Volatility: The Role of the Jump, Jumps Intensity, and Leverage Effect. *Journal of Forecasting*, 40(5), 921–941, 2021. doi:10.1002/for.2752.

Zhehui Wang, Xiaozhe Gu, Rick Siow Mong Goh, Joey Tianyi Zhou, and Tao Luo. Efficient Spiking Neural Networks With Radix Encoding. *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–13, 2022. doi:10.1109/TNNLS.2022.3195918.

Shixian Wen, Allen Yin, Tommaso Furlanello, M. G. Perich, L. E. Miller, and Laurent Itti. Rapid adaptation of brain–computer interfaces to new neuronal ensembles or participants via generative modelling. *Nature Biomedical Engineering*, pp. 1–13, November 2021. doi:10.1038/s41551-021-00811-z.

Gavin Wilde and Justin Sherman. Targeting Ukraine through Washington: Russian election interference, Ukraine, and the 2024 US election. `https://www.atlanticcouncil.org/in-depth-research-reports/issue-brief/targeting-ukraine-through-washington/`, March 2022.

Jing Wu, Owen G. Ward, James Curley, and Tian Zheng. Markov-modulated Hawkes processes for modeling sporadic and bursty event occurrences in social interactions. *The Annals of Applied Statistics*, 16(2), 1171–1190, June 2022. doi:10.1214/21-AOAS1539.

A.D. Wyner and J. Ziv. Some asymptotic properties of the entropy of a stationary ergodic data source with applications to data compression. *IEEE Transactions on Information Theory*, 35(6), 1250–1258, November 1989. doi:10.1109/18.45281.

Shunxin Xiao, Shiping Wang, Yuanfei Dai, and Wenzhong Guo. Graph neural networks in node classification: Survey and evaluation. *Machine Vision and Applications*, 33 (1), article 4, January 2022. doi:10.1007/s00138-021-01251-0.

Qiusheng Yuan, Peifang Wang, Xun Wang, Bin Hu, and Li Tao. Phytoremediation of cadmium-contaminated sediment using *Hydrilla Verticillata* and *Elodea Canadensis* harbor two same keystone *Rhizobacteria Pedosphaeraceae* and *parasegetibacter*. *Chemosphere*, 286(1), article 131648, January 2022. doi:10.1016/j.chemosphere.2021.131648.

Qiang Zhang, Aldo Lipani, Omer Kirnap, and Emine Yilmaz. Self-Attentive Hawkes Process. In *Proceedings of the 37th International Conference on Machine Learning*, Proceedings of Machine Learning Research, pp. 11183–11193. PMLR, November 2020.

Yizhou Zhang, Karishma Sharma, and Yan Liu. Vigdet: Knowledge informed neural temporal point process for coordination detection on social media. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pp. 3218–3231. Curran Associates, Inc., 2021.

Dengyong Zhou, Olivier Bousquet, Thomas Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems*, volume 16. MIT Press, 2003.

J. Ziv and N. Merhav. A measure of relative entropy between individual sequences with application to universal classification. *IEEE Transactions on Information Theory*, 39 (4), 1270–1279, July 1993. doi:10.1109/18.243444.