DI Barbara Gigerl

Utility of Transfer Entropy for the Extraction of Influence Networks

MASTER'S THESIS

to achieve the university degree of
Diplom-Ingenieurin
Master's degree programme: Computer Science

submitted to
Graz University of Technology

Supervisor: Assoc.Prof. Dipl.-Ing. Dr.techn. Denis Helic
Institute of Interactive Systems and Data Science

Graz, October 2019

AFFIDAVIT

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master's thesis.

Date	Signature

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das in TUGRAZonline hochgeladene Textdokument ist mit der vorliegenden Masterarbeit identisch.

Datum	Unterschrift

Abstract

More and more people use social networks like Twitter, Instagram, and Facebook today. Understanding their behavior when interacting with social media becomes very important. One aspect is the analysis of influential users in social networks and the construction of influence networks. Influence networks are weighted, directed networks with social media users as nodes. A link between user A and user B means that A influences B. The link weight indicates how strong the influence is. One possibility to extract such influence networks is transfer entropy, an information-theoretic measure. Transfer entropy indicates for two stochastic processes X and Y, how much the uncertainty of Y is reduced by knowing X. If the reduction is high, the behavior of Y can be predicted easily by the knowledge of X, and X strongly influences Y.

This thesis investigates the problems and advantages of using transfer entropy to extract influence networks. Researchers have shown that transfer entropy can be used to analyze social media structures. Nevertheless, their research has been limited to empirical datasets and failed to address synthetic data in detail. We conduct a large-scale analysis of the utilization of transfer entropy based on synthetic data, generated by Poisson point processes. The data generation process supports several tunable parameters that allow constructing datasets modeling specific situations. For example, the frequency or the fraction of the independent actions of an actor can be adjusted. This allows us to understand the behavior of transfer entropy in detail. The most important finding is that when comparing actors, the activeness of actors must be considered. The width of bins are parameters that specify how much of the history of an actor is considered. The analysis shows that the bin widths are data-dependent and should be aligned with the average inter-event time of the actors' events.

We investigate three different real-world datasets and apply the insights gained from the synthetic analysis to the empirical analysis. The River dataset contains information about the water level of four different European rivers. We show that the flow direction of rivers can be inferred using transfer entropy. The LFM-1b dataset contains listening events on Last.fm. Our analysis results show that users of different countries and genres influence each other when listening to music. The Twitter dataset comprises of Tweets posted by Russian trolls in 2016. In our analysis, we show that the most influential hashtags are political. Additionally, we examine the importance of bin widths in empirical analysis and conclude that it depends on the aim of the study, whether the choice of bin widths affects the outcome.

Kurzfassung

Immer mehr Menschen verwenden heutzutage soziale Netzwerke wie Twitter, Instagram und Facebook. Daher wird es auch immer wichtiger, ihr Verhalten zu verstehen, wenn sie mit dem sozialen Medium interagieren. Ein Aspekt davon ist die Analyse einflussreicher User und die Konstruktion von Influence-Netzwerken . Ein Influence-Netzwerk ist ein gewichtetes, gerichtetes Netzwerk mit Social Media Usern als Knoten. Wenn ein Link zwischen User A und User B existiert, heißt das, dass A Einfluss auf B hat. Das Gewicht des Links gibt an, wie stark der Einfluss ist. Eine Möglichkeit solche Influence-Netzwerke zu extrahieren ist Transfer Entropy . Transfer Entropy gibt für zwei stochastische Prozesse X und Y an, wie stark die Unsicherheit in Y durch das Wissen über X reduziert wird. Wenn die Reduktion hoch ist, kann das Verhalten von Y gut vorhergesagt werden und X beeinflusst Y.

Diese Arbeit untersucht die Probleme und Vorteile von Transfer Entropy um Influence Networks zu extrahieren. Forscher haben gezeigt, dass Transfer Entropy verwendet werden kann, um Strukturen von Social Media zu analysieren. Allerdings waren empirische Datensätze der Schwerpunkt ihrer Analyse. Synthetische Datensätze wurden kaum behandelt. Wir untersuchen detailliert, wie man Transfer Entropy verwenden kann, um Wissen aus synthetischen Daten zu extrahieren. Synthetische Daten werden durch Poisson-Prozesse generiert. Die Datengenerierung unterstützt einige einstellbare Parameter, die es erlauben, auf bestimmte Situationen zugeschneiderte Datensätze zu generieren. Zum Beispiel kann die Frequenz und der Anteil unabhängiger User Events eingestellt werden. Diese Methode erlaubt es uns, das Verhalten von Transfer Entropy im Detail zu untersuchen. Das wichtigste Resultat der Analyse ist, dass das Aktivitätslevel der Actors eine große Rolle spielt. Die Breite der Bins sind Parameter, welche angeben, wie viel der Geschichte eines Users Einfluss auf das aktuelle Resultat haben soll. Die Analyse zeigt, dass die Breite er Bins von den Daten abhängt und man die durchschnittliche Inter-Event Zeit der User als Anhaltspunkt nehmen sollte.

Wir untersuchen drei verschiedene, empirische Datensätze und wenden die Erkenntnisse der synthetischen Analyse an. Der Fluss-Datensatz enthält Informationen über den Wasserstand von vier verschiedenen europäischen Flüssen. Wir zeigen, dass die Fließrichtung der Flüsse mittels Transfer Entropy bestimmt werden kann. Der LFM-1b-Datensatz enthält Listening Events des Online-Musikarchivs Last.fm. Unsere Analyse zeigt, dass User verschiedener Länder und Genres sich gegenseitig beeinflussen, wenn sie Musik hören. Der Twitter-Datensatz besteht aus Tweets welche von russischen Internet-Trollen gepostet wurden. In unserer Analyse zeigen wir, dass die einflussreichsten Hashtags ein politisches Thema haben. Wir untersuchen, wie wichtig die Breite der Bins in einem empirischen Experiment ist und kommen zu der Erkenntnis, dass es vom Ziel der Studie abhängt, ob die Wahl der Breiten das Endergebnis beeinflusst.

Acknowledgments

First of all and most importantly, I want to thank my supervisor Denis Helic for his excellent guidance and help. His constructive reviews and insightful comments were extremely helpful while working on this thesis.

This thesis marks the end of my second master's at TU Graz. Finishing two master's programs was not a funny, enjoyable afternoon walk in the sun for me. It was hard work and I was pushing myself to the limit constantly. However, considering how much I learned and how many interesting challenges and problems I solved during my studies, I am beyond happy that I made this decision.

Of course this would not have been possible without the support of my parents Erna and Rupert. Thank you for keeping me motivated and understanding my passion for computer science. I also want to thank the rest of my family, including the cat, which reminded me of the important things in life: food and sleep.

I am very grateful for all my friends, in particular for best friend David. Finally, I want thank Alex for so many things he has done for me, for the late-night whiteboard discussions, the beetroot stew, numerous night shifts, computational resources, providing solutions to unsolvable problems and his efforts to keep me positive-minded.

Contents

Chapter 1

Introduction

There is no doubt that social networks are a central component of our everyday lives. A social network consists of nodes and links between nodes. Nodes are actors, and links describe interactions between them. Twitter is a very famous example of social networks, with about 321 million monthly users [46]. Users can follow each other, post tweets, or retweet tweets from other users. Nodes are, for example, users, and two users are linked if one user follows another user. Recently, there has been a considerable amount of literature on the topic of social network analysis. A huge variety of different methods and approaches have been proposed and evaluated.

One important research topic is to identify influential users in social networks. For example, on Twitter, users often do not make the decision of following or tweeting independently from others but are influenced by other users. Influential users are a small user group influencing another, larger user group [80, 81]. Many different methods to detect influential users have been published [14, 22, 49, 60, 64, 67, 68, 74, 83, 87].

One possibility to analyze social networks and identify influencers is information theory, a mathematical field related to probability theory. It provides important concepts including entropy, joint entropy, conditional entropy and mutual information [25]. Entropy refers to the amount of uncertainty in one random variable. Joint entropy refers to the amount of uncertainty in two random variables using their joint probability mass function. Mutual information describes the amount of uncertainty which is taken away from one random variable by knowing another random variable while conditional entropy represents how much uncertainty remains.

Transfer entropy interchangeably referred to as information transfer is an information-theoretic concept that allows analyzing two stochastic processes utilizing entropy [71,80]. It describes how much uncertainty of a stochastic process is reduced by the knowledge of another stochastic process. In social networks, users can be seen as stochastic processes represented by a list of timestamps. For example, on Twitter, each timestamp in the list might be the time where a user posted a tweet. Transfer entropy can be used to analyze how good the behavior of one user, Alice, can be predicted by the action of another

user, Bob. The prediction is good if the uncertainty about Alice can be reduced because the behavior of Bob is known. It is an asymmetric measure, *i.e.*, Bob might add more knowledge to Alice's process than vice versa.

In social networks, influencers can be identified using transfer entropy [80]. Influencers are those nodes that transfer the most information, *i.e.*, make other nodes predictable. The main advantage of the approach is that it is model-free. Most other methods for influencer detection require causal knowledge, for example, whether Alice retweets a tweet of Bob. However, very often, this information is not available. The main disadvantage of the approach is that it requires more data than other methods to be able to provide meaningful insights.

Another well-studied research topic in the area of social networks is the generation of user data to model user behavior in social networks. The central question about these studies is how users behave when they generate content [89]. Researchers identified a few central characteristics of user behavior. First, users show a strong daily and weekly pattern of activity [41]. Second, users tend to act not only once but several times in a short time frame [15, 35]. For example, when a user is in the process of writing and answering emails, he or she does not only answer one email but tends to answer all unread emails. A synthetic model can simplify the analysis of user behavior. The model reflects true circumstances and provides tuneable parameters, for example, how often a user answers emails per day. Numerous methods how to solve this problem have been published [28,35,54,79,89]. The simplest method to generate user data is a Poisson point process, which will result in a list of timestamps. The number of timestamps is Poisson distributed and depends on the frequency λ. Homogeneous Poisson point processes use a constant λ, while in non-homogeneous Poisson point processes λ depends on the time t, *i.e.*, is a function $\lambda(t)$.

In this thesis, we use Poisson point processes to generate synthetic data using two different models. We evaluate transfer entropy based on this data in situations where two users exist. Furthermore, it is applied to situations where multiple users interact in a network, which results in an influence network. The insights gained from the synthetic analysis are then applied to three real-world scenarios.

1.1 Motivation

Influence networks are weighted, directed networks representing the influence between two nodes. In many areas, it is crucial to know who influential users are. For example, in marketing, it is important to create efficient advertising concepts. These concepts often involve the use of influential persons in social media, and therefore, it is very important to identify influentials. Innovation science is another research area where it is essential to know influential persons. Whenever a company publishes a new product, it enters the so-called innovation adoption life cycle [67]. This life cycle comprises five stages and describes customers using the product in each stage. In the beginning, there are innovators and early adopters, the ones who use the product first. It is very useful for a

company to know how much influence these users have on other potential users.

In various fields, it is crucial not only to identify influentials but also to have a bigger picture of how influence flows in a network of multiple actors. For example, during elections, a party wants to know about influential persons and how the influence of these persons can reach potential voters. Advertisement companies want to know how influence spreads in their customer network such that they can create tailored advertisements.

In this thesis, we want to test and evaluate the concept of transfer entropy to identify influentials and extract influencer networks. Transfer entropy is a concept that has mainly been used in neuroscience to investigate the interaction between neurons in the human brain. There have also been several studies of transfer entropy in social networks [17,18,80]. These studies focus on real-world data only. Only Ver Steeg et al. [80] provide a small analysis of the concept using synthetic data. One of the goals of this thesis is to explore in detail how transfer entropy behaves in different situations, including very active or very inactive actors or actors with a very strong or very weak influence. This is only possible in a lab environment that allows tuning the most important parameters to create data which accurately represents a specific situation.

To evaluate the insights we gain from the synthetic analysis, we also execute a study based on real-world data. The first dataset includes water level measurements of rivers in Austria and Germany. We show that it is possible to find out in which direction rivers flow using transfer entropy. While this is only to be seen as a proof-of-concept, the second analysis provides insights into the influence among users on Last.fm. Finally, we work with a dataset consisting of tweets from Russian trolls. This analysis aims to gain more insights into the troll's behaviors according to the hashtags they use.

1.2 Outline

This thesis is structured as follows: Chapter 2 provides detailed background information on networks and graphs and discusses the most fundamental quantities of information theory and the concept of transfer entropy. Additionally, we give an overview of solutions existing in literature which deal with the problem of identifying influencers and modeling user activity. In Chapter 3, we examine the usage of transfer entropy to infer causal relations in synthetic data. In Chapter 4, we conduct experiments to study three real-world datasets. Finally, Chapter 5 concludes and summarizes our work.

Chapter 2

Background

In this chapter, we give an overview of background information and essential findings in recent literature related to the topic of the thesis. First, we present important network- and graph-related definitions. Second, we quickly introduce the topic of information theory. In the third section, we present the concept of transfer entropy in detail. The final sections suggest how to identify influential users and how to model user activity in social networks.

2.1 Network Basics

A network is a set of objects which are connected by links [32, 57]. What exactly is presented by an object or a link depends entirely on the application and is very flexible. For example, every person is part of a huge social network. The objects could be friends and family, and the links could be interactions with them. In the World Wide Web, nodes are web sites that are connected by hyperlinks [43]. Networks are very similar to the concept of a graph, and in scientific literature, both terms are used interchangeably [15], as it is also done in this thesis. In a network, nodes are connected by links, while in a graph, vertices are connected by edges. However, there is no difference except that the term *network* is used to describe real systems like the World Wide Web. The term *graph* is used in mathematical literature, for example, when talking about a metabolic graph [15, 57].

A graph $G = (V, E)$ consists of a set of vertices (nodes) V which are connected by a set of edges (links) E where $E \subset [V]^2$ [31]. $|V|$ denotes the number of vertices in the graph. $|E|$ denotes the number of edges in the graph. There are several possibilities to present graphs mathematically. Link lists or edge lists enumerate every single link in the network [43]. A link between nodes v and w can be written as (v, w). A link list is a set of single links $L = \{(v, w)|v, w \in V\}$. Link lists are a very simple form of representing networks, but the disadvantage is that it is not beneficial for mathematical computations. It is costly to check whether a link exists because the whole link list

has to be searched, which might contain up to $|V|^2$ elements. Another possibility are adjacency lists [38]. For each vertex v, the adjacency list enumerates all neighbors of v: $L_A(v) = \{w|w \in V\}$. In the worst case, $|V|$ elements have to be searched in order to find out if there is a link between two nodes. It is also not suited for mathematical applications. The representation of the network through an adjacency matrix is often the most convenient way, especially in mathematical settings. The adjacency matrix A of a graph is a $|V| \times |V|$ matrix. The matrix elements $A_{ij} \in A$ are such that:

$$A_{ij} = \begin{cases} 1 & \text{if } (i,j) \in E \text{ (there exists a link between } i \text{ and } j) \\ 0 & \text{otherwise} \end{cases} \tag{2.1}$$

The advantage of adjacency matrices is that it is possible to check in constant time whether a link exists or not. Adjacency matrices can also represent multigraphs, but hypergraphs cannot be represented easily. A multigraph is a graph where two nodes are connected by an arbitrary number of edges [31]. In such graphs, the number of connections often adds additional information. For example, consider a network of employees in a company where each link represents an email communication. The number of links tells how many emails were sent from one employee to another. A_{ij} would then be the number of emails sent. A hypergraph is a graph where an edge connects two or more vertices [31]. In the employee network, an edge could represent a project team. Self-edges are also possible in networks and can be easily represented by adjacency matrices. A self-edge or loop is an edge between a vertex and itself. In the employee example, this means that an employee is a monologizing via email.

2.1.1 Types of networks

The simplest type of a network is a set of nodes connected by links. Sometimes it might be necessary to add more semantics to networks. One way to do this is to add direction to the edges of a graph. A network in which each link has a direction is called *directed* network [31, 43, 57]. Every edge is assigned a source vertex and a destination vertex. An undirected network is a network in which an edge is just a connection between two vertices, without any direction. In the network connecting employees of a company, a directed edge that originates at employee Alice and goes to employee Bob means that Alice sent an email to Bob. In the World Wide Web, a directed link from one web page to another means that the web page has a hyperlink to the other. Another example is a network where the nodes are airports, and the links are flights between those airports. An edge from John F. Kennedy International Airport to London Heathrow means that there is a flight from New York to London, but not the other direction.

The adjacency matrices of directed and undirected networks are different. Adjacency matrices of undirected networks without self-links are symmetric. If there is a link between i and j, there is also a link between j and i. However, adjacency matrices of directed networks are not necessarily symmetric. If there is a link from i to j, $A_{ij} = 1$ but A_{ji} is not automatically 1 [43, 58]. Directed networks can be viewed as undirected

networks in which each undirected edge between two vertices has been replaced by two directed ones.

Link weights or strengths allow to make further annotations to the network. Without weights, links can either be there or not (0 or 1). By adding weights as real numbers to edges, links can be weak or strong according to the assigned weight (weight: $E(G) \to \mathbb{R}$) [43, 58]. Networks with weighted edges are called *weighted* networks. In the employee network, weights are, for example, the number of emails sent per communication. In the airport network, it is the number of passengers transported per year on this flight route. Weighted networks can also be represented by the adjacency matrix.

$$A_{ij} = \begin{cases} w_{ij} & \text{if } (i,j) \in E \text{ (there exists a link between } i \text{ and } j) \\ 0 & \text{otherwise} \end{cases} \tag{2.2}$$

The entries of a weighted adjacency matrix now represent the link weight between two nodes i and j. The weights are usually positive numbers, and zero means that the strength is zero, *i.e.*, there is no link [58]. However, theoretically, there is no reason why edge weights cannot be negative. For example, in social networks, positive edge weights can represent friendship, and negative edge weights can represent antipathy and dislike.

Of course, directed, undirected, weighted, and unweighted networks can be mixed arbitrarily. There exist directed-weighted, directed-unweighted, undirected-weighted, and undirected-unweighted networks.

2.1.2 Examples of real-world network

In recent years, networks have been subject to extensive study for many researchers. Very often, real-world networks are studied. Watts and Strogatz [82], as well as Newman [57], suggest four different categories of real-world networks: social networks, information networks, technological networks, and biological networks.

The study of social networks has been part of sociologic science for decades. Social networks are networks in which nodes are people or groups of people. The edges between the nodes are social contacts or interactions between them [57, 58]. Sociologists often call nodes *actors* and edges *ties*. Social interactions can be friendship, meetings, or messages being sent. Researchers have conducted numerous studies about social networks, for example, to investigate friendship patterns between individuals. A very famous example is Zachary's karate club [88], a study that has been conducted about social interactions in a university karate club. The study shows that social networks often contain a form of community structure. A community is a subset of vertices of the graph which are densely connected with each other, but connections between communities are less dense [37]. Communities in social networks can represent social groups according to some common criteria. In Zachary's karate club, there is a link between two members in case the two members were friends outside the karate club. One day, a karate teacher and the club president had an argument that spread out to all the other members of the club. The club split into two communities. One formed around the karate teacher and one around

the president. However, the study about the Karate club was partially subjective because members had to estimate individually if there is friendship between them or not. Many studies suffer from the problem of subjectivity [57]. Today, the usage of the Internet offers new methods to overcome this problem. For example, the microblogging service Twitter allows users to interact and share thoughts in small messages, so-called tweets. Twitter offers a public API that allows researchers to access massive amounts of data. One can define an interaction between two users as a retweet, for example. This is strictly recorded by Twitter, and therefore, the problem of subjectivity is eliminated. Influence networks are a special case of social networks. They are represented as directed and very often weighted networks. The directed links between objects show that one node influences another. The weight indicates how strong the influence is.

The second category is information networks or knowledge networks. Information networks consist of nodes, which are pieces of data or knowledge that are linked together [57, 58]. Very famous examples are citation networks. In citation networks, nodes are academic papers. Two nodes are linked if one paper cites another paper [33, 58]. Another notable example is the World Wide Web. In general, information networks are man-made, but they are thought to have social structures, too [84].

Technological networks are very similar to information networks, but their primary goal is to distribute a certain resource [58]. The Internet is the most important example of technological networks. It consists of routers, computers, and many other devices that represent nodes. The connections are established wireless or via cable. The goal of the Internet is to distribute information to the nodes. Another example is the electric power grid in a particular country. Nodes would be households, electric power distribution stations, and power plants in general. They are connected by power supply lines.

The last category is biological networks. Their primary goal is to represent biological systems and interactions in these systems in a convenient way [58]. What exactly is represented by nodes and links depends entirely on the biological system which is represented. Protein-protein interaction networks can be used to construct mathematical models of the physical contacts between proteins in a cell. Nodes represent proteins, and edges represent interactions between those proteins. Edges are usually undirected and weighted [65]. Food webs are another example of biological networks that represent the interconnection of food chains. Nodes can be animals and plants. A directed edge between two nodes means that the source node preys on the destination node. For example, given the nodes *pelican* and *fish*, there exists an edge from *pelican* to *fish* because pelicans eat fish. Furthermore, the human brain is a biological system that researchers have been trying to model for the last decades using neural networks. In a neural network, neurons, cells that take an input and generate a specific output, are wired together [58].

2.2 Fundamental quantities of information theory

Information theory is a mathematical field that deals with communication and the storage and transmission of information [25]. The central quantities of information theory are

entropy, relative entropy, and mutual information. Their definitions are based on probability theory. Information theory is widely applied in many different areas nowadays, for example, economics or physics.

2.2.1 Entropy

In computer science, measuring information is very important. Entropy is a quantity that is very often used to describe the amount of information a random variable contains. In other words, entropy measures the uncertainty of a random variable [25]. Let X be a discrete random variable with respective probability mass function $P(X = x) = p(x), x \in \mathcal{X}$ where $\mathcal{X}$ denotes the alphabet. The entropy $H(X)$ is defined by [73]

$$H(X) = -\sum_{x \in \mathcal{X}} p(x) \log_2 p(x) \tag{2.3}$$

The unit for entropy is bits. Intuitively, entropy describes the expected surprise of a random variable. The less probability is assigned to an event, the more surprising its occurrence, and the higher entropy (uncertainty) is. From another point of view, entropy is the amount of missing information and the number of questions which are necessary to identify the information.

For example, assume that Alice thinks about a number between 0 and 3, and Bob has to guess the number. All numbers between 0 and 3 can be represented by 2 bits. Bob first asks about the first bit and then about the second bit. He needs two questions to get the correct number. Therefore, entropy is computed by:

$$H(X) = -\left(\frac{1}{4}\log_2\frac{1}{4} + \frac{1}{4}\log_2\frac{1}{4} + \frac{1}{4}\log_2\frac{1}{4} + \frac{1}{4}\log_2\frac{1}{4}\right) = 2 \text{ bits}$$

Entropy has several important properties [25]:

1. $H(X) \geqslant 0$: Entropy is always positive.
2. $\max H(X)$ for $p(x) = (\frac{1}{N}, ...\frac{1}{N})$: Entropy of a random variable X with N possible outcomes reaches its maximum if X is uniformly distributed.
3. $\min H(X)$ for $p(x) = (1, 0, 0, ...)$: Entropy of a random variable X reaches its minimum if $P(X = k) = 1$ for some $k \in \mathcal{X}$.

2.2.2 Joint entropy

Joint entropy $H(X, Y)$ of two discrete random variables X and Y is defined using the joint probability mass function $P(X = x, Y = y) = p(x, y), x \in \mathcal{X}, y \in \mathcal{Y}$ for alphabetes $\mathcal{X}$ and $\mathcal{Y}$ [25]:

$$H(X, Y) = -\sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log_2 p(x, y) \tag{2.4}$$

Intuitively, it is the uncertainty associated with two random variables X and Y and means the amount of questions which must be asked to find out about both X and Y. If X and Y are independent random variables, then $H(X,Y) = H(X) + H(Y)$.

For example, if Alice thinks about two numbers between 0 and 3, Bob needs two questions for the first number and two questions for the second number, in total four questions, to guess the correct one.

2.2.3 Conditional entropy

Conditional entropy $H(Y|X)$ of two discrete random variables X and Y is defined using the conditional probability $P(Y = y|X = x) = p(y|x), x \in \mathcal{X}, y \in \mathcal{Y}$ for alphabets $\mathcal{X}$ and $\mathcal{Y}$ [25]:

$$H(Y|X) = -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(y|x)\log_2 p(y|x) \tag{2.5}$$

Conditional entropy describes the amount of uncertainty still associated with Y after subtracting the uncertainty removed by X.

For example, assume Bob wants to know whether Alice likes computer games [42]. In general, the probability that Math and Computer Science students like computer games is 0.9, but the probability that History (H) students like computer games is only 0.25. Let X be a discrete random variable representing the study of Alice. Let Y be a discrete random variable telling about whether Alice likes computer games. If Bob knows what field of study Alice chose, the uncertainty about whether she likes computer games is lower. If Bob does not know the field of study, the overall uncertainty about the question is higher.

Conditional entropy $H(Y|X)$ is zero if and only if the value of Y is known by knowing X [25]. In case X and Y are independent random variables, X cannot reduce the uncertainty about Y, and therefore, the knowledge about X is useless, *i.e.*, $H(Y|X) = H(Y)$. Furthermore, $H(Y|X) \leqslant H(X)$ because knowing X can only reduce the uncertainty about Y. In the worst case, the reduction is 0.

There is one important relation between conditional entropy and joint entropy which is called the *chain rule*:

$$\begin{aligned} &H(X,Y) = H(X) + H(Y|X) \Leftrightarrow \\ &H(Y|X) = H(X,Y) - H(X) \end{aligned} \tag{2.6}$$

It allows expressing conditional entropy in terms of joint entropy and vice versa.

2.2.4 Relative entropy

Relative entropy, also known as Kullback-Leibler divergence or Kullback-Leibler distance, is defined for discrete probability distributions $p(x), q(x)$ by [25, 52]:

$$\begin{aligned} D(p||q) &= \sum_{x \in \mathcal{X}} p(x) \log_2 \frac{p(x)}{q(x)} \\ &= \mathbb{E}\left(\log_2 \frac{p(x)}{q(x)} \right) \end{aligned} \tag{2.7}$$

Relative entropy is the expected value of the logarithmic difference between two probability distributions. Therefore, it can be seen as a measure of how different probability distributions are. Computing the divergence is especially useful when approximating one distribution with another and wanting to know how much information is lost because of the approximation. However, the term *distance* is misleading since it is not symmetric.

2.2.5 Mutual information

The definition of mutual information is based on the definition of relative entropy. For two discrete random variables X and Y with probability distributions $p(x), p(y), x \in \mathcal{X}$ and $y \in \mathcal{Y}$ as well as the joint probability mass function $p(x, y)$, mutual information $I(X;Y)$ is [25]:

$$\begin{aligned} I(X;Y) &= D(p(x,y)||p(x)p(y)) \\ &= \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x,y) \log_2 \frac{p(x,y)}{p(x)p(y)} \end{aligned} \tag{2.8}$$

While mutual information shows how much uncertainty is taken away from Y by knowing X, conditional entropy describes how much uncertainty remains. Therefore, it can also be computed as the difference between entropy and conditional entropy.

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) \\ &= H(Y) - H(Y|X) \end{aligned} \tag{2.9}$$

Mutual information is symmetric. If X and Y are independent random variables, $I(X;Y) = 0$ because then $H(X|Y) = H(X)$ and $H(Y|X) = H(Y)$. It can also be used to measure how independent two random variables are [86]. Figure 2.1 shows the relationship between all fundamental quantities of information theory, entropy, mutual information, conditional entropy, joint entropy.

2.3 Transfer entropy

Transfer entropy, also referred to as information transfer, is used to analyze two stochastic processes, X and Y, and is written as $TE_{X \to Y}$ or respectively, $TE_{Y \to X}$. Transfer entropy describes how much uncertainty of the stochastic process Y is reduced by the knowledge of stochastic process X [44, 71, 80].

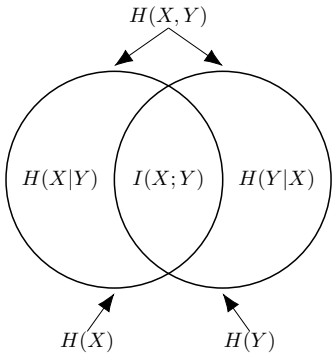


Figure 2.1: Relationship between mutual information $I(X;Y)$, conditonal entropies $H(X|Y), H(Y|X)$, entropies $H(X), H(Y)$ and joint entropy $H(X,Y)$

2.3.1 Granger causality

Transfer entropy is "a nonlinear generalization of Granger causality" [16, 80]. Granger causality is used to test if two random variables are connected by a causal link. Usually, these two random variables are part of a time series. According to the definition, a random variable X Granger-causes a random variable Y if, given the history of X and Y, the history of X helps to predict the future of Y. Assume the future of Y, y_t should be predicted by a linear prediction model:

$$y_t = a_1 y_{t-1} + a_2 y_{t-2} + ... + a_n y_{t-n} + ... + e_y \tag{2.10}$$

e_y is the error of prediction. If X should also be considered for the prediction, one could write:

$$\begin{aligned} y_t = a_1 y_{t-1} + a_2 y_{t-2} + ... + a_n y_{t-n} + ...+ \\ b_1 x_{t-1} + b_2 x_{t-2} + ... + b_n x_{t-n} + ... + e_{y|x} \end{aligned} \tag{2.11}$$

$e_{y|x}$ is the error of the second prediction. X is Granger-causal to Y if the variance of the error of the second model is smaller than the variance of the error of the first model, *i.e.*

$$\mathrm{var}(e_{y|x}) < \mathrm{var}(e_y) \tag{2.12}$$

A smaller error variance means that X helps to predict Y and therefore there is a causal relationship between them [29].

2.3.2 Definition of transfer entropy

Similar to Granger causality, transfer entropy works with lists of timestamps. More specifically, for the stochastic process X, the list of timestamps (the history of X), is denoted by

$$S_X = \{t_j : 0 < t_1 < t_2\} \tag{2.13}$$

However, transfer entropy is a probabilistic measure and the list of timestamps is not. Timestamps can be converted into probability distributions via the transformation into binned random variables [80]. A binned random variable is assigned to a specific time frame. If an event happens in that time frame, the binned random variable is 1. Otherwise, it is 0.

$$B_X(a,b) = \begin{cases} 1 & \text{if } \exists t_j \in S_X \cap (a,b] \\ 0 & \text{otherwise} \end{cases} \tag{2.14}$$

We refer to the binned random variable at time t for the history of a stochastic process X as X_t. The distance between a and b is called a bin width, $\delta = b - a$. Usually, one does not only want to compute one bin but k bins with bin widths $\delta_0, \delta_1, \delta_2, ...\delta_k \in \mathbb{R}$. This results in the joint probability distribution describing a sequence of adjacent bins [80]:

$$P\left(B_X(t, t-\delta_0) = X_t, B_X(t-\delta_0, t-\delta_0-\delta_1) = X_{t-1}, ...\right) \tag{2.15}$$

For reasons of simplicity, let $X_t^{t-k} = \{X_t, ...X_{t-k}\}$ and $Y_t^{t-k} = \{Y_t, ...Y_{t-k}\}$. We obtain a shorter notation for the joint probability function $P(X_t^{t-k}, Y_t^{t-k})$.

Transfer entropy works with these probability distributions. One can rewrite the formula using the chain rule introduced in Equation 2.6.

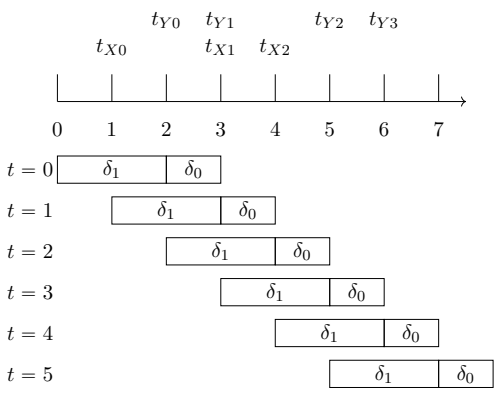
$$\begin{aligned} T_{X\to Y} &= H(Y_t|Y_{t-1}^{t-k}) - H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k}) \\ &\overset{\text{Chain rule (2.6)}}{=} H(Y_t, Y_{t-1}^{t-k}) - H(Y_t) - H(Y_t, Y_{t-1}^{t-k}, X_{t-1}^{t-k}) + H(Y_t) \\ &= H(Y_t, Y_{t-1}^{t-k}) - H(Y_t, Y_{t-1}^{t-k}, X_{t-1}^{t-k}) \end{aligned} \tag{2.16}$$

Intuitively, it can be described as the reduction of uncertainty of random variable Y_t given the history of X. The first term describes the uncertainty about Y_t given the history of Y. The second term describes the uncertainty about Y_t given the history of Y *and* the history of X. The difference between those two describes the reduction of uncertainty.

For example, consider the case where X and Y represent Twitter users and their histories S_X and S_Y describe when they posted a tweet. X can only reduce the certainty of Y if there is information flow from X to Y. For example, if X tweets about climate change and Y tweets about climate change minutes later because Y read the post of X, there is information that flows from X to Y. Instead of information flow, the term information transfer is also used interchangeably for transfer entropy. Another possibility is to view transfer entropy as the amount of information about Y that can be explained by the history of X but cannot be explained by the history of Y alone [80].

Transfer entropy is asymmetric, *i.e.*, $TE_{X\to Y} \neq TE_{Y\to X}$. Intuitively, there might be information flow from user X to user Y but not vice versa.

Figure 2.2 shows an example of the computation of transfer entropy for two stochastic processes X and Y and their respective list of timestamps. The right shows the list of timestamps S_X and S_Y, the values of the binned random variables for each step in time, and the probabilities which are computed from the binning process. Marginalizing these



$S_Y = \{2, 3, 5, 6\}, S_X = \{1, 3, 4\}, k = 1$

t	Y_t	Y_{t-1}	X_t	X_{t-1}
0	1	0	0	1
1	1	1	1	1
2	0	1	1	1
3	1	1	0	1
4	1	1	0	1
5	0	1	0	0

$P(Y_t = 1, Y_{t-1} = 0, X_t = 0, X_{t-1} = 1) = 1/6$
$P(Y_t = 1, Y_{t-1} = 1, X_t = 1, X_{t-1} = 1) = 1/6$
$P(Y_t = 0, Y_{t-1} = 1, X_t = 1, X_{t-1} = 1) = 1/6$
$P(Y_t = 1, Y_{t-1} = 1, X_t = 0, X_{t-1} = 1) = 2/6$
$P(Y_t = 0, Y_{t-1} = 1, X_t = 0, X_{t-1} = 0) = 1/6$
$H(Y_t|Y_{t-1}) = 0.8091,\ H(Y_t|Y_{t-1}, X_{t-1}) = 0.5408$
$\rightarrow TE_{X\rightarrow Y} = 0.2683$
$H(X_t|X_{t-1}) = 0.8091,\ H(X_t|X_{t-1}, Y_{t-1}) = 0.6666$
$TE_{Y\rightarrow X} = 0.1425$

Figure 2.2: Example computation of transfer entropy for two stochastic processes X and Y

probabilities gives the respective distributions to compute the entropies which are needed to compute transfer entropy.

2.3.3 Normalized transfer entropy

The total value of transfer entropy can be difficult to interpret sometimes. Especially when comparing different scenarios and situations which use different parameters or configurations, it is hard to do evaluations based on absolute values for transfer entropy. Therefore, we define normalized transfer entropy [40] by

$$\begin{aligned} \hat{TE}_{X\rightarrow Y} &= \frac{TE_{X\rightarrow Y}}{H(Y_t|Y_{t-1}^{t-k})} \\ &= \frac{H(Y_t|Y_{t-1}^{t-k}) - H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k})}{H(Y_t|Y_{t-1}^{t-k})} \end{aligned} \tag{2.17}$$

The normalized transfer entropy will map the value of transfer entropy to the range [0,1], and therefore, $\hat{TE}_{X\rightarrow Y}$ can be seen as the amount of information flow in percent. If X is a perfect predictor for Y then Y is completely determined by X, *i.e.*, Y strictly follows X. $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k})$ becomes zero and we obtain 1 as the total normalized transfer entropy (Case 1 in Equation 2.18). If X and Y are completely independet from each other, knowing X does not change the entropy of Y and $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k}) = H(Y_t|Y_{t-1}^{t-k})$. The total normalized transfer entropy is then 0 (Case 2 in Equation 2.19).

Case 1: Y fully depends on X

$$H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k}) = 0$$
$$\Rightarrow \hat{TE}_{X\to Y} = \frac{H(Y_t|Y_{t-1}^{t-k}) - 0}{H(Y_t|Y_{t-1}^{t-k})} = 1 \quad (2.18)$$

Case 2: Y fully independent from X

$$H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k}) = H(Y_t|Y_{t-1}^{t-k})$$
$$\Rightarrow \hat{TE}_{X\to Y} = \frac{H(Y_t|Y_{t-1}^{t-k}) - H(Y_t|Y_{t-1}^{t-k})}{H(Y_t|Y_{t-1}^{t-k})} = 0 \quad (2.19)$$

However, this solution has some issues. For example, if $H(Y_t|Y_{t-1}^{t-k})$ is very small and at the same time Y fully depends on X, so $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k}) = 0$, $\hat{TE}_{X\to Y}$ will be 1. The smaller $H(Y_t|Y_{t-1}^{t-k})$ is, the less information is actually transferred but still the normalized transfer entropy is 1. This problem actually occurs whenever $H(Y_t|Y_{t-1}^{t-k})$ is very small and $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k})$ is comparably big.

Another possibility is to choose $\log k$ as the normalization factor.

$$\hat{TE}_{X\to Y} = \frac{TE_{X\to Y}}{\log k} \quad (2.20)$$

This is based on the observation that $0 \leqslant TE_{X\to Y} \leqslant \log k$ because entropy reaches the maximum value in case the probability distribution is uniform.

$$\begin{aligned} 0 &\leqslant TE_{X\to Y} \\ &= H(Y_t|Y_{t-1}^{t-k}) - H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k}) \\ &\leqslant H(Y_t|Y_{t-1}^{t-k}) \\ &\leqslant H(Y_t) \\ &\log k \end{aligned} \quad (2.21)$$

This solution manages to overcome the problem described above because $\hat{TE}_{X\to Y}$ is big in case $H(Y_t|Y_{t-1}^{t-k})$ is big and small otherwise because the scale factor is constant.

2.4 Identification of Influencers in Social Networks

In social networks, actors interact with each other. For example, on Twitter, users can follow each other, retweet tweets, or react to tweets. In many cases, users do not make the decision of following or tweeting independently from others but are influenced by other users. One important research topic is to find and identify influentials. Influentials are a small user group that influences the behavior of another larger user group [80]. In many research areas, it is essential to identify influentials because they have a big effect on the public opinion [81].

In 1955, Katz and Lazarsfeld introduced the theory of the two-step flow of communication [47]. The theory says that people do not form their opinion directly based on mass media, but rather based on influential personalities who interpret mass media and put the information into context. The theory is called a two-step flow because, in the first step, information flows from the mass media to the influentials, and in the second step, information flows from the influentials to the people. Back in 1955, a good example of the theory would be the weather forecast. Meteorologists publish the weather forecast, but people do not read the forecast directly but listen to the weatherman on the radio. Examples where the theory applies can still be found today in social networks like Instagram. For example, on Instagram, brands are selling and advertising face care products. Those brands contact influential persons on Instagram to advertise their products and show them to their followers because it turns out that this way of advertising is much more effective. Regular users react more positive on influencer advertisements than on brands' advertisements.

In the decades after Katz and Lazarsfeld published the theory, the idea of influentials became a central element of research in various areas, including marketing, communication science, and innovation science [81]. For example, if a company creates innovation and invents a new product, influential users need to be identified because the whole mass of other potential new users will follow them [56].

In the area of computer science, researchers have identified several effective methods to detect influential actors in a social network. In 2010, Weng et al. [83] showed how to use PageRank to identify influential twitter users. Based on that, many other influencer detection methods using PageRank have been published [49, 51, 87]. Influentials can also be identified based on the number of retweets, followers, and mentions [22], as well as the number of new followers over a specific time frame in the past [64]. Other ideas include the construction of follower trees and information cascades [14]. A similar approach was published by Silva et al. [74], combining information diffusion and PageRank. Romero et al. [68] show how to detect the amount of passive information consumed by users and find out about influencers using the influence-passivity score. In this thesis, we identify influencers using a method published based on the concept of transfer entropy [80]. Most of the studies in this area use Twitter as the main resource for their dataset because Twitter offers an easy-to-use API, which allows to construct a large dataset comfortably. However, most of the insights gained apply to any social network.

2.4.1 PageRank

PageRank is used by Google to determine the relevance of their search engine results and was invented by Page et al. [60]. The basic idea of the algorithm is that web pages that have a lot of ingoing links from important web sites are very important themselves. The algorithm starts with each page having the same rank. Then, iterating over a predefined number of steps, each web page updates its rank by summing over the rank of ingoing links. A web page A has an ingoing link if another web page B links to A.

TwitterRank modifies the PageRank algorithm [83]. Instead of pages, the algorithm works with twitter users. The *following* relationship between users represents links. This means that user A links to user B if A is followed by B. The influence of a twitter user is strong if the user is followed by influential users. The influence of a user on a follower depends on the amount of content that the followers receive from the user. However, TwitterRank focuses on certain topics, *i.e.*, identifies topic-specific influencers.

Kwak et al. [49] compare the ranking created by PageRank to a ranking created by the number of followers and the number of retweets. The set of tweets which they use focuses on trending topics of a specific time frame. They conclude that the rankings by PageRank and the number of followers are similar, but the ranking by the number of retweets differs. The reason for this is that Twitter users more often retweet news pages, for example, CNN or The New York Times, than tweets of regular users.

TURank, a method published in 2010 [87], is based on ObjectRank, which is an extension of PageRank. ObjectRank additionally distinguishes different kinds of edges and nodes by applying weights to them. TURank applies ObjectRank to the so-called user-tweet graph. The user-tweet graph comprises users and tweets as objects which are linked by edges of a specific type. In case a user tweets a tweet, there is a *post* edge from the user to tweet. If a user follows another user, there is a *follow* edge. A *retweet* edge is created from a tweet t to the retweet tweet of t. They introduce reverse edges for all of the three types mentioned above, calling them *posted*, *followed* and *retweeted*. The edge weights depend on several different parameters, including the number of outgoing edges and an assignment made by the authors by hand. ObjectRank is applied to this user-tweet graph to obtain the respective ranks. Their experiments show that, for example, the algorithm assigns users with a low number of followers, but a high number of retweeted tweets, a higher rank than users with many followers and a low number of retweeted tweets. TURank was also shown to filter out long conversation tweets between users because they should not affect influence.

2.4.2 Retweets, Indegree and Mentions

Cha et al. [22] shows how to measure influence using retweets, indegree and mentions. Indegree is equivalent to the number of followers a user has. Mentions can be made in a tweet to link to another user. For example, this tweet mentions Barack Obama: `Thinking about the elections. @BarackObama should run for president again.` The authors provide a novel categorization of influence: Indegree influence provides information about the audience size of a specific user by counting followers. A high Retweet influence of a user means that he or she creates tweets that motivate other users to share these contents. Mention influence describes the number of conversations created which want to engage a specific user. The authors of the paper construct the dataset by collecting tweets for the time range of a month and filtering out unsuitable users, for example, private or inactive users. Then, they compute the influence measured of all those users, as described above.

The most influence, according to indegree influence, comes from news sources and celebrities, for example, actors and models. News sources have a high Retweet influence, together with content creation pages, e.g., Tweetmeme. Celebrities have the biggest mention influence because regular users often started conversations with celebrities, for example, in the form of gossip. The authors provide a toplist for users according to influence of all three categories. There are only a few users which appear on all three lists. Combining all three influence measures, they show that users who are often retweeted are also often mentioned in Tweets and vice versa. However, the correlation between the number of followers and the other measures is very weak. The authors examine other measures, for example, the overall number of tweets or the number of users who are followed by a specific user. They do not include them in the analysis because they very often rated bots and spammers as very influential.

Another approach was published by Qasem et al. in 2015 [64]. They examine the usage of the number of new users a potential influencer has interacted with as a measure of influence. The method is based on the thought that the more new users are attracted by a particular user u in a specific time frame, the more influential u must be. The authors do not only apply this analysis to data gathered from Twitter, but also to data from Asterisk. Asterisk is an open-source software project, and they considered the mailing list, which serves an easy communication method among Asterisk developers. If two developers appear in the same mailing thread, they are considered to be connected. In the Twitter dataset, they focus on the retweet-connection.

2.4.3 Information diffusion

The spread of information in social network has recently been a heavily studied research topic. Information diffusion is determined by parameters like the spreading rate and the topology of the social network [50]. The two most widely-used approaches for modeling information diffusion are the Linear Threshold model and the Independent Cascade model [50]. Both models work with a directed graph representing the social network. Nodes in the graph are either active or inactive. Information starts at an initial set of active nodes and spreads in discrete time steps. In each time step, a node can become active or inactive. The information diffusion stops in case no more nodes can be activated.

In the Linear Threshold model, each node in the network is assigned a random threshold. When information reaches a particular node, the node is only activated in case the total amount of incoming information exceeds the previously assigned threshold [21]. In the Independent Cascade model, each node w has a single chance to activate each of its neighbors v_i. A neighbor v_i becomes active in the next time step with a certain probability. Independently of whether w managed to activate v_i, w will never activate v_i later. Information diffusion processes can be used to analyze and identify influential nodes in a network, as it was shown by Bakshy in 2011 [14] and by Silva et al. in 2013 [74].

The study which was done by Bakshy [14] is also based on a Twitter dataset. They track the diffusion of URL posts, which are tweets containing a URL. Tracking starts at a

specific seed node, the user who posted the URL, and follows the path of followers and followers' followers until the information diffusion process terminates. They say that a twitter user A influences his follower B if first A posts the URL and then B. If A is not the only person followed by B who posted the URL, there are several possibilities to deal with the situation. The authors suggest to either choose the user who posted the URL first, the one who posted the URL most recently or to split the influence credit among all users. In the next step, influence trees are constructed for every initial posting of a URL. The final influence score for every seed node is given by the number of users in the influence tree. It is shown that the end result, *i.e.*, which users are the most influential ones, is not determined by the choice of influence assignment in case more users posted the URL. The drawbacks of this approach are that it might also happen that a user posts an URL without being influenced by one of the users he follows. Additionally, the notion of influence might be defined too strict here because they assume it is necessary to retweet an URL such that influence actually exists. In real world, it might often suffice that a user opens the URL in his browser to talk about influence.

ProfileRank was published by Silva et al. [74] in 2013. The algorithm tries to identify influential users by applying ideas of both information diffusion and PageRank. ProfileRank is based on the assumption that "relevant content is created and propagated by influential users and influential users create relevant content" [74]. The authors used Twitter as the example platform for their experiments. In the first step, a random surfer is started to navigate through twitter randomly. The random surfer clicks on random tweets, which leads to a user profile. On the user profile, the surfer again clicks on a tweet and so on. User influence is computed by counting the number of visits of a specific user profile. It is mentioned that the significant advantage of PageRank is its independence from the meaning of the tweet. It is easy to identify the meaning of a tweet containing only text, but it is harder to deal with image or video content. Since PageRank does not rely on such information, it is applicable in those situations.

2.4.4 Influence-passivity score

A different approach focuses on the passivity of users in social networks [68]. The main assumption is that the majority of users does not actively publish information but is more of a passive information consumer. For example, on Twitter, most users do not actively tweet and retweet information but rather only read through their feed without acting. They define influential users as users who act more actively than passively and additionally have a broad audience, *i.e.*, a large number of followers. How passive a user is depends on how easy it is for other users to influence him. In order to determine how influential a specific user is, the method considers the collective passivity of all followers.

In detail, the algorithm operates on a weighted directed graph with the users as nodes. There is an edge between two users if user A follows user B. The weights are determined by the amount of influence A effectively has on B in relation to the amount of influence A tried to have on B, the so-called relative influence. Relative passivity can be defined accordingly and by normalizing both measures using the total incoming influence/passivity,

the so-called acceptance and rejection rates can be computed.

Their findings show that the Influence-passivity score method can be used to successfully identify the most influential users, mainly news sites and bloggers but also the most passive users, which were exclusively bots. A significant advantage of the algorithm is that it applies to other platforms than Twitter easily.

2.4.5 Transfer Entropy

In 2007, researchers showed that transfer entropy can be used to investigate the relationship between neurons in the human brain [40]. In 2012, Ver Steeg et al. [80] showed that the concept could also be used in social networks to find out about influential nodes. The big advantage is that the approach does not rely on information about causal relations. Almost all methods which were discussed so far required to know whether user A follows user B or user A retweeted tweets posted by user B. Transfer entropy does not require this information to characterize influential nodes, but solely needs to know when certain events happened, e.g., when users A and B tweet something about a particular topic. Therefore, the approach is called model-free by the authors.

They use a Twitter dataset to evaluate their methods and test if transfer entropy can be used in order to infer information transfer from the timestamps of 3.5 million tweets of 800.000 users. They exclude users who did not reach a certain activity level, *i.e.*, users who did not at least make 10 tweets in the observation period. The results show that the majority of accounts, which were the source of the most information transfer, are promotional accounts and bots. In general, the authors state that the advantage of the method is that it is model-free, but the disadvantage is that it requires more data than other approaches. They also suggest to pick the bin resolution based on a user's activity to avoid coarse-grained information for very active users.

2.5 Modeling User Activity in Social Media

Social media platforms are very complex systems. Understanding those systems is sometimes not easy. Very often it helps to narrow down the complexity by building a simple model from the complex system. Since social networks rely on user-generated content, the central question of such models is how users behave [89]. For example, researchers analyzed the behavior of users writing emails. They found that the times when users write emails are clustered [89]. This means that through a day, a user has long, inactive periods where no emails are written. When an active period occurs, the user often writes not only one email but several emails at once. However, several other characteristics of user behavior have already been identified.

Over the last years, researchers have suggested numerous stochastic models to cope with these characteristics. In the following, we want to discuss some of them. The simplest way to model user behavior is to use a Poisson point process [15, 79], which we also use in this thesis. Malmgren et al. [54] propose the Cascading Non-homogeneous Poisson point

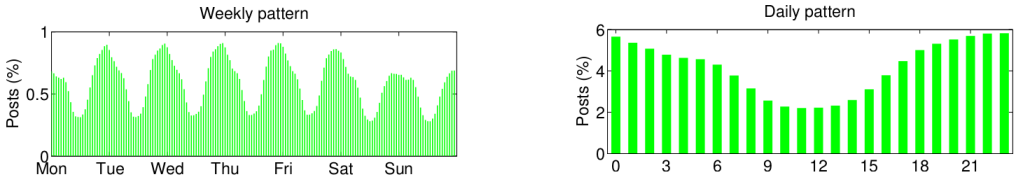
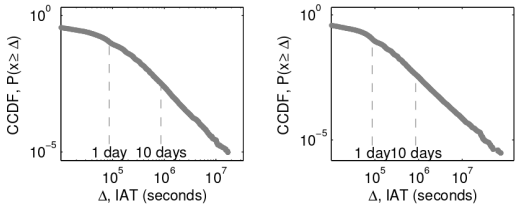


Figure 2.3: User posting behavior in a social network over a week and day by L. Guo, E. Tan, S. Chen, X. Zhang, and Y. E. Zhao in *Analyzing patterns of user content generation in online social networks*, 2009 [41]



(a) Reddit (b) Twitter

Figure 2.4: Inter-arrival times of postings on Reddit and Twitter follows the power law. Image by A. Ferraz Costa, Y. Yamaguchi, A. Juci Machado Traina, C. Traina Jr, and C. Faloutsos in *Mining and modeling temporal activity in social media*, 2015 [35]

process (CNPP) model, which uses a special form of Poisson point processes. Zhang et al. [89] describe a new class of point process models, Interpretable Point Processes. Furthermore, in 2013, Vaz de Melo et al. [28] proposed the Self-Feeding Process (SFP), combining non-homogeneous Poisson point processes and power laws. The Rest-Sleep-Comment (RSC) model [35] focuses on the inter-event times of postings on social media and tries to cover all identified characteristics.

2.5.1 Characteristics of user behavior

User activity in social networks is characterized by timestamps in most cases [77]. Those timestamps describe when and how often a user executes a certain action. An action can be leaving a comment, sending an e-mail, retweeting, or following another user. The list of timestamps of user X is denoted as $S_X = \{t_j : 0 < t_1 < t_2 ...\}$. The number of timestamps in a certain time interval is described as the frequency [77].

A study by Guo et al. [41] observes and analyzes users of various social networks. They found out that there is a strong daily and weekly pattern of user posting behavior to social media. For example, users of the social media platform Instagram might check new posts twice a day, in the morning before work and in the evening. Additionally, on Saturdays and Sundays, they might change their pattern and check it at lunchtime, too. Another example using an answer social network, such as Stack Overflow, is given in the study. The pattern of user behavior can be seen in Figure 2.3. It shows that users generate more posts on weekends than on weekdays. The peak time for posting

during a day is 11 pm, the least posts are generated around 6 am. Another approach was introduced by Ferraz et al. [35], which focuses on inter-event times (inter-arrival times) between postings. The inter-event time is the time between two consecutive postings of the same user. They found out that these inter-event times have periodic spikes every 24 hours. This was also shown by the study we discussed above.

The inter-event times follow a heavy-tail distribution, *i.e.*, the power law [15, 35]. This means that most postings are done in small intervals while the minority of postings is done in long intervals, which creates a bursty pattern of events over time. Periods of high activity are followed by periods of resting. An example is shown in Figure 2.4 for the online services Reddit and Twitter. There are various reasons for this behavior, for example, the natural circadian cycle of humans.

2.5.2 Poisson point process models

Although it was shown that the inter-event times of posts in social networks follow the power law, the simplest way to model human activity is a Poisson point process.

Point processes, in general, are used to describe the random scattering of points in any dimension. In most cases, one-dimensional data is generated in the notion of time points [66]. At every step in time, the point process can only take one of two possible values: 1 in case an event occurs at that point in time or 0 otherwise. Point processes can be specified by the distribution of inter-event times. Famous examples of point processes are Poisson point processes and Hawkes processes. Hawkes processes are models describing situations in which the occurrence of an event increases the chance of other events happening immediately after.

Poisson point processes are characterized by the Poisson distribution, a discrete probability distribution. If X is a Poisson random variable and equal to k, the probability is given by:

$$P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!} \tag{2.22}$$

λ is the single Poisson parameter and defines the distribution. For example, X can represent the number of events happening in a specific time frame. The probability that the event occurs k times is given by the equation above. λ describes the frequency in which events happen on average.

Poisson point processes are characterized by the following properties [13, 48]:

1. *Number of points is Poisson distributed:* A Poisson point process is simulated in time. For any bounded time frame in this time, the number of points is Poisson distributed. For example, the number of events happening in the time frame is Poisson distributed.
2. *Independence:* Now, considering two non-overlapping timeframes, the number of points in both time frames are independent random variables.

We distinguish homogeneous or stationary Poisson point processes and non-homogeneous

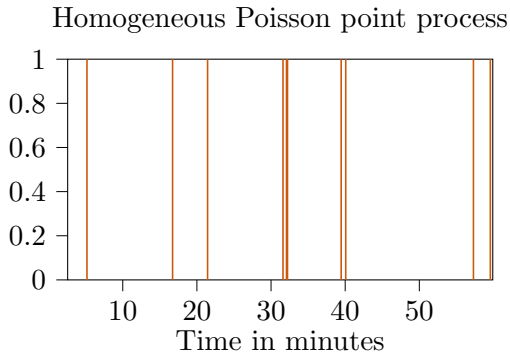


Figure 2.5: Example of points generated by a homogeneous Poisson point processes for $T = 1$ hour and $\lambda = 0.6$

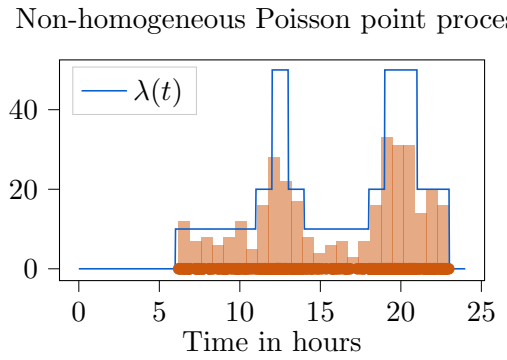


Figure 2.6: Example of points generated by a non-homogeneous Poisson point processes for $T = 1$ day and

$$\lambda(t) = \begin{cases} 0 & 0 \leqslant t < 6, t \geqslant 23 \\ 10 & 6 \leqslant t < 11, 14 \leqslant t < 18 \\ 30 & 11 \leqslant t < 12, 13 \leqslant t < 14, 18 \leqslant t < 19, \\ & 21 \leqslant t < 23 \\ 50 & 12 \leqslant t < 13, 19 \leqslant t < 21 \end{cases}$$

or inhomogeneous Poisson point processes.

Homogeneous Poisson point processes are Poisson point processes having a constant parameter λ. Homogeneous Poisson point processes have a third, additional property [48]:

3. *Memoryless:* The distances of two points in the Poisson point process follow an exponential distribution with parameter λ. This means that the inter-event times are exponentially distributed. Memoryless means that the fact that an event occurred at any time t_1 does not affect the probability that another event occurs at any other time t_2.

An example of such a process would be the customers arriving at a fast-food restaurant, the number of earthquakes in a certain area [19], or visitors of a web site. Examples of points generated with a Poisson point process are given in Figure 2.5.

Non-homogeneous Poisson point processes do not use a constant parameter λ, but a function $\lambda(t)$ describing the intensity at a certain point in time t. This can be useful to describe processes where the amount of events depends on the time. For example, customers arriving at a fast-food restaurant would probably arrive more often around lunch- or dinner-time. A web site of the Austrian government will have more visitors in the morning or in the evening, but almost none during night time. An example of a non-homogeneous Poisson point process is given in Figure 2.6. It shows the model of a restaurant simulated over one day. $\lambda(t)$ gives the average number of customers arriving at different times of the day.

In this thesis, we generate synthetic data for user activity on social media using homogeneous and non-homogeneous Poisson point processes.

2.5.3 Cascading Non-homogeneous Poisson point process (CNPP) model

Malmgren et al. show in [53,54] that Cascading Non-homogeneous Poisson point processes can be used to model email communication. They found out that email communication is periodic and cascading. *Periodic* refers to the recurring peaks in user activity. *Cascading* refers to cascades of activity, which are active periods of varying length in which the user sends emails. They propose the CNPP approach, which combines a homogeneous Poisson point process and a non-homogeneous Poisson point process to model user activity when writing emails. The non-homogeneous process is used to represent periodicity in user behavior using a frequency rate of $\lambda_p(t)$. In order to define $\lambda_p(t)$, they analyze the level of user activeness over a week such that the week can be separated into intervals where each interval has a certain level of activeness associated. Depending on in which period t falls, $\lambda_p(t)$ gives different intensity values. The result of this process is a list of times where an active period occurs. The homogeneous Poisson point process is used to model these active periods (cascades) in which λ_a is chosen constant and represents the average number of emails written in the period. It delivers the actual list of timestamps where emails are written. The authors show that the model can be applied to email communication and suggest that it might also be applicable in the situation of telephone calls or running errands.

2.5.4 Bivariate Point Process models

In 2018, researchers conducted a study which analyzes tweets from Donald Trump, and posts of users on Sina Weibo are analyzed [89]. They find that the interaction of the user with social media happens in episodes. On Twitter, users tend to alternate tweeting and retweeting during active periods, and that tweets and retweets are generated by different mechanisms. This results in two different classes of events, tweets, and retweets. Moreover, this motivates the usage of a bivariate point process. The suggested point process uses several parameters, including the probability that the user starts an episode with a tweet an not a retweet, the varying time between two episodes, the number of tweets and retweets in an episode, and the inter-event time between tweets and retweets in an episode.

2.5.5 Self-Feeding Process (SFP)

For a long time, a major drawback of modeling user activity using Poisson point processes was the assumption that inter-event times are independent and identically distributed. However, this is not the case because when users write emails, for example, the probability that they write an email at a certain time depends on previously written ones. The Self-Feeding Process (SEP) model tries to overcome this using a Markovian approach. It is assumed, that the time of the next event happening depends on exactly the previous one. In order to achieve this, every inter-event time is defined as an exponentially distributed random variable. Exponential probability distributions are characterized by the rate

parameter, which depends on the previous inter-event time in the SEP model. Thus, SEP generates events such that inter-event times follow the power law, and the process behaves like a Poisson point process within a short time frame. An additional advantage of the SEP model is that it usually requires only a single parameter, the desired median of inter-event times.

2.5.6 Rest-Sleep-and-Comment (RSC) Model

As already stated above, inter-event times show periodic spikes, and the inter-event time t_i depends on the previous, t_{i-1}. Inter-event times follow a bimodal distribution. The CCDF is a long tail. The purpose of the RSC model, as described in [35], is to match those characteristics as best as possible. There are three phases the algorithm can be in, Rest, Sleep, and Comment (Active). Using the variable t_{clock}, the algorithm keeps track of the current time of the day. t_{clock} advances whenever an event happens. In the Comment phase, events are generated using a self-correlated process, which ensures that inter-event times are not independent. In this phase, users actually post something on social media. In the Rest phase, no events are generated. These are the passive phases during the day where users do not post anything on social media. In the Sleep phase, a single event is generated, which is only used to advance t_{clock} to the next Comment phase. In other words, this phase is the time of the day where the user sleeps, and the event is used to move the clock time to the morning. RSC has some interesting applications. Besides the simulation of user behavior, it can be used for the detection of bots on the Internet.

Chapter 3

Synthetic Analysis

This chapter presents an analysis of the behavior of transfer entropy based on synthetic test data. The test data generation process can be easily configured by a set of parameters. The test data presents a specific scenario, for example, a highly active actor, and therefore the behavior of transfer entropy in such cases can be studied. First, we outline how test data is created in general using two different models, the Influencer-Follower model and the Coupled model, and describe the set of tunable parameters. Next, we discuss the influence of these parameters in a scenario of two actors and in a scenario of a network of multiple actors. Additionally, we discuss different approaches to normalize transfer entropy.

3.1 Test data generation

Transfer entropy between agents in a social network depends on many different parameters, including the activity of agents and their strength of influence [80]. In order to better understand the influence of those parameters, we construct a way to generate meaningful test data such that aspects of real-life networks are covered as good as possible. The parameters which are covered by our test data generation framework are the strength of influence an actor has on another and the level of activity with which an actor operates. The strength of influence γ is the amount of influence an actor has on another actor. It describes how likely it is that actor B executes an action because actor A executed one recently. Some actors act more frequently than others, some actions are independent, and some are influenced by other actors. μ describes the number of independent actions executed by an actor. In some cases, actors act entirely on their own. The frequency of those events is described by parameter λ.

As shown by Ver Steeg et al. [80], a method to cover all these aspects is the usage of a Poisson point process. We distinguish between two major cases: the Influencer-Follower model and the Coupled model. Actors in the Influencer-Follower model can either influence other actors, *i.e.*, be an influencer, or be influenced by other actors,

i.e., be a follower but not both at the same time. Actors in the Coupled model can do both, and it is possible to be influencer and follower at the same time. The outcome of the test data generation process is a list of timestamps S_X for each actor X in the network which describes when a certain event occurs. In order to control the amount of generated timestamps we use parameter T. It describes the time boundary when the data generation process ends, *i.e.*, $\forall t_i \in S_X : t_i < T$.

3.1.1 Influencer-Follower model

Using the Influencer-Follower model, we simulate networks which are strictly separated into two groups of users: influencers and followers. Influencers have their own independent pattern of activity, which is not aligned to the pattern of other actors in any way. They operate in a completely isolated, time-independent way. In order to characterize the activity generated by influencer X, we use a homogeneous Poisson point process with a constant frequency rate for the influencer's activity λ. The activity pattern of followers is partially influenced by the activity pattern of their influencers. They operate in a time-dependent way because if the influencer X causes an event, the probability that the follower Y also causes an event is higher. In order to characterize the activity generated by follower Y, we use a non-homogeneous Poisson point process with a time-dependent rate for the follower's activity $\lambda_Y(t)$ [80].

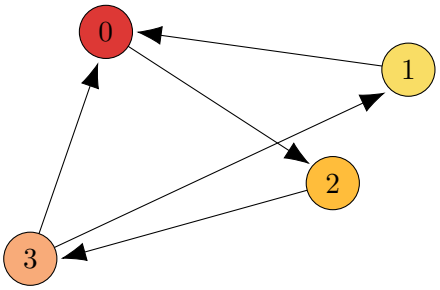
$$\lambda_Y(t) = \mu + \gamma \sum_{t_i \in S_x^t} \min\left(1, \left(\frac{1 \text{ hour}}{t - t_i}\right)^3\right) \tag{3.1}$$

μ represents the rate of constant background activity, *i.e.*, the number of events generated by Y independently without being influenced. γ denotes how strong the influence of X is. S_X denotes the set of timestamps of events executed by X. S_X^t includes all timestamps between 0 and t. The usage of S_X^t adds time-dependency to the Poisson point process of Y. In case a follower has more than one influencer, the equation changes to the following [80].

$$\lambda_Y(t) = \mu + \sum_{X \in \mathcal{I}(Y)} \gamma_X \sum_{ti \in S_x^t} \min\left(1, \left(\frac{1 \text{ hour}}{t - t_i}\right)^3\right) \tag{3.2}$$

Compared to Equation 3.1, all actors which influence Y, denoted by $\mathcal{I}(Y)$, are now considered.

In real life, there exist several examples of these networks. For instance, some users on the popular photo-sharing platform Instagram act literally as influencers having a certain amount of followers. Influencers create photos independently from each other. Influencer Alice might post a photo of colorful autumn leaves in the street. Follower Bob, who follows Alice, might then, as an event, either repost this photo or feel inspired by Alice and post his own photo of colorful autumn leaves in the street. One might argue that



$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

$$\mathcal{I}(0) = \{1, 3\}$$
$$\mathcal{I}(1) = \{3\}$$
$$\mathcal{I}(2) = \{0\}$$
$$\mathcal{I}(3) = \{2\}$$

Figure 3.1: Example network constructed from adjacency matrix A consisting of four nodes. On the right side, the influencer set $\mathcal{I}$ of each node is shown.

this is not a strict Influencer-Follower model because influencers might also affect each other. However, these cases can be viewed as corner-cases and are often undesired when analyzing data.

3.1.2 Coupled model

The Coupled model represents situations that are more similar to real-world scenarios than situations shown by the Influencer-Follower model. There is no strict separation into two groups, as the Influencer-Follower model suggested. Every actor can be influencer and follower at the same time. They act both independently and inspired by others. In order to characterize the activity generated by an actor X, a non-homogeneous Poisson point process is used similar to the one used for followers in the Influencer-Follower model. The activity rate for the point process is time-dependent. The model can be characterized by Equations 3.1 and 3.2. Note that in a network of interacting actors, $\mathcal{I}(Y)$ is given by the adjacency matrix for each actor Y. An example of this can be seen in Figure 3.1.

In real life, almost every social network can be better represented by the Coupled model than by the Influencer-Follower model. A good example is Spotify, a popular music streaming platform. Users can listen to music there, and other users can see which music they listen to. If Alice listens to a song by the Beatles, Bob might have a look at her song history and also feel like listening to the Beatles. However, the same might happen vice versa. Alice could also be inspired by Bob's song history. Compared to the Instagram case, on Spotify, the Influencer-Follower structure is less present than it is on Instagram, and therefore, the Coupled model would be the better decision to model the network synthetically.

The Influencer-Follower model is a special case of the coupled model. Influencers simply have an empty influencer set $\mathcal{I}(X) = \varnothing$. Using the background activity rate μ, which is different for each actor, the role of λ can be modeled.

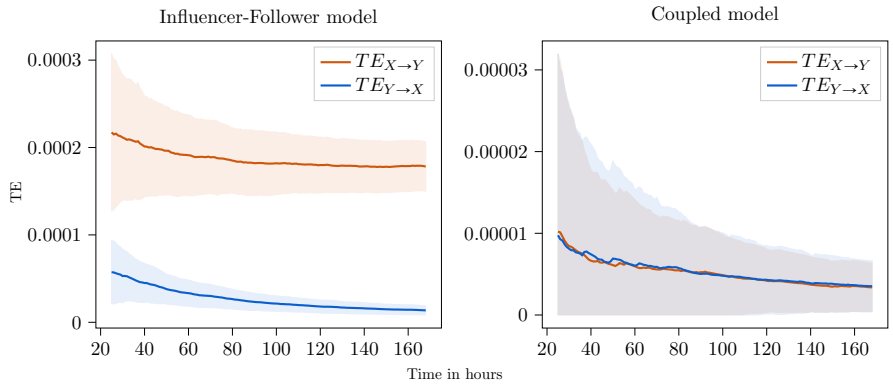


Figure 3.2: Computation of transfer entropy over a longer time frame (6 days). We simulate 200 pairs of users and take the average transfer entropy which is evaluated every hour. The left side shows the results for the Influencer-Follower model. The right side shows the results for the Coupled model. The shaded regions represent the standard deviation.

3.2 Computing transfer entropy over time

The first experiment studies transfer entropy as a function of time over a longer time frame. We expect that transfer entropy is higher in the beginning because there is only limited knowledge about X and Y available. With each time step, more and more is known about X and Y, and uncertainty decreases. However, at some point in time, knowing more about the respective stochastic process does not help anymore. The curve converges to a particular value, as it was already shown in literature [24, 27, 63]. In the Influencer-Follower experiment, we expect that $TE_{Y\to X}$ converges to a value that is almost zero because there is no influence. In the Coupled experiment, we expect that $TE_{Y\to X}$ is higher than $TE_{X\to Y}$ since we chose parameters such that the influence from Y to X is higher.

We set $T = 1$ week $= 168$ hours which means that we simulate actor behavior over a time span of one week. We compute transfer entropy for the last six weeks every 60 minutes. The past three hours of history are considered by choosing the bin widths to $\delta_0 = 1$ second, $\delta_1 = 1$ hour, $\delta_2 = 2$ hours. We set the rate of independent actions to a very low value ($\mu = 0.01$) because we do not want this parameter to influence the results much. In the Influencer-Follower model, the influencer activity rate λ is set to 0.9, and the influence strength is set to 2. In the Coupled model, the influence strengths are set to $\gamma_X = 0.01$ and $\gamma_Y = 0.05$.

As described by Figure 3.2, transfer entropy is high in the beginning because uncertainty in both processes is high due to insufficient knowledge. We reach convergence in both cases after around 100 hours. The Influencer-Follower example shows clearly that influencer X influences follower Y, but not vice versa since $TE_{Y\to X}$ is almost zero in the end. In the Coupled model Y influences X stronger than X influences Y by setting $\gamma_X < \gamma_Y$. This can be observed in the Figure since $TE_{Y\to X}$ is almost always above $TE_{X\to Y}$.

Interestingly, when comparing the Influencer-Follower model and the Coupled model, the curve is much less smooth in the Coupled model than it is in the Influencer-Follower model. This might be because in the Coupled model, there are two processes influenced by each other. In the Influencer-Follower model, the information flow is unidirectional.

3.3 Comparison of parameters

The first part of our experiments examines transfer entropy observed between two users in the Influencer-Follower model. We omit the Coupled model, but provide a detailed analysis of it in the next section. The reason why we conduct the study with only two actors is that it helps to break down the complexity of processes in the whole system and form a better understanding of how different parameters influence transfer entropy. Unless otherwise stated, we use the following standard configurations in the experiments:

- Influencer activity rate $\lambda = 0.9$
- Influence strength $\gamma = 2$
- Background activity rate $\mu = 0.01$
- Bins: $\delta_0 = 1$ second, $\delta_1 = 1$ hour, $\delta_2 = 2$ hours
- Observation time $T = 7$ days

The notation of the bins is organized as follows: δ_0 refers to the bin used for the most recent events. δ_i refers to bins used for events further in the past. The higher i, the more time passed since the occurrence of the event.

3.3.1 Comparing background activity rate (μ) and influence strength (γ)

Expectations. In this experiment, we investigate the influence of background activity rate (μ) and the influence strength (γ) on transfer entropy between two users in the Influencer-Follower model. First, we expect that transfer entropy decreases as the background activity rate μ increases. The follower then does more things on his own and independent from the influencer. Second, we anticipate that transfer entropy increases as the influence strength γ increases. The more the influencer inspires the follower's actions, the more transfer entropy will be there. Therefore, we expect to see the highest transfer entropy in the case where γ is high, and μ is low. The lowest transfer entropy is expected to be in the opposite case, with a low γ and a high μ.

Results. In the test, we observe 200 users for 7 days and report the average transfer entropy at the end of the period. According to the standard configurations, λ is set to 0.9 in the experiment. Fourteen different values for μ and γ are tested in the range from 0 to 3.

Figure 3.3 shows the results of the experiment. The number of timestamps generated for the influencer X is 130 on average. It does not depend on the configuration since the influencer behaves independently from μ and γ. The number of timestamps generated for the follower Y for each configuration can be seen in Table 3.1. The more the follower

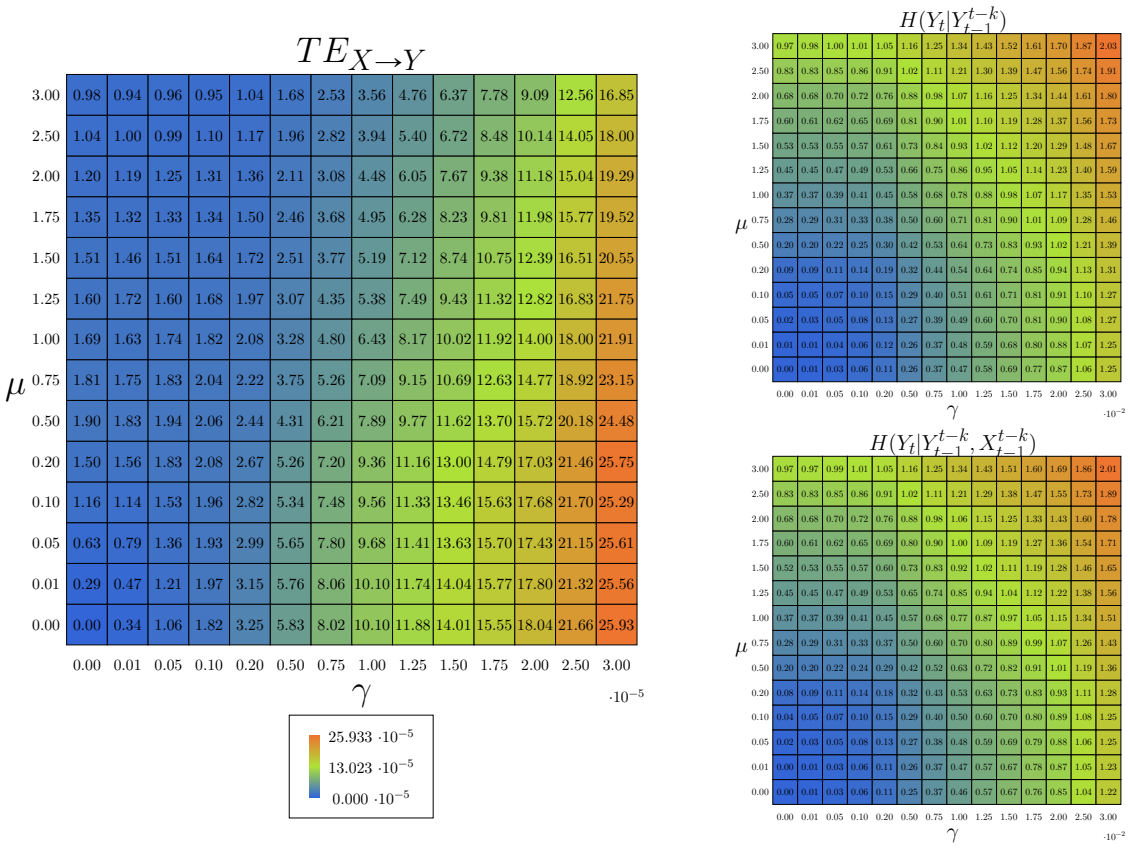


Figure 3.3: Influence of background activity rate μ and influencer strength γ on transfer entropy in the Influencer-Follower model. The first heat map shows $TE_{X\to Y}$, the transfer entropy we computed from X (influencer) to Y (follower). The higher the influence strength and the lower the background activity rate, the higher transfer entropy. On the right, there are two heat maps showing the conditional entropies $H(Y_t|Y_{t-1}^{t-k})$and $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k})$ which are highest when the amount of timestamps is the largest because then, uncertainty is the highest.

does independently (higher μ), the more independent events are generated. The higher the influence strength (higher γ), the more events are generated, which are inspired by the influencer.

Considering $TE_{X\to Y}$, four different cases can be distinguished. First, if both values are low, the transfer entropy is low, too. This means that there are not many timestamps in S_Y. $H(Y_t|Y_{t-1}^{t-k})$ is low because there is not much activity, and therefore, Y seems to be very certain.

In the case where μ is high, and γ is low, transfer entropy is low, even though it is higher than in the previous case. The uncertainty in Y_t given the history of Y increases because more about Y is known. At the same time, since the influence strength is low, the history of the influencer does not reduce the uncertainty of Y much.

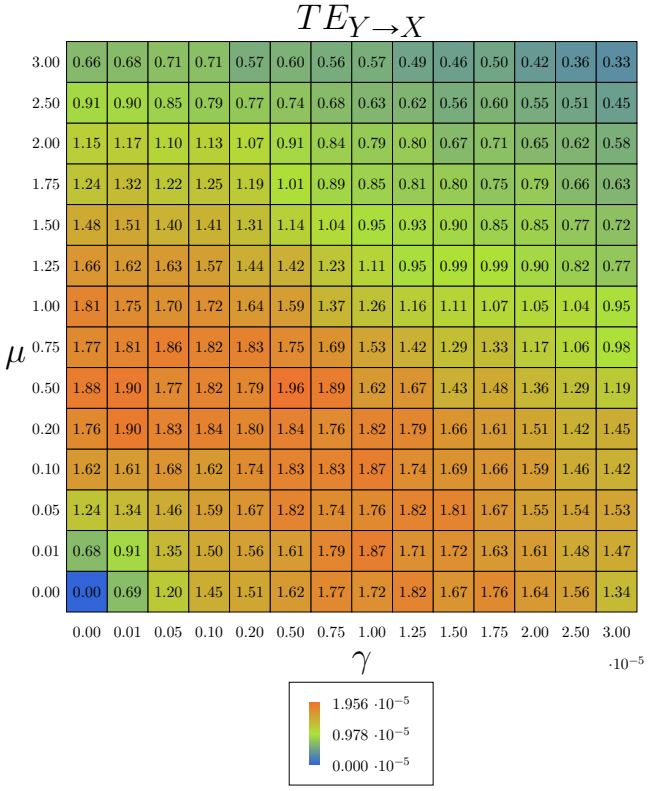


Figure 3.4: Influence of background activity rate μ and influencer strength γ on transfer entropy in the Influencer-Follower model. The heat map shows $TE_{Y\to X}$, the transfer entropy we computed from Y (follower) to X (influencer).

The most information flow happens if γ is high, and μ is low, *i.e.*, the influence strength is high, and the background activity rate is low. The follower's history is very similar to the influencer's history because it follows everything the influencer does. The uncertainty in Y, given the history of Y, is reduced a lot by the knowledge of the history of X. At the same time, the uncertainty in both processes is rather low.

In the last case, where both γ and μ are high, transfer entropy is lower. The reason for this is that now there is high uncertainty in the process of the follower. The follower acts very often, around ten times more often than the influencer, and does a lot independently.

Figure 3.4 shows the amount of transfer entropy from the follower to the influencer. In general, the absolute value of transfer entropy is more than 20 times lower than the values for $TE_{X\to Y}$, meaning that the influence is much lower in general. Transfer entropy is low for small values of μ and γ, then increases in the range $\mu \in [0.2, 1.25], \gamma \in [0.5, 2]$ before it decreases again. One might think of it as Y is not useful in explaining the history of X when there are not enough events or too many events. The uncertainty in X remains constant for all configurations, but the uncertainty in X given the history of X and Y changes.

Summary. The higher the background activity rate μ, the lower the transfer entropy is. The higher the influence rate γ, the higher the transfer entropy is.

Length of S_Y

		γ													
		0.0	0.01	05	0.1	0.2	0.5	0.75	1	1.25	1.5	1.75	2	2.5	3
	0	0	1	9	18	39	96	146	190	242	291	333	382	479	577
	0.01	1	3	11	20	40	98	146	194	243	289	344	389	485	580
	0.05	7	9	17	26	46	102	152	198	250	299	350	396	488	589
	0.1	14	16	24	32	52	110	159	206	255	301	353	399	497	588
	0.2	28	30	38	48	67	124	173	222	269	318	370	418	512	608
	0.5	72	73	81	91	112	168	214	265	313	361	409	457	555	648
μ	0.75	107	109	117	128	147	204	251	299	351	397	448	489	590	685
	1	145	145	153	164	182	238	287	335	382	436	477	530	629	725
	1.25	180	182	189	199	218	275	320	373	420	470	514	565	655	755
	1.5	214	218	226	236	252	313	362	410	458	507	548	594	697	800
	1.75	251	252	261	271	291	348	396	448	495	544	590	638	740	836
	2	289	289	299	305	326	384	435	481	527	577	622	674	771	875
	2.5	359	360	370	377	398	456	503	553	600	648	694	739	841	938
	3	431	433	444	451	471	529	576	624	671	720	766	819	915	1004

Table 3.1: Number of timestamps generated for Y, the follower for each configuration of background noise rate μ and influencer strength γ. As μ increases,the number of timestamps increases because the follower does more independently. As γ increases, the number of timestamps increases because the follower's dependence from the influencer is higher.

3.3.2 Comparing influencer activity rate (λ) and influence strength (γ)

Expectations. In this experiment, we investigate the influence of influencer activity rate (λ) and the influence strength (γ) on transfer entropy between two users in the Influencer-Follower model. First, we expect that transfer entropy increases as the influencer activity rate λ increases. This was already shown by several experiments in literature [80]. They found that users with a high activity level (bots, spammers) automatically cause a higher transfer entropy. Second, we anticipate that transfer entropy increases as the influence strength γ increases. The more the follower's actions are inspired by the influencer, the more transfer entropy will be there. Therefore, we expect to see the highest transfer entropy in the case where λ is high, and γ is high. The lowest transfer entropy is expected to be in the opposite case, with a low λ and a low γ.

Results. In the test, we observe 200 users for 7 days and report the average transfer entropy at the end of the period. According to the standard configurations, μ is set to 0.01 in the experiment. Fourteen different values for λ and γ are tested in the range from 0 to 3.

Table 3.2 shows the number of timestamps generated for the influencer X for each configuration. Since λ is the only parameter that can be configured for X, the number of

Length of S_X

λ													
0.0	0.01	0.05	0.1	0.2	0.5	0.75	1.0	1.25	1.5	1.75	2.0	2.5	3.0
0	1	7	14	28	71	108	144	179	216	251	287	359	431

Table 3.2: Average number of timestamps generated for X, the influencer for each configuration of influencer activity rate λ. The influencer strength γ does not affect the number of timestamps of the influencer.

Length of S_Y

		γ													
		0.0	0.01	0.05	0.1	0.2	0.5	0.75	1	1.25	1.5	1.75	2	2.5	3
	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	0.01	1	1	1	1	1	2	3	3	4	4	5	5	6	7
	0.05	1	1	1	2	3	6	9	11	14	17	20	23	29	34
	0.1	1	1	2	3	5	12	16	22	28	32	39	43	54	62
	0.2	1	1	3	5	10	22	32	45	55	66	75	87	108	133
	0.5	1	2	6	12	23	55	81	107	134	161	185	215	271	322
λ	0.75	1	3	9	18	34	82	121	160	202	244	285	328	398	484
	1	1	3	12	22	44	110	162	219	269	326	378	430	538	643
	1.25	1	4	14	28	55	134	202	265	336	402	469	545	665	808
	1.5	1	4	17	33	66	165	243	321	405	477	561	640	805	974
	1.75	1	5	20	39	76	186	281	378	463	565	661	756	934	1130
	2	1	5	23	43	86	215	322	429	539	643	753	849	1071	1289
	2.5	1	6	28	54	108	267	403	537	672	800	934	1072	1341	1609
	3	1	7	33	66	129	321	479	643	803	963	1129	1282	1604	1944

Table 3.3: Number of timestamps generated for Y, the follower, for each configuration of influencer activity rate λ and influencer strength γ. As λ and γ increase,the number of timestamps increases because the follower rarely acts independently due to the low background noise rate.

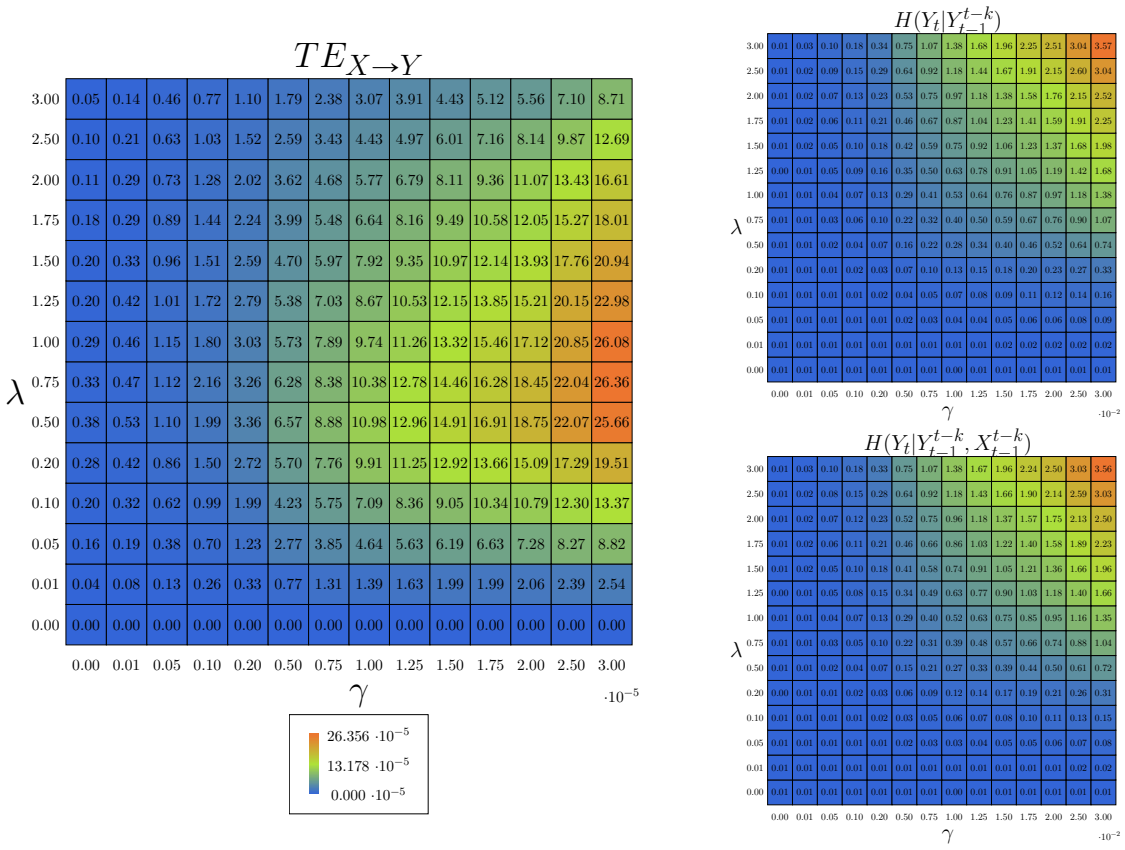


Figure 3.5: Influence of influencer activity rate λ and influencer strength γ on transfer entropy in the Influencer-Follower model. The heat map shows $TE_{X\to Y}$, the transfer entropy we computed from X (influencer) to Y (follower). If the influencer is too active or too inactive, transfer entropy does not provide a meaningful measure of influence for these cases. For a fixed λ (for each row), transfer entropy increases with the influence strength γ. On the right, there are two heat maps showing the conditional entropies $H(Y_t|Y_{t-1}^{t-k})$and $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k})$ which are highest when the amount of timestamps is the largest because then, uncertainty is the highest.

timestamps does not change for different values of γ. The higher λ, the more active is the user, and the more timestamps are generated. Table 3.3 shows the number of timestamps generated for the follower Y for each configuration. Both λ and γ influence the follower, and the higher both parameters, the more timestamps are generated. γ can be seen as the fraction of following events, while μ is the fraction of independent events. The more events the influencer generates, the more often the follower follows and the absolute number of following events increases.

The first part of our analysis focuses on $TE_{X\to Y}$. The results are shown in in Figure 3.5. A very interesting pattern can be identified in the heat map. The most information is transferred for configurations around $(\lambda = 0.75, \gamma = 3)$. The increase is neither linear nor diagonal but slightly round.

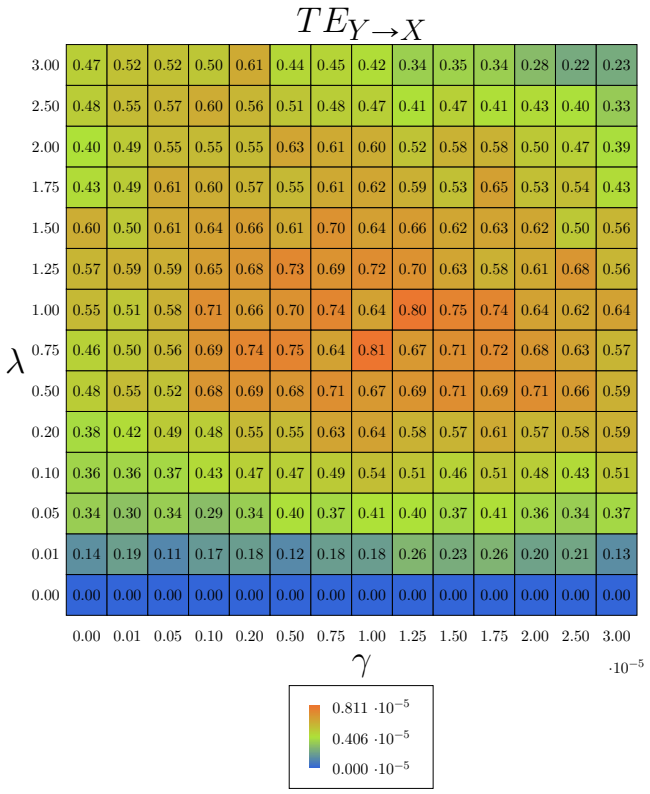


Figure 3.6: Influence of influencer activity rate λ and influencer strength γ on transfer entropy in the Influencer-Follower model. The heat map shows $TE_{Y\to X}$, the transfer entropy we computed from Y (follower) to X (influencer).

In the left lower part of the heat map, transfer entropy is very low. The influencer does not do much. There are less than 100 events generated. Since also the influence strength is low and μ is very low, transfer entropy is low because there is simply not enough data to obtain a meaningful analysis of timestamps. Both $H(Y_t|Y_{t-1}^{t-k})$ and $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k})$ are small because there are so few events that it seems that both processes are very predictable.

In the left upper part of the heat map, similar observations can be made. However, now the influencer does more, and between 100 and 500 events are generated. Entropy $H(Y_t|Y_{t-1}^{t-k})$ increases when the influence strength is higher.

Considering the right half of the heat map, for $\lambda \in [0, 0.75]$, transfer entropy is high if γ is high too. Transfer entropy grows as both parameters grow. This would mean that the more the influencer does and the higher the strength of influence is, the higher is the amount of transferred information. This confirms the findings which were made by [80] about spammers and bots and our initial assumption.

However, the reverse thing happens for $\lambda \in [1, 3]$. Transfer entropy grows as γ grows

but decreases as λ grows. It is also shown by $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k})$ and $H(Y_t|Y_{t-1}^{t-k})$ that uncertainty increases as the number of events increases. In the case where $(\gamma = 3, \lambda = 3)$, the uncertainty in both processes reaches its maximum. There are so many events that it is very hard to make predictions about the stochastic process.

Considering our empirical analysis, this means that we need to be careful with the choice of the number of actors events. First, actors who do not do much need to be excluded $\lambda \in [0, 0.2]$. Second, actors who do too much also need to be filtered out $\lambda \in [1.5, 3]$. Regarding the probability distribution $P(Y_t, Y_{t-1}^{t-k})$, we will see a peak in the bin *0/11* because it is very likely, when there are a lot of events, that at least one historic bin is filled. An example for $\gamma = 3, \lambda = 3$ can be seen in Figure A.1.

A possibility to avoid this situation is to choose bin widths carefully by considering the distribution of time between events, the *inter-event time*.

It also seems to be very important that the number of events of the influencer and the follower is nearly the same. From Tables 3.2 and 3.3 we observe that ideally the ratio between the number of influencer timestamps and the number of follower timestamps is around 1:4. Additionally, the level of activity must be sufficiently high. A possibility to ensure this is to take the actor with the least amount of events N and then randomly select between N and $2 \cdot N$ timestamps from the history of the other actors to compute transfer entropy. This ensures that the ratio of events is met for each actor. The whole process is repeated sufficiently often, and the transfer entropy can be computed as an average of all values.

Our second assumption, the higher γ, the higher the transfer entropy, is confirmed by the results. If one considers each row of the heat map individually, transfer entropy increases as γ increases. This is true for every single row except the first one, where $\lambda = 0$, which is due to the lack of events.

In the second part of the analysis, we focus on $TE_{Y \to X}$. The results are presented in Figure 3.6. Although again, transfer entropy is much lower than in the first case, the highest values lie in the middle in this case and spread like a circle. A possible reason for this is that in the middle region, the number of timestamps for both X and Y is very similar. For high values of λ and γ, there are many timestamps generated for the follower Y. Then, the knowledge of the follower process does not help that much any more to predict the behavior of the influencer. For low values of λ, there are again not enough events to make any meaningful assumptions.

Summary. We must exclude actors below and above a certain activity level from analysis because then, transfer entropy does not provide a meaningful measure of influence. For a fixed influencer activity rate λ, transfer entropy increases as the strength of influence γ increases.

3.3.3 Comparing background activity rate (μ) and influencer activity rate (λ)

Expectations. In this experiment, we investigate the influence of background activity rate (μ) and influencer activity rate (λ) on transfer entropy between two users in the Influencer-Follower model. First, we anticipate that since the background activity rate refers to the actions done by the follower individually, transfer entropy decreases with an increasing background activity level. Second, we expect to see the same effects regarding the influencer activity rate as in the last experiments: a too low value for λ means there are not enough events, a too high value for λ means that there are too many events to compute a meaningful value for transfer entropy.

Results. In the test, we observe 200 users for 7 days and report the average transfer entropy at the end of the period. According to the standard configurations, γ is set to 2 in the experiment. Fourteen different values for μ and λ are tested in a range from 0 to 3.

Table 3.2 shows the number of timestamps generated for influencer X for each configuration. Since λ is the only parameter that can affect X's behavior, the number of timestamps does not change for different values of μ and is, therefore, the same as in the previous experiment. Table 3.4 shows the number of timestamps generated for the follower Y for each configuration. The higher μ and λ, the higher the number of timestamps. We are able to explain based on tha table what a value of $\mu = 3$ effectively means: If $(\mu = 3, \lambda = 0)$, there are 431 events in the history of Y. If $(\mu = 0, \lambda = 3)$, there are 1283 events in the history of Y. If $(\mu = 3, \lambda = 3)$, there are 1722 events in the history of Y. This means that the fraction of independently executed events is around 25% in this case, and 75% are influenced. For $(\mu = 3, \lambda = 1)$, the ratio of both event types is equal.

Length of S_Y

	λ \ μ	0	0.01	0.05	0.1	0.2	0.5	0.75	1	1.25	1.5	1.75	2	2.5	3
	0	0	1	6	14	29	72	107	144	182	215	251	287	361	431
	0.01	4	5	11	18	33	77	112	147	184	220	255	293	363	437
	0.05	22	23	28	33	50	94	129	164	201	235	276	310	379	454
	0.1	41	44	50	56	72	113	150	186	222	258	296	330	399	475
	0.2	84	87	94	98	113	158	191	228	267	302	337	374	446	517
	0.5	214	215	225	227	242	289	321	356	395	426	469	502	578	644
λ	0.75	321	322	329	334	353	391	429	464	501	533	573	608	677	745
	1	427	430	430	440	450	500	540	576	607	640	679	717	790	856
	1.25	537	536	542	551	567	613	645	677	714	753	794	818	893	969
	1.5	644	642	650	650	669	714	749	790	818	855	900	935	998	1069
	1.75	743	756	755	771	786	829	860	892	931	965	1007	1037	1105	1179
	2	862	854	879	871	885	927	964	998	1032	1080	1109	1149	1226	1285
	2.5	1074	1077	1078	1092	1114	1153	1184	1215	1251	1294	1325	1360	1438	1509
	3	1283	1288	1298	1311	1312	1360	1389	1420	1461	1504	1542	1566	1648	1722

Table 3.4: Number of timestamps generated for Y, the follower, for each configuration of influencer activity rate λ and background activity rate μ. As λ and μ increase, the number of timestamps increases because due to the follower acts more often independently and follows more often what the influencer does.

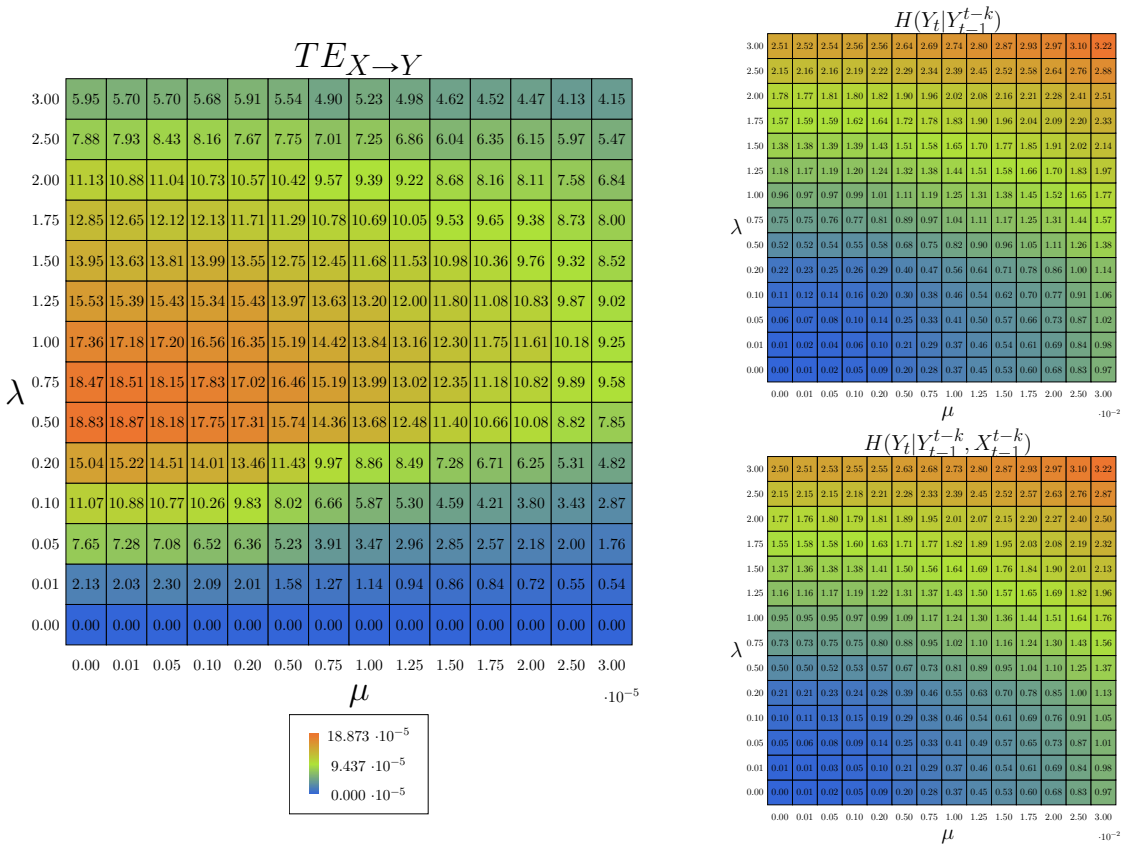


Figure 3.7: Influence of influencer activity rate λ and background activity rate μ on transfer entropy in the Influencer-Follower model. The first shows $TE_{X\to Y}$, the transfer entropy we computed from X (influencer) to Y (follower). If the influencer is too active or too inactive, transfer entropy does not provide a meaningful measure of influence for these cases. For a fixed λ (for each row), transfer entropy decreases with the background activity rate μ. On the right, there are two heat maps showing the conditional entropies $H(Y_t|Y_{t-1}^{t-k})$and $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k})$ which are highest when the amount of timestamps is the largest because then, uncertainty is the highest.

The first part of our analysis focuses on $TE_{X\to Y}$. The results are shown in Figure 3.7. The first heat map shows results that are similar to the ones presented in Section 3.3.2. The amount of transfer entropy again seems to concentrate around the region of $\lambda \in [0.2, 1.25]$.

When comparing the two lower halves of the heat map, it is noticeable that transfer entropy is higher in the left part where the follower's independent actions are not frequent. The lower half is also a region where the uncertainty in the process of Y is relatively low because the general activity level is low. The process X helps to predict the process of Y if there are enough events. This is the case for $\lambda > 0.1$. One must always keep in mind that the uncertainty level of the process X is constant for a fixed λ and independent from μ. Therefore, for each λ, the higher μ is, the lower is the transfer entropy because the

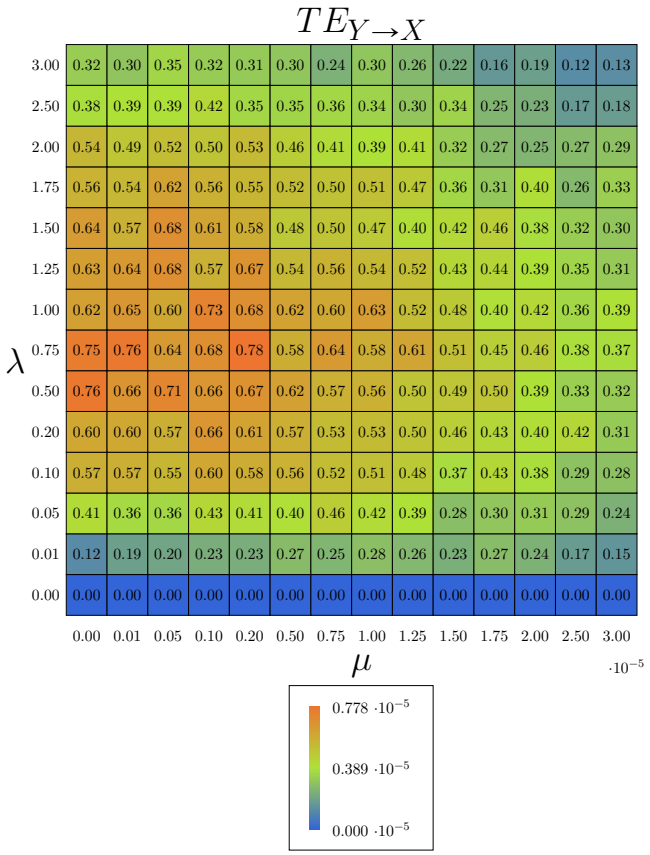


Figure 3.8: Influence of influencer activity rate λ and background activity rate μ on transfer entropy in the Influencer-Follower model. The heat map shows $TE_{Y \to X}$, the transfer entropy we computed from Y (follower) to X (influencer).

stochastic process Y has more uncertainty itself. However, if X does not fulfill a certain level of activity ($\lambda \in [0, 0.01]$), it cannot help to reduce uncertainty at all.

$\lambda \in [0.2, 1.25]$ and $\mu \in [0, 1]$ is the area where transfer entropy is the highest. Interestingly, in this region, the fraction of individual events is between 0 and 21%. If more than 21% of the follower's actions are individual actions, transfer entropy decreases.

We again observe that if λ is too high, transfer entropy decreases because there is a lot of uncertainty in both processes due to the huge number of events. As it was already shown in the experiment before, we must be careful to not consider too active or too inactive influencers in our analysis. However, what is new is that we must also make sure that the followers are not too active ($\mu < 1$). Limiting the number of events might also improve analysis results.

Furthermore, for a fixed λ, our assumptions are confirmed. Considering one row in the heat map, transfer entropy decreases as the amount of background activity μ increases. This is true for every single row.

The second heat map, which is shown in Figure 3.8, shows $TE_{Y\to X}$. Interestingly, the distribution pattern of the amount of information transfer looks similar to the pattern of $TE_{X\to Y}$ but a little bit more irregular. The reason for this is that the timestamp list of Y is very similar to the timestamp list of X in case the influencer does many things independently, *i.e.*, μ is low. Then, it seems like there is information flow from the follower to the influencer. However, if the follower starts to do more things independently, the timestamp list becomes more diverse, and there is less similarity. In case λ is high, we assume that there are again many events causing high uncertainty in the process of X and low transfer entropy.

Summary. We must exclude actors below and above a certain activity level from analysis because then, transfer entropy does not provide a meaningful measure of influence. For a fixed influencer activity λ, transfer entropy decreases as the background activity rate μ increases.

3.4 Normalization and correction methods

In Section 3.3, we observe several problems that reduce the meaningfulness of transfer entropy. This includes too active or too inactive influencers. In this section, we test several normalization and correction methods in order to see whether they improve results for transfer entropy. We evaluate normalized transfer entropy, correction of systematic bias, and permutation tests. Our experiments are only executed for the comparison of λ and γ. However, the results are similar in all other cases.

3.4.1 Transfer entropy normalized by $H(Y_t|Y_{t-1}^{t-k})$

Normalizing transfer entropy by $H(Y_t|Y_{t-1}^{t-k})$ gives approximately the amount of information flow in percent, as we discuss in Section 2.3.3. We denote it by $\hat{TE}_{X\to Y}$. It is computed by dividing $TE_{X\to Y}$ through $H(Y_t|Y_{t-1}^{t-k})$. $\hat{TE}_{X\to Y}$ is 1 in case X can remove all uncertainty from Y. $\hat{TE}_{X\to Y}$ is 0 in case X cannot remove any uncertainty from Y. We expect to be able to answer more clearly if there exists information flow between X and Y or not because we limit the effect which $H(Y_t|Y_{t-1}^{t-k})$ has on the size of transfer entropy. Problems related to a small $H(Y_t|Y_{t-1}^{t-k})$ leading to a big information flow might occur.

Figure 3.9 shows transfer entropy normalized by $H(Y_t|Y_{t-1}^{t-k})$. We run the experiment for 200 users over a time frame of 7 days. The average normalized transfer entropy value after the simulation period is reported. The background activity rate μ is set to 0.01. We see that whenever both parameters are low but nonzero, transfer entropy is very high. In the range $\lambda \in [0.01, 0.2]$, transfer entropy grows with the strength of influence γ. In the range $\lambda \in [0.2, 1.75]$, the transfer entropy is high in the beginning for small values of γ.

However, we run into the problems described in Section 2.3.3. In the region ($\lambda \in$

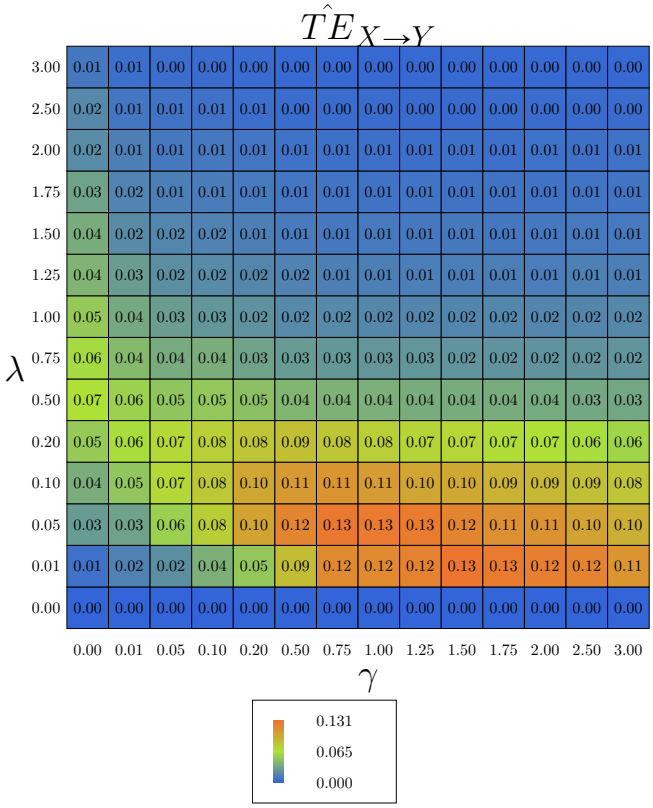


Figure 3.9: Transfer entropy normalized by $H(Y_t|Y_{t-1}^{t-k})$ for different configurations of influencer activity rate λ and influence strength γ. We run into problems of a small $H(Y_t|Y_{t-1}^{t-k})$ leading to a comparable large transfer entropy.

$[0.01, 0.2], \gamma \in [0.2, 3]), H(Y_t|Y_{t-1}^{t-k})$ is very low because there are not many events. Uncertainty is low in the process of Y. However, since $H(Y_t|Y_{t-1}^{t-k})$ is so small and $TE_{X\to Y}$ is small too, we get something comparably big even though the information flow is not big at all. For example, for $(\lambda = 0.01, \gamma = 0.75)$, $H(Y_t|Y_{t-1}^{t-k})$= 0.0001 and $TE_{X\to Y}$= 0.0000131, the normalized value is 0.12. For $(\lambda = 0.75, \gamma = 3)$, $H(Y_t|Y_{t-1}^{t-k})$= 0.0107 and $TE_{X\to Y}$= 0.0002636, the normalized value is 0.02 due to the high value of $H(Y_t|Y_{t-1}^{t-k})$. Therefore, we conclude that normalizing transfer entropy by $H(Y_t|Y_{t-1}^{t-k})$ does not improve insights that are already given to us by raw values.

3.4.2 Transfer entropy normalized by $\log k$

Normalizing transfer entropy by $\log k$ gives approximately the amount of information flow in percent. We denote it by $\hat{TE}_{X\to Y}$ and compute it by dividing $TE_{X\to Y}$ through $\log_2 k$. $k = 3$ in our experiment and refers to the number of bins used. $\hat{TE}_{X\to Y}$ is high in case the probability distributions used are very uniformly. Since $\log_2 k$ is a constant factor, we expect a meaningful scaling of transfer entropy values to a percentage scale.

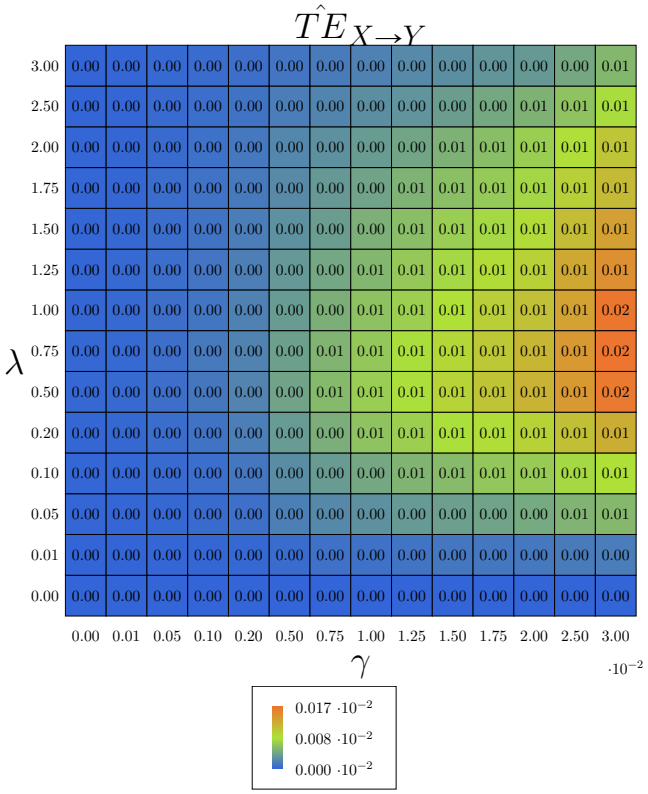


Figure 3.10: Transfer entropy normalized by $\log_2 k$ where $k = 3$ is the number of bins for different configurations of influencer activity rate λ and influence strength γ. The normalization method only provides a way to map the transfer entropy values to another scale.

We anticipate that this makes it easier to compare transfer entropy values.

Figure 3.10 shows the results of the experiment. The experiment is run for 200 users over a time frame of 7 days. The average normalized transfer entropy value after the simulation period is reported. The background activity rate μ is set to 0.01.

We see that the appearance of the heat map has not changed, confirming our expectations. However, the advantage is, that we do not have to deal with absolute values for transfer entropy any more but can work with a percentage value, which often makes outcomes easier to understand. However, it does not provide a solution to the problem of too active or too inactive actors. Additionally, it does not reduce the possibly misleading effects which $H(Y_t|Y_{t-1}^{t-k})$ has on the result. Therefore, we will not apply it to further experiments.

3.4.3 Systematic bias correction

The experiments we execute only take events into account, which happen in a certain period of time. From these events, we compute the binned random variables. All these

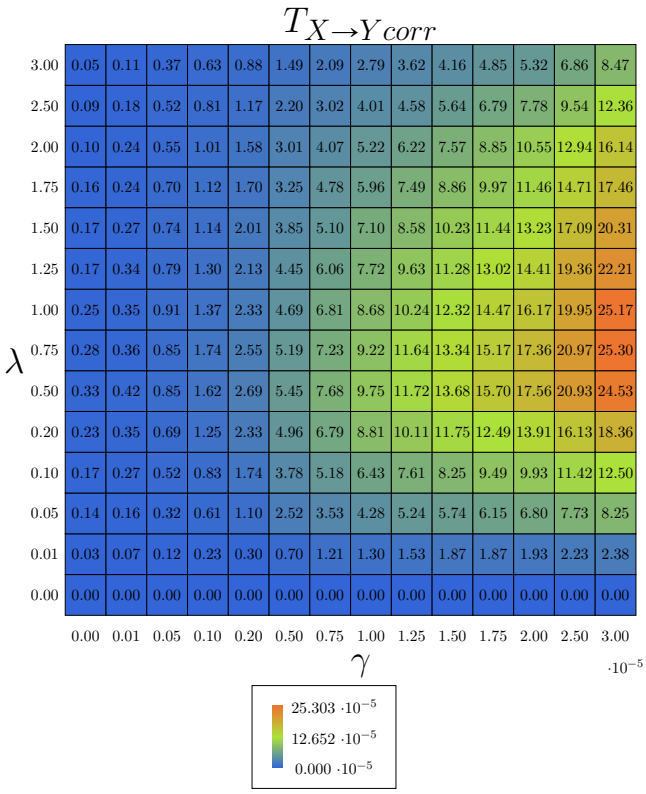


Figure 3.11: Bias-corrected transfer entropy using the Panzeri-Treves bias correction [61,62] for different configurations of influencer activity rate λ and influence strength γ. The bias correction lowers the value of transfer entropy, but does not change the relations of the values.

steps reduce the information which would be needed in order to compute the true transfer entropy and therefore distort the result. For example, given an agent acting at times $0.2, 0.5, 0.7$ and 1.2 and bin widths $\delta_0 = 1, \delta_1 = 1$, information is thrown away since the events happening at times 0.2, 0.5 and 0.7 will be in the same bin in the beginning. Actors cannot be observed infinitely long. The resulting distortion is called systematic bias [61, 62].

There are several methods of how to deal with this bias. For example, one could always ensure to take into account enough measurements such that the deviation of the estimated from the true transfer entropy is insignificant [80]. However, in real-life situations, this is not always possible, especially in cases where data is already limited. As suggested by Ver Steeg et al. [80], in such cases, it has been shown that the Panzeri-Treves bias correction, $b(H(B|A))$, for two discrete random variables A and B, gives satisfying results. It estimates the average systematic bias which occurs.

$$b(H(B|A)) = -\frac{1}{2 \cdot N \cdot ln(2)} \sum_{b \in dom(B)} (N_b - 1) \tag{3.3}$$

N is the number of joint samples of A and B. N_b is the number of unique variables $a \in dom(A)$ which are observed for a specific $b \in dom(B)$. The bias estimate subtracted from the computed transfer entropy. The equation for the corrected transfer entropy $T_{X \to Y corr}$ is then given by:

$$T_{X \to Y corr} = \Big(H(Y_t | Y_{t-1}^{t-k}) - b(H(Y_t | Y_{t-1}^{t-k})) \Big) - \Big(H(Y_t | Y_{t-1}^{t-k}, X_{t-1}^{t-k}) - b(H(Y_t | Y_{t-1}^{t-k}, X_{t-1}^{t-k})) \Big)$$

We expect that this method reduces the systematic bias introduced by sampling. Especially in situations, where we do not have much data, the corrected transfer entropy value might be useful.

Figure 3.11 shows the results of the experiment. We run it for 200 pairs of users over a time frame of 7 days. The average, bias-corrected transfer entropy value is reported after 7 days. Like in previous experiments, $\mu = 0.01$. It can be seen that the bias correction lowers the resulting transfer entropy, *i.e.*, the raw value is higher than the corrected value. The appearance of the heat map is identical to the appearance of the heat map using raw values when ignoring the absolute transfer entropy values.

Most importantly, this means that the necessity of bias correction depends on the use case. If there are two agents, X and Y, and the main interest lies in the absolute values of influence, the bias estimate must be considered. However, if the main interest lies in the relation, *i.e.*, whether X influences Y more or vice versa, the bias correction becomes less important since the relative values do not change. Since we are not interested in the absolute value in further experiments, we will not work with bias-corrected transfer entropy.

3.4.4 Permutation tests

In all our experiments, we observe that it is unknown how large the information flow from the influencer to the follower effectively is. For example, we often saw a small transfer entropy value from the follower to the influencer even though there was zero influence. The transfer entropy was observed because follower and influencer have similar histories. We need a method which allows to distinguish real from non-existent information flow. A method to deal with this is to execute a permutation test. As suggested by Santos et al. [69], this test compares the raw transfer entropy value with the transfer entropy value computed for permutations of influencer and follower histories.

Figure 3.12 shows the procedure of a permutation test for 200 user pairs. First, for every user pair, test data is generated which results in two lists of timestamps, $S_X = \{t_{X0}, t_{X1}, ...\}$ of length n_X and $S_Y = \{t_{Y0}, t_{Y1}, ...\}$ of length n_Y. These values are considered for the true or raw transfer entropy. Then, S_X and S_Y are concatenated, and the result is stored in variable S_{XY}. A permutation of S_{XY} is created by randomly shuffling it. We create a new timestamp list for X by choosing the first n_X elements from

```
1  for userPair = 1 to 200
2    //Compute true TE
3    S_X ,S_Y = generateTestData(μ, λ, γ)
4    TE = computeTransferEntropy(S_X, S_Y, bins, T)
5
6    //Compute perm. TE
7    S_XY = S_X ∪ S_Y
8    for permRun = 1 to 150
9      S_XYperm = permute(S_XY)
10     S_Xperm = S_XYperm[0:n_X]
11     S_Yperm = S_XYperm[n_X:n_X + n_Y]
12     TE_perm = computeTransferEntropy(S_Xperm, S_Yperm, bins, T)
13
14 Report average TE, average TE_perm
```

Figure 3.12: Procedure of a permutation test for 200 user pairs. First, test data is generated, and the true or raw transfer entropy is computed. Then, the transfer entropy is computed from a permutation of test data. Finally, the averages are reported for alle user pairs. n_X and n_Y refer to the number of timestamps generated for X and Y.

S_{XY}. We create a new timestamp list for Y by choosing the last n_Y elements from S_{XY}. In the end we have constructed S_{Yperm} such that $S_{Yperm} = S_{XYperm} \backslash S_{Xperm}$. Finally, we report the average raw and permuted transfer entropy values.

In general, we expect that if there is information flow from X to Y, the raw transfer entropy is higher than the average permuted value. If there is no information flow from X to Y, the raw transfer entropy is lower than the average permuted value. A normalizing effect can be achieved by computing the difference between the raw and permuted values. The normalized value can also be viewed as the true influence since the influence due to effects like history similarity is eliminated.

Figure 3.13 shows the results of the permutation test. We omit to test experiments where $\lambda \in [0, 0.5]$ because in these cases, it might happen that the history of an actor only contains one or two elements, and permuting them is useless. In the first row of the figure, the transfer entropy computed from permuted histories is represented. The distribution of the amount of transfer entropy is identical to the raw values.

In the second row of the figure, the normalized transfer entropy is illustrated. In the first heat map, which illustrates $\hat{TE}_{X \to Y}$, the raw influence reaches its maximum in the same region where also raw transfer entropy reaches its maximum. The second heat map shows $\hat{TE}_{Y \to X}$. It must be interpreted carefully, because the more negative the value, the more certain we can be that there is no influence.

In general, the test results show that using permutation tests, we can determine with great certainty if there is influence between two actors (positive normalized value) or not (negative normalized value). However, the computational effort of this method is considerable since a great number of permutations and user pairs have to be taken into account to provide a meaningful test result. Therefore, we will not work with this method in the following experiments.

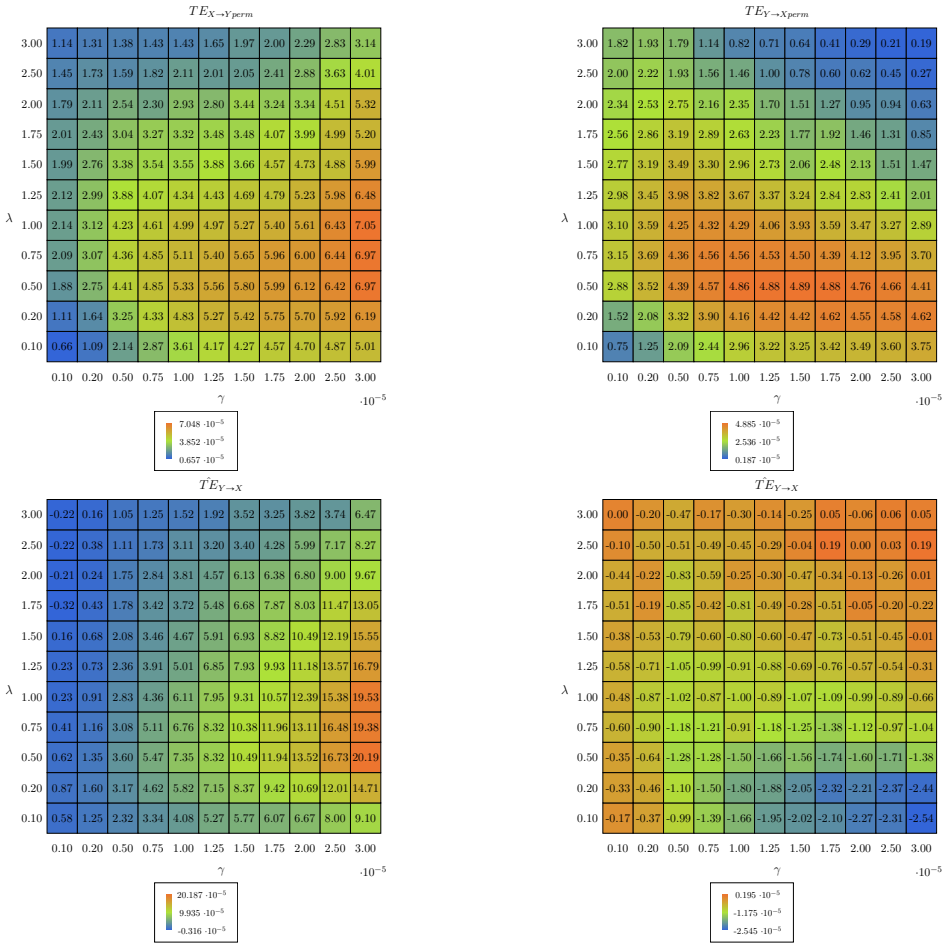


Figure 3.13: Results of normalization using permutation tests. The first row shows the average transfer entropy computed for a permutation of S_X and S_Y. The second row shows the normalized value, which is the difference between the raw and permuted value. If the normalized value is positive, there is true influence between X and Y. If the normalized value is negative, there is no true influence between X and Y.

3.5 Comparing bin widths δ_i

For the empirical analysis, only one parameter needs to be configured: the size and widths of bins. More and wider bins mean that the time frame, in which events are analyzed, is larger. Events which happened a long time ago still influence the present. Wider bins also fail to recognize single events. If the history of an actor has many peaks, small bin widths will capture this peak in a more extreme way than wider bin widths, which is sometimes not desired. Smaller bins are prone to capture less of the agent's history. In summary, choosing appropriate bin widths is not an easy task. In neuroscience, bins are often equally wide, depending on the data [80]. For the analysis of human behavior, it turned out that bins of increasing width, starting with the smallest possible unit, lead to reasonable outcomes [80].

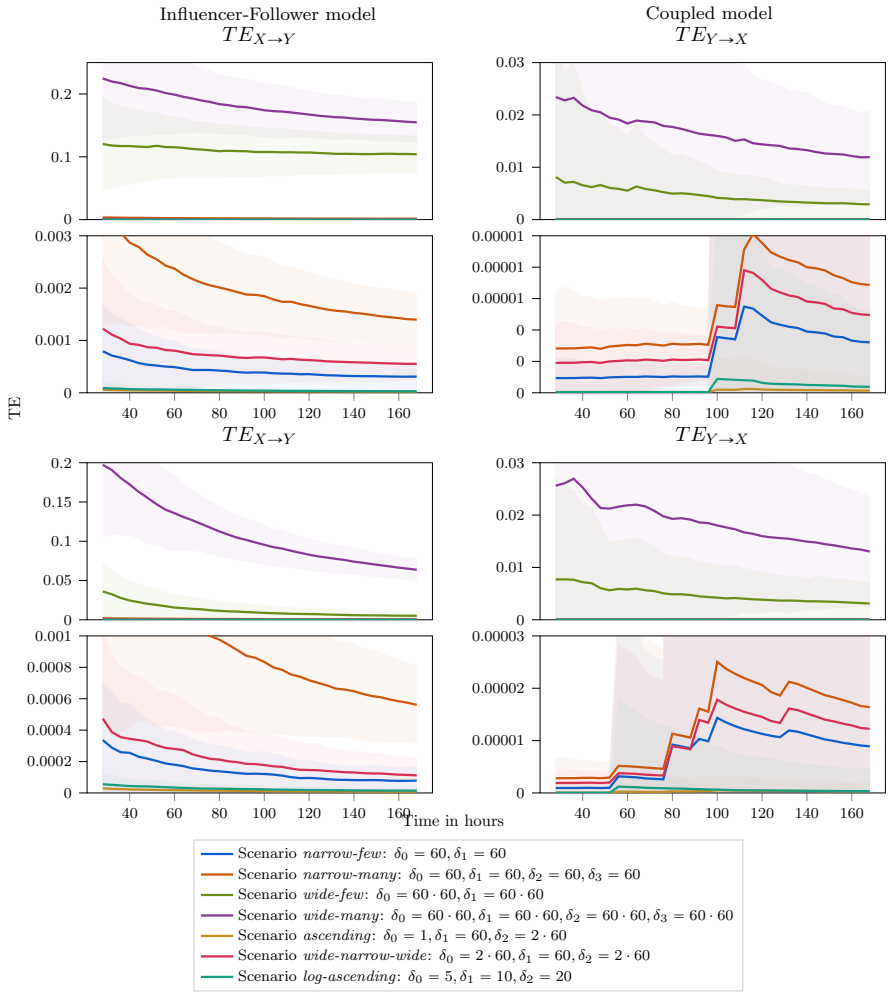


Figure 3.14: Comparison of different bin widths in the Influencer-Follower model (left) and in the Coupled model (right) over a time frame of 6 days. The more and wider bins we use, the higher is the transfer entropy due to the high uncertainty in the process.

We assume that the uncertainty in Y is larger if bins are too wide because then, many events are binned into one bin, and relatively much information is lost. Therefore, we expect the transfer entropy to be higher if the bins are wider. Accordingly, if bins are narrow, less information is cut away, and the amount of uncertainty in the process is reduced. We assume that transfer entropy is lower if bins are more narrow.

We execute our experiments using the Influencer-Follower model and the Coupled model over a time frame of 6 days and report the average transfer entropy computed for 200 user pairs. Standard configurations apply, which means we set $\lambda = 0.9, \gamma = 2, \mu = 0.01$ in the Influencer-Follower model and $\mu = 0.01, \gamma_X = 0.01, \gamma_Y = 0.05$ in the Coupled model.

Figure 3.14 shows the influence of bin widths on the test results. In both the Influencer-Follower model and the Coupled model, the bins chosen in test scenarios *wide-many* and *wide-few* lead by far to the highest amount of transfer entropy. In these scenarios, the bin widths are significantly wider than in the other scenarios, which confirms our expectation that wide bin widths lead to higher uncertainty.

narrow-many, *wide-narrow-wide* and *narrow-few* result in a higher transfer entropy than *log-ascending* and *ascending*. The bins are significantly smaller in the last two scenarios, which explains why there is such a huge difference. Interestingly, *narrow-many* results in a higher transfer entropy than *narrow-few*, from which we conclude that more bins also lead to a higher transfer entropy.

Figure 3.14 shows the test results for the Coupled model on the right side. An interesting observation can be made in the range $100 \leqslant T \leqslant 120$ for $TE_{X \to Y}$. First, transfer entropy rises rapidly, before it decreases and finally rises again. A peak can be seen around $T = 110$ before transfer entropy converges. This cannot be observed in *wide-many* and *wide-few*, but we assume that the peak happens before the observation starts ($T < 40$). In the plot showing $TE_{Y \to X}$, this behavior seems to start earlier and finish later.

In the empirical analysis, we align the bin widths according to the data. Ver Steeg et al. [80] propose a method to use different bin widths according to the average inter-event times of actors. We will apply this method in the empirical analysis. They also suggest to choose different bins for each actor. This would eliminate the problem of too active and too inactive actors.

3.6 Synthetic influence networks

The second part of our experiments examines the utility of transfer entropy in a network consisting of multiple actors. We execute tests based on both the Influencer-Follower model and the Coupled model. The basic element in these scenarios is the weighted adjacency matrix A, which defines the structure of the test network. $a_{ij} \in A$ defines how much actor i influences actor j, *i.e.*, it stores γ as a link weight. Using this adjacency matrix and the other parameters (background activity rate μ, bin widths δ_i, influencer activity rate λ), we generate a directed, weighted test network as described in Section 3.1. Each node of the network represents an actor, and every actor has a list of timestamps.

The result of each experiment is the transfer entropy matrix TE, which is obtained by computing the transfer entropy between every pair of actors in the network. Each entry $te_{ij} \in TE$ represents the amount of information flow from actor i to actor j. TE is a weighted adjacency matrix as well because if a node influences another, there is transfer entropy between them. The goal of our experiments is that the transfer entropy matrix can recover the initial network structure.

TE and A are compared against each other by computing similarity. We define two similarity measures, first the absolute difference of their Frobenius norms and second,

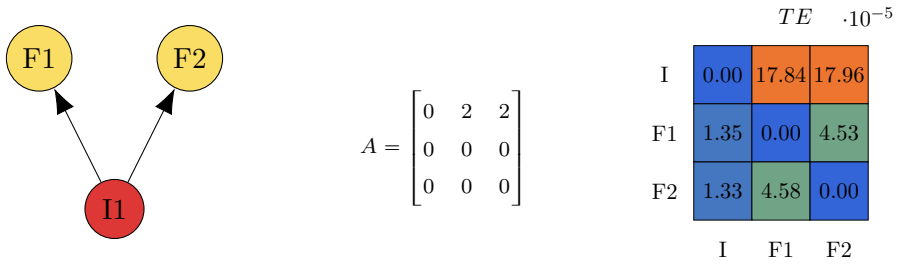


Figure 3.15: Test results for a small network with three nodes using the Influencer-Follower model. On the left, a sketch of the network is given. In the middle, there is the weighted adjacency matrix where $a_ij \in A$ describes the amount of influence from actor i to actor j. On the right, we illustrate the computed transfer entropy matrix.

the Frobenius norm of their differences.

$$sim_1 = \left| \|\overline{A}\|_F - \|\overline{TE}\|_F \right| \tag{3.4}$$

$$sim_2 = \|\overline{A} - \overline{TE}\|_F \tag{3.5}$$

sim_1 and sim_2 represent the similarity between A and TE. If sim_1 is low, the two matrices have a similar Frobenius norm, and therefore they are very similar. If sim_1 is high, TE and A are not similar. The same holds for sim_2: if the differences are high, they are not similar and vice versa. Since the value ranges of A and TE can be very different, the values are normalized by the maximum value: $\overline{a_{ij}} = a_{ij} / \max_{ij} A_{ij}$.

3.6.1 Small network with 3 actors

The first test case investigates a very small network consisting of three actors. We simulate actors' behavior over a period of $T = 7 \cdot 24$ hours and report the average transfer entropy over 200 repetitions.

The first experiment uses the Influencer-Follower model. There are one influencer, actor I, and two followers, actors F0 and F1. The influencer influences both actors equally strong with $\gamma = 2$. The influencer activity rate is set to $\lambda_I = 0.9$. The background activity rate is set to $\mu = 0.01$. Figure 3.15 shows a plot of the network and the weighted adjacency matrix. We expect that there is high information flow from the influencer to the followers, but not vice versa or between the followers.

On the right side of Figure 3.15, we plot the transfer entropy matrix. There is high information flow from the influencer to the followers There is also a smaller information flow from the followers F1 and F2 to the influencer I. This is the same phenomenon that we already saw in earlier tests when $TE_{Y \to X}$ was nonzero. Using a permutation test, one could identify this as no influence. Interestingly, there is also transfer entropy between F1 and F2. This likely happens because F1 and F2 both have the same influencer. Their pattern of activity is so similar that it seems like they influence each other. The similarity values between A and TE are $sim_1 = 0.04$ and $sim_2 = 0.38$.

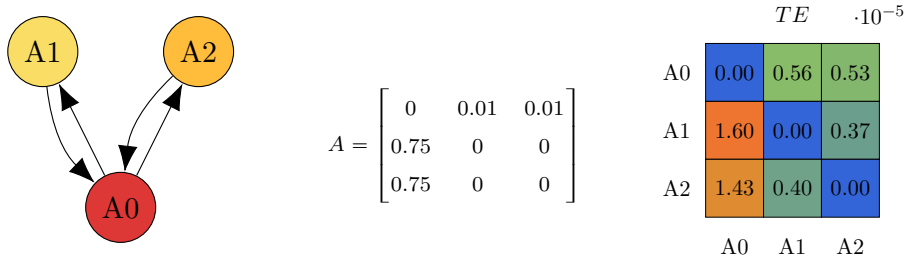


Figure 3.16: Test results for a small network with three nodes using the Coupled model averaged over 200 repetitions. On the left, a sketch of the network is given. In the middle, there is the weighted adjacency matrix where $a_ij \in A$ describes the amount of influence from actor i to actor j. On the right, we illustrate the computed transfer entropy matrix.

In the second experiment, we use the Coupled model such that actor A0 influences both A1 and A2 with $\gamma_{A0} = 0.01$ and actors 1 and 2 both influence actor 0 with $\gamma_{A1} = \gamma_{A2} = 0.75$. The network is plotted in Figure 3.16, including the weighted adjacency matrix. We expect that there is high information flow from A1 and A1 to A0. There should be a smaller information flow from A0 to A1 and A2, which is only about 1% of the transfer entropy computed in the other direction.

The transfer entropy matrix correctly shows the strong influence from actors A1 and A2 to actor A0. There is a slighter influence from actor A0 to actors A1 and A2. However, our assumption about the amount of information flow is not met. The computed transfer entropy from A0 to A1, and A2 is about 30% the information flow from A1 and A2 to A0, not only 1%. Similar to the first experiment, there is also an information flow between A1 and A2. The similarities are $sim_1 = 0.05$ and $sim_2 = 1.76$.

3.6.2 Small network with 5 actors

The second test case investigates a small network consisting of five actors. We simulate actors' behavior over a time span of $T = 7 \cdot 24$ hours and report the average transfer entropy over 200 repetitions.

The first experiment uses the Influencer-Follower model. There are two influencers, actors I1 and I2, and three followers, actors F1, F2, and F3. I1 influences all actors equally strong with $\gamma_{I1} = 2$. I2 influences all actors equally strong with $\gamma_{I2} = 1$. The influencer activity rate is set to $\lambda = 0.9$. The background activity rate is set to $\mu = 0.01$. Figure 3.17 shows a plot of the network and the weighted adjacency matrix. We expect that there is high information flow from the influencer to the followers.

The transfer entropy matrix shows high information flow from I1 and I2 to all other followers. According to the weighted adjacency matrix, I1 influences followers twice as much as I2. This relation is not detected by the transfer entropy matrix because the information flows originating at I1 are three times higher than the information flows originating at I2. The findings show a small transfer entropy between the influencers.

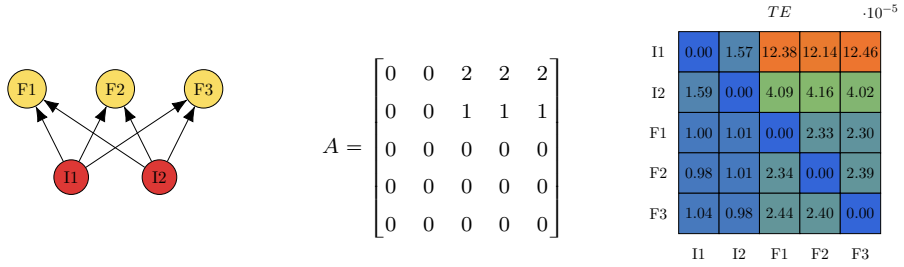


Figure 3.17: Test results for a small network with five nodes using the Influencer-Follower model averaged over 200 repetitions. On the left, a sketch of the network is given. In the middle, there is the weighted adjacency matrix where $a_ij \in A$ describes the amount of influence from actor i to actor j. On the right, we illustrate the computed transfer entropy matrix.

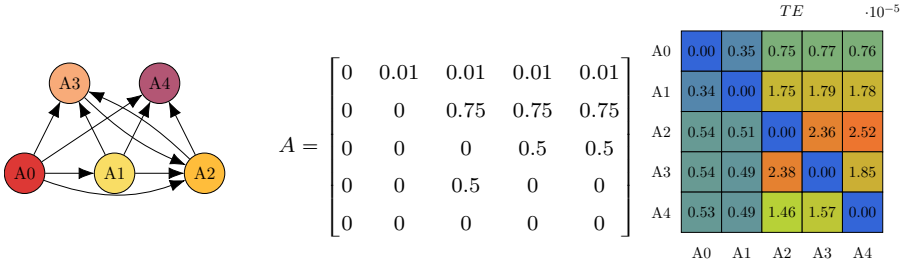


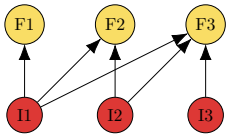
Figure 3.18: Test results for a small network with five nodes using the Coupled model averaged over 200 repetitions. On the left, a sketch of the network is given. In the middle, there is the weighted adjacency matrix where $a_ij \in A$ describes the amount of influence from actor i to actor j. On the right, we illustrate the computed transfer entropy matrix.

We suspect that this happens because the influencers have a very similar activity pattern since they use the same λ. As already observed in the previous section, there is a small information transfer between the followers. The similarities between the adjacency matrix and the transfer entropy matrix is $sim_1 = 0.05$ and $sim_2 = 0.61$.

The second experiment uses the Coupled model. A sketch of the network is given in Figure 3.18. A0 influences everyone, but only very slightly, A1 influences everyone except A0. A2 influences A3 and A4, but A3 also influences A2. A4 does not influence anyone. The background activity rate is set to $\mu = 0.01$. In Figure 3.18, we show the weighted adjacency matrix and transfer entropy matrix.

The transfer entropy matrix correctly suggests that A0 influences all other nodes. Interestingly, A0 influences everyone with the same strength, but the transfer entropy from A0 to A1 is only half the amount of the transfer entropy from A0 to all the other nodes. We also observe a reverse transfer entropy from A1 to A0. The information transfer from A1 to A2, A3, and A4 is also correctly shown by the transfer entropy matrix.

According to the transfer entropy matrix, A1 and A2 both influence A4. A1 influences A4 more than A2 influences A4. However, the transfer entropy matrix shows the opposite. It



$$A = \begin{bmatrix} 0 & 0 & 0 & 2 & 2 & 2 \\ 0 & 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

μ	sim_1	sim_2	μ	sim_1	sim_2
0.0	0.97	1.32	1.0	0.88	1.17
0.01	0.98	1.34	1.25	0.85	1.13
0.05	0.97	1.32	1.5	0.84	1.13
0.1	0.99	1.34	1.75	0.84	1.12
0.2	0.96	1.30	2.0	0.82	1.10
0.5	0.93	1.24	2.5	0.75	1.04
0.75	0.93	1.24	3.0	0.68	1.00

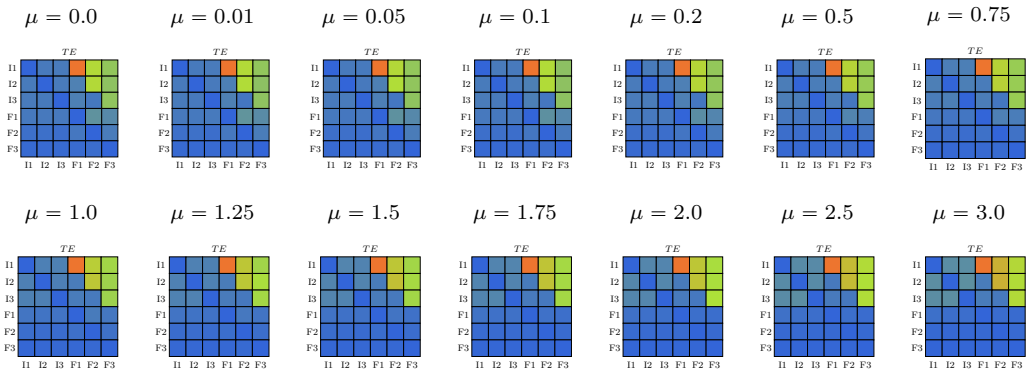


Figure 3.19: Test results for a network of influencers and followers using different values for μ. In the first row, we show a sketch of the network, the weighted adjacency matrix, and the similarity table. In the second row, we show transfer entropy matrices for different values of μ.

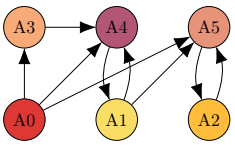
shows that there is more information flow from A2 to A4 than from A1 to A4. A possible explanation for this is that $TE_{A2 \to A4}$ includes the information transfer from $TE_{A1 \to A2}$. Additionally, there is the direct influence from A1, $TE_{A1 \to A4}$. This sums up and results in a higher value.

We can see the same effects of transfer entropy between influencers and reverse transfer entropy, like in the previous section. The similarity between TE and A is $sim_1 = 0.38, sim_2 = 1.52$.

3.6.3 Background activity rate (μ) in a network

This experiment evaluates the influence of μ on transfer entropy in a network. Again, we simulate the actors' behavior over a period of $T = 7 \cdot 24$ hours and report the average transfer entropy over 200 repetitions.

In the first experiment, we use the Influencer-Follower model. There are three influencers, I1, I2 and I3, and three followers, F1, F2, and F3. I1 influences all followers. I2 influences F2 and F3. I3 influences only F3. We use the following parameter configuration: $\lambda = 0.9$, $\gamma = 2$ and $\mu = 0.01$. We expect that the higher the background activity rate μ, the higher the similarity values are and, therefore, the less similar are the transfer entropy matrices



$$A = \begin{bmatrix} 0 & 0 & 0 & 0.25 & 0.25 & 0.25 \\ 0 & 0 & 0 & 0 & 0.25 & 0.25 \\ 0 & 0 & 0 & 0 & 0 & 0.25 \\ 0 & 0 & 0 & 0 & 0.25 & 0 \\ 0 & 0.25 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.25 & 0 & 0 & 0 \end{bmatrix}$$

μ	sim_1	sim_2
0.01	0.10	2.51
0.05	0.92	2.90
0.1	1.24	3.25

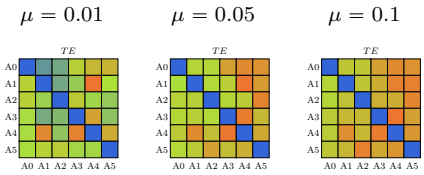


Figure 3.20: Test results for a network using the Coupled model. In the first row, we show a sketch of the network, the weighted adjacency matrix and the similarity table. In the second row, we show transfer entropy matrices for different values of μ.

and the weighted adjacency matrix.

Figure 3.19 shows the outcome of the experiment using the Influencer-Follower model for different values of background activity rate μ.

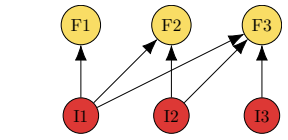
Interestingly, the information flow to F1 is the highest. The information flow to F2 is lower, and the information flow to F3 is the lowest. We explain this with the fact that only one influencer, I1, influences F1. Therefore, I1 can predict the behavior of F1 very well alone. I1 and I2 influence F2. Therefore, both I1 and I2 would be both needed to explain F2's behavior, but we always only look at either $TE_{I1\to F2}$ or $TE_{I2\to F2}$. Both actors partially explain the behavior of F2, but no one can do it perfectly. The same happens for F3, which is influenced by three actors. It can be viewed as the transfer entropy being fragmented due to the three influencers.

Contracting our assumption, the higher the background activity rate μ, the higher the similarity is. This might happen because the higher μ, the less difference is there between $TE_{I1\to F1}$, $TE_{I1\to F2}$, and $TE_{I1\to F3}$ because the fraction of actions which are done independently is increased.

In the second experiment, we use the Coupled model. A sketch of the network is given in Figure 3.20. There are six actors in the network, which all have the same influence strength. A0 influences A3, A4, and A5. A1 influences only A3 and A4, and A2 influences only A5. Additionally, A1 and A4, and A2 and A5 influence each other. A3 influences A4. In the Coupled model we use $\mu = 0.05$ and $\gamma = 0.25$. Figure 3.20 shows the results obtained for the Coupled model.

Generally, the opposite happens here: the higher the background activity rate, the lower the similarity. However, this network has a much more complex structure so we must not make direct comparisons.

The transfer entropy matrices fail to mark the right upper triangle of influence clearly.



$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

γ	sim_1	sim_2
(1.0, 1.0, 1.0)	0.78	1.45
(3.0, 3.0, 3.0)	0.96	1.27
(5.0, 2.0, 0.5)	0.32	0.59
(0.5, 1.0, 1.5)	0.61	1.25

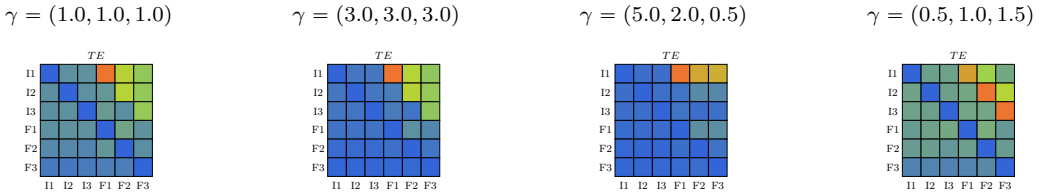


Figure 3.21: Test results for a network of influencers and followers using different values for γ. In the first row, we show a sketch of the network, the weighted adjacency matrix, and the similarity table. In the second row, we show transfer entropy matrices for different values of γ. the influence strength is given as a vector $\gamma = (\gamma_{I1}, \gamma_{I2}, \gamma_{I3})$, denoting the amount of influence each actor has.

The connections between A1 and A4, as well as between A5 and A2, are highlighted. These are the only nodes in the network which influence each other. The transfer entropy between A3 and A4 is also highlighted in many transfer entropy matrices.

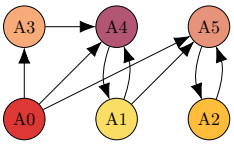
3.6.4 Influence strength (γ) in a network

This experiment investigates how the influence strength γ affects transfer entropy between actors in a network. We observe actors over a period of $T = 7 \cdot 24$ hours and report the average transfer entropy over 200 repetitions.

In the first experiment, we use the Influencer-Follower model. The network is identical to the one represented in Section 3.6.3. We set $\lambda = 0.9$ and $\mu = 0.01$. We expect that the higher the influence in a network, the more similar the transfer entropy matrix is to the weighted adjacency matrix.

Figure 3.21 shows a plot of the network and the adjacency matrix A. Note that here, A is the *unweighted* adjacency matrix. The weights are given by the vector $\gamma = (\gamma_{I1}, \gamma_{I2}, \gamma_{I3})$. However, the similarity is computed between the transfer entropy matrix and the weighted version of the adjacency matrix. We test four different combinations of values for γ, which are listed in the similarity table and as headings for the transfer entropy matrices.

The first two experiments compare transfer entropy using a low and a high influence strength. If the influence is low, more reverse information flow from the followers to the influencers appears. Again we see similar effects regarding the "fragmentation" of transfer



$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

γ	sim_1	sim_2
(0.01, 0.01, 0.01, 0.01, 0.01, 0.01)	1.42	3.69
(0.05, 0.05, 0.05, 0.05, 0.05, 0.05)	1.60	3.84
(0.25, 0.25, 0.25, 0.25, 0.25, 0.25)	0.95	2.97
(0.5, 0.5, 0.5, 0.5, 0.5, 0.5)	0.52	2.67
(0.01, 0.01, 0.01, 0.25, 0.25, 0.25)	2.34	3.67
(0.01, 0.01, 0.25, 0.25, 0.5, 0.5)	2.34	2.67

$\gamma = (0.01, 0.01, 0.01, 0.01, 0.01, 0.01)$ $\gamma = (0.05, 0.05, 0.05, 0.05, 0.05, 0.05)$ $\gamma = (0.25, 0.25, 0.25, 0.25, 0.25, 0.25)$ $\gamma = (0.5, 0.5, 0.5, 0.5, 0.5, 0.5)$ $\gamma = (0.01, 0.01, 0.01, 0.25, 0.25, 0.25)$ $\gamma = (0.01, 0.01, 0.25, 0.25, 0.5, 0.5)$

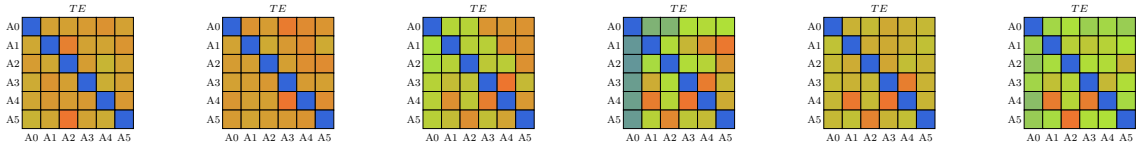


Figure 3.22: Test results for a network of influencers and followers using different values for γ. In the first row, we show a sketch of the network, the weighted adjacency matrix, and the similarity table. In the second row, we show transfer entropy matrices for different values of γ. The influence strength is given as a vector $\gamma = (\gamma_{Ai})$, denoting the amount of influence each actor has.

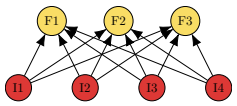
entropy in case a follower is influenced by multiple actors. The similarity according to sim_2 is higher in the case where the influence strength is higher and therefore the experiment confirms our expectations.

Comparing the last two scenarios, (5,2,0.5) and (0.5,1,1.5), some interesting observations can be made. For example, in the first scenario, the transfer entropy matrix does not show any outgoing links from I2 or I3 but only from I1. Information transfer is so small there that it is almost not displayed. In the second scenario, the transfer entropy matrix manages to detect the link from I2 to F2 and from I3 to F3. The amount of reverse transfer entropy is much higher in this case than in the last one.

In the second experiment, we use the Coupled model. We reuse the network from Section 3.6.3. The background activity rate is set to $\mu = 0.05$. Figure 3.22 shows the results obtained for the Coupled model.

Generally, we also observe here that the higher the influence strength is, the more similar is the transfer entropy matrix to the adjacency matrix. Both similarity measures confirm this relation.

The results achieved for the first two test cases ($\gamma = 0.01$ and $\gamma = 0.05$ for all actors) fail to infer the network structure. We assume that due to the complex network structure and the slight influence, connections cannot be recognized as links. The similarity value and the links detected according to the transfer entropy value for the third test, where each actor has the same influence of 0.25, is also very good.



$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 2 & 2 & 2 \\ 0 & 0 & 0 & 0 & 2 & 2 & 2 \\ 0 & 0 & 0 & 0 & 2 & 2 & 2 \\ 0 & 0 & 0 & 0 & 2 & 2 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

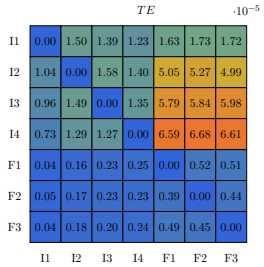


Figure 3.23: Results for an Influencer-Follower network of 7 nodes. The activity level λ is different for each influencer and is sorted in ascending order. Influencer I1 is the least active, and influencer I4 is the most active actor. The left side shows a sketch of the network, in the middle, there is the weighted adjacency matrix while the right side shows the transfer entropy matrix.

The best similarity value is achieved in the test where each actor has the same influence strength of 0.5. This test gives some other interesting insights. First, the influence of all actors to A0 is much lower than the other information flows in the network. Next, the information flow from A0 to A1 and A2 is also marked as very low, and the links between A0 and A3, A4, and A5 are detected correctly. There is transfer entropy between A1 and A2, which originates from both having the same indirect influencer, A0, via their direct influencers A4 and A5. The transfer entropy between A4 and A3 is very high but should not be. Again, this might be because both are influenced by A0.

The last two scenarios describe situations in which influencers do not have the same influence strength. Since A4 and A5 have the strongest influence in the network, $TE_{A4 \to A1}$, $TE_{A4 \to A3}$, and $TE_{A5 \to A2}$ are highlighted.

3.6.5 Influencer activity rate (λ) in a network

In this experiment, we investigate the influence of λ in a network. The observation time frame is $T = 7 \cdot 24$ hours, and we report the average transfer entropy over 200 repetitions at the end of the time frame.

Since the Coupled model does not employ the influencer activity rate λ, we only do our evaluation with the Influencer-Follower model. We generate a network with four influencers and three followers. Every influencer influences every follower and has its own activity rate. We choose $\lambda_{I1} = 0.1, \lambda_{I2} = 0.75, \lambda_{I3} = 1, \lambda_{I4} = 1.4$ such that I1 is the least and I4 is the most active influencer. The parameters μ and γ are set to $\mu = 0.01$ and $\gamma = 2$. In the preliminary study, it was already shown that the highest amount of information transfer is recorded for $\gamma = 2$ in the region of $\lambda \in [0.5, 1]$.

Figure 3.23 shows a sketch of the network, the weighted adjacency matrix, and the transfer entropy matrix. In the experiment, it is observed that the most active actor, I4, is the source of the most information flow. Transfer entropy increases as λ increases. However, the results are slightly misleading. Each of the followers has a certain amount of

input transfer entropy, which means it has a certain amount of individual events, events inspired by I1, events inspired by I2, events inspired by I3, and events inspired by I4. The amount of individual events is always constant at $\mu = 0.01$. The amount of events that are inspired by an influencer is higher if the influencer is more active. In the transfer entropy matrix, we actually see the amount of information flowing from each influencer to each follower. Since I4 is the most active, the information flow is maximized there. We compute the similarity between TE and A with $sim_1 = 0.689, sim_2 = 1.541$.

3.7 Summary

We conclude, that information transfer between actors is affected by the background activity rate (μ), influence strength (γ), and influencer activity rate (λ). We execute experiments with synthetic data generated by Poisson point processes and investigate the influence of those parameters on the outcome of the experiments. The main findings and observations of the synthetic study are:

- Actors below and above a certain activity level should be excluded from analysis because then, transfer entropy provides misleading results.
- For a fixed influencer activity rate λ, transfer entropy increases with the influence strength γ and decreases with the background activity rate μ.
- Normalization of transfer entropy by division through $H(Y_t|Y_{t-1}^{t-k})$ or $\log k$, where k is the number of bins, does not provide further insights useful for analysis. Systematic bias correction seems to only lower transfer entropy by a more or less constant factor. Permutation tests can be used to evaluate how strong the influence between two actors really is. However, the method is too computationally intensive to be applied practically.

In the next section, we conduct an empirical study. In this study, we take into account all the important points above which are learned from the synthetic study.

Chapter 4

Empirical Analysis

In this chapter, we evaluate whether we can utilize transfer entropy to extract meaningful knowledge from real-world datasets. In Section 4.1, we describe the application on a small dataset dealing with water level of rivers and show how to determine in which direction rivers run. In Section 4.2, we use the LFM-1b dataset, which contains listening events created by users of Last.fm. We show how to recover interesting social connections between users of different countries and genres. In Section 4.3, we analyze tweets posted by twitter trolls and investigate the strength of influence of certain hashtags.

4.1 Flow direction of rivers

We first evaluate the concept of transfer entropy on a small dataset, which includes information about the water level of rivers. We focus on four different rivers, Saar, Rhine, Danube, and Mur and cities along those rivers. In each city, there is a stream gaging station which reports the water level of the respective river periodically. We investigate how transfer entropy can be used in connection with the information about water levels in order to determine the direction in which rivers flow.

4.1.1 Description of the dataset

The dataset contains data from two different data sources. First, we take the dataset used by Budhathoki et al. [20]. It provides water levels in centimeters recorded for two German rivers, Saar and Rhine. Measurements were taken every 15 minutes for one month (25.6.2017 - 24.7.2017). The measurement stations are located in Fremersdorf, Hanweiler, and Sanktarnautal for the Saar river and Speyer, Mannheim, Worms, and Mainz for the Rhine river. Second, we create our own dataset from data provided by eHYD[1], an online archive providing various kinds of measurement data for Austrian rivers and lakes. We focus on two large Austrian rivers, Danube and Mur. For the Danube, we consider the measurement stations Kienstock, Korneuburg, Melk, and Thebnerstrasse.

[1] `https://ehyd.gv.at`

```
datetime;graz;mureck;friesach
03.01.1987-00:00;47.0;72.3;62.7
04.01.1987-00:00;38.5;65.0;53.0
05.01.1987-00:00;38.8;56.0;48.6
06.01.1987-00:00;35.8;59.5;45.7
07.01.1987-00:00;38.4;55.9;46.5
08.01.1987-00:00;34.9;57.6;41.7
09.01.1987-00:00;31.3;54.3;37.4
10.01.1987-00:00;32.4;48.7;33.5
11.01.1987-00:00;28.8;55.2;35.6
12.01.1987-00:00;27.4;49.1;33.9
```

```
graz:  [05.01.1987-00:00; 07.01.1987-00:00;
10.01.1987-00:00;]
mureck:  [05.01.1987-00:00; 06.01.1987-00:00;
08.01.1987-00:00; 11.01.1987-00:00;]
friesach:  [07.01.1987-00:00; 11.01.1987-00:00]
```

Figure 4.1: Example of preprocessing applied to a part of the Mur dataset. The left side shows the raw data. The right side shows the preprocessed data. We convert the data into a list of timestamps for each city.

Measurements were taken once a day over a time span of 17 years (7.10.1999 - 31.12.2016), yielding a total of 6296 water level measurements per city. For the Mur, we consider gaging stations in Graz, Mureck, and Friesach. The observation time covers even 29 years, yielding in total, 10956 data points per city.

4.1.2 Data preprocessing

Raw data is given in the form of a list of timestamps associated with the water level in centimeters measured at that time in a specific city. However, the computation of transfer entropy expects a list of timestamps. Therefore, we define the event *Water level rises.* The event happens whenever the water level measured at time T is larger than the water level measured at time $T-1$. $T-1$ is then reported as the timestamp when the event happens. In other words, given timestamps $t0, t1$ and water levels $l0$ measured at $t0$ and $l1$ measured at $t1$, the event occurs at time $t0$ if $l1 > l0$.

An example for the river Mur is given in Figure 4.1. On the left side we see the dates and the time measurements made on those dates. Our conversion results results in a list of timestamps per city, denoting when the event *Water level rises* happens.

The underlying assumption of our experiments is that if the water level rises in one city, the water level will also rise in the next city downstream. Instead, we could have also defined the event *Water level falls.* Table 4.1 shows how many of these events are finally recorded per city per river. The number of timestamps is almost equal for Mur and Danube but differs more for Rhine and especially for Saar.

As a second preprocessing step, we map the timestamps to smaller time units. The concept of transfer entropy does not know the notion of time. It does not bother if we work with seconds, days or even years as the smallest time unit. However, in our dataset, measurements are reported on a regular basis, for example, every day. It is not possible that two *Water level rises* events happen within one day. We define the smallest unit as one day. For example, two events happening at the 1^{st} and 26^{nd} of January would be mapped to the timestamps 1 and 26. This step does not affect the outcome of the experiment, but drastically reduces the computational effort.

These preprocessing steps now allow us to construct a binned random variable. In case the water level rises in the time frame covered by one bin, the random variable will be 1 and 0 otherwise. Then, the transfer entropy between cities $C1$ and $C2$, $TE_{C1\to C2}$, is computed. We assume that in case the river flows from $C1$ to $C2$, $TE_{C1\to C2}$ will be higher than $TE_{C2\to C1}$. If the water level rises in $C1$ it will also rise in $C2$, and not vice versa.

4.1.3 Experiment design

We conduct the same experiment for each river, which is constructed as follows:

1. **Preprocessing:** We apply the preprocessing steps as described in the previous section.
2. **Choice of bins:** The average inter-event time is computed for each city in order to indicate the size of bin widths. We choose four different bin width configurations for each river, which are aligned to the average inter-event time.
3. **Computation of TE:** We compute the transfer entropy between cities for all the different bin width configurations.
4. **Discussion:** In the last step, we provide a detailed discussion of the results. The results are represented using the transfer entropy matrix. Additionally, we provide an intuitive graph in which correct edges are marked in green and incorrect edges are marked in red.

Steps 2-4 are explained in detail for each river in the following sections.

4.1.4 Results for Saar river

The Saar is a river in France and Germany. It is 246km long. The minimal time unit for Saar data is 15 minutes. Figure 4.2 shows the inter-event distribution for all cities. Note that outliers are removed before plotting the histogram using the Tukey outlier test [78] using parameter $k = 10$. The outlier filtering is only done for the illustration. Sanktarnautal has the largest average inter-event time, followed by Fremersdorf and Hanweiler. The overall average inter-event time is $\bar{t} = 1.23$ hours $= 74$ minutes. We round this to $\bar{t} = 75$ minutes.

Saar		**Rhine**		**Mur**		**Danube**	
Hanweiler	153	Speyer	519	Friesach	3953	Melk	2420
Sanktarnual	599	Mannheim	736	Graz	4000	Kienstock	2435
Fremersdorf	994	Worms	599	Mureck	3947	Korneuburg	2452
		Mainz	762			Thebnerstrasse	2425

Table 4.1: The dataset consists of data about four rivers, Saar, Rhine, Mur, and Danube. There are 3 or 4 cities along each river which delivers measurement data about the water level. The number of times in which the water level rises is given together with the city.

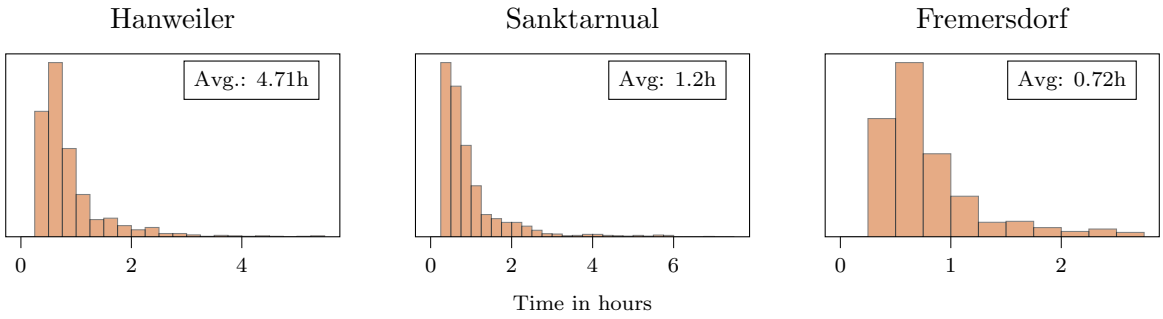


Figure 4.2: Distribution of inter-event times in the Saar dataset. To make the illustration more understandable, outliers are excluded. The overall average is 0.8294 hours.

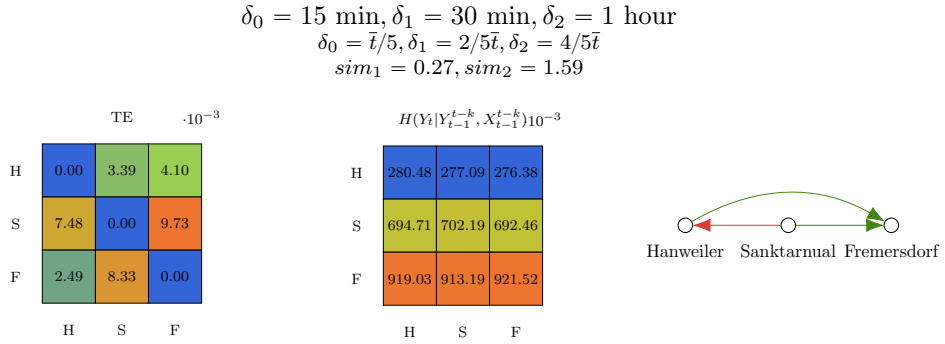


Figure 4.3: Test results for the Saar river using bins of width $\delta_0 = \bar{t}/5, \delta_1 = 2/5\bar{t}, \delta_2 = 4/5\bar{t}$. The first heat map shows the transfer entropy matrix between the cities. The second heat map shows $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k})$. Note that if $X = Y$ then $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k}) = H(Y_t|Y_{t-1}^{t-k})$. The row refers to Y, the column refers to X. The graph shows the edges between the cities based on the transfer entropy matrix. A red edge means an incorrect edge, a green edge means a correct edge. We compute the similarity between the correct adjacency matrix and the transfer entropy matrix.

In the first experiment, we test a few narrow bins. For the most recent history, we set $\delta_0 = 15$ min $= \bar{t}/5$. For the history one step further in the past, we set $\delta_1 = 30$ min $= 2/5\bar{t}$. For the history furthest in the past, we set $\delta_2 = 60$ min $= 4/5\bar{t}$. Every link is detected correctly except the one from Hanweiler to Sanktarnual, as it is shown in Figure 4.3. The second heat map shows the entropy of each stochastic process. Since the bins are chosen very narrow the history of Hanweiler is captured very well, which leads to a small uncertainty in the respective stochastic process. There are many Fremersdorf events, the history is captured less well, and there is high uncertainty. The same thing happens for Sanktarnual, which is why the transfer entropy originating there is higher in general. This leads to an incorrectly inferred link between Sanktarnautal and Hanweiler. Therefore, we need to choose bin widths which capture the histories of Sanktarnual and Fremersdorf better.

The results of the second, third and fourth experiment are shown in Figure 4.4. In

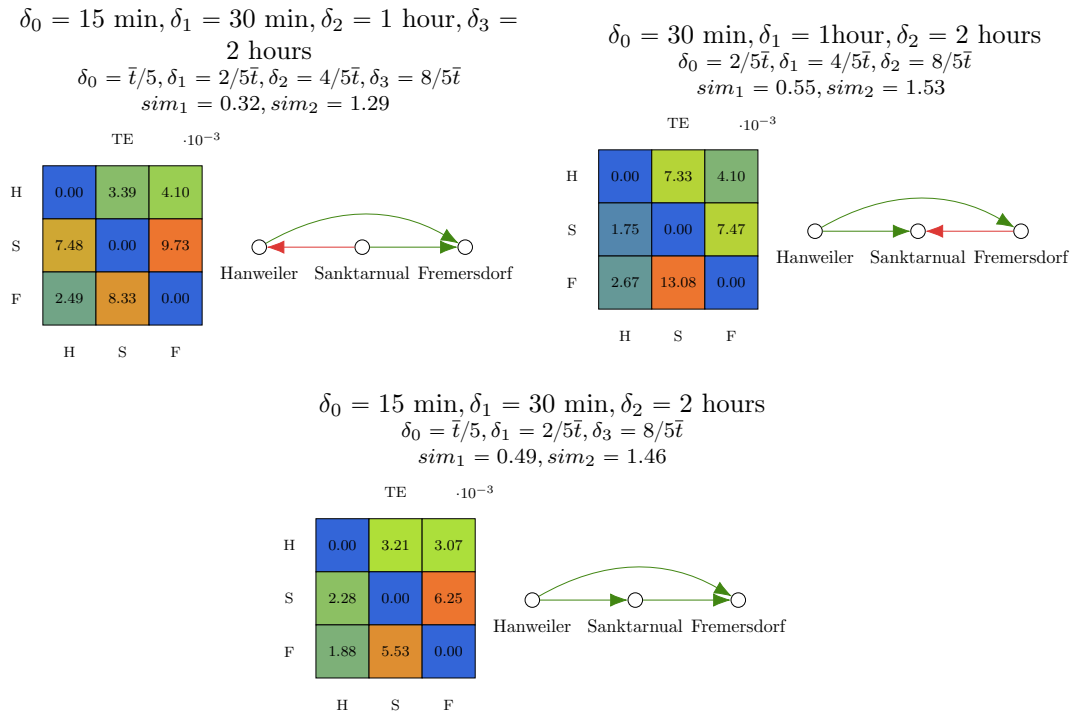


Figure 4.4: Test results for the Saar river using different configurations of bin widths. The first heat map shows the transfer entropy matrix between the cities. The graph shows the edges between the cities based on the transfer entropy matrix. A red edge means an incorrect edge. A green edge means a correct edge. We compute the similarity between the correct adjacency matrix and the transfer entropy matrix.

the second experiment, we choose one more bin, $\delta_3 = 2$ hours $= 8/5\bar{t}$. Again, the link between Hanweiler and Sanktarnual is not detected correctly. Compared to the first experiment, the absolute transfer entropy originating at Hanweiler is now higher. The link between Hanweiler and Sanktarnual is still detected incorrectly. The difference between $TE_{H \to S}$ and $TE_{S \to H}$ became smaller because, due to the additional bin, the history of Sanktarnual can be captured better which leads to a reduction in uncertainty.

The third experiment works with three bins of size 30 minutes, 1 hour, and 2 hours. Now, the link between Hanweiler and Sanktarnual is detected correctly, but the link between Sanktarnual and Fremersdorf is wrong. This might happen because the first bin is too wide for Fremersdorf data, which has a very small inter-event time.

In the last experiment, we try to combine bin widths such that the insights from the first and the third experiment are realized optimally. We observe in the first experiment that bins should be wide, and in the third experiment that the first bin must be 15 minutes. Therefore, we set the bin widths to 15 minutes, 30 minutes, and 2 hours. The transfer entropy matrix is able to detect all links correctly.

For each of the transfer entropy matrices, we do a comparison with the adjacency matrix,

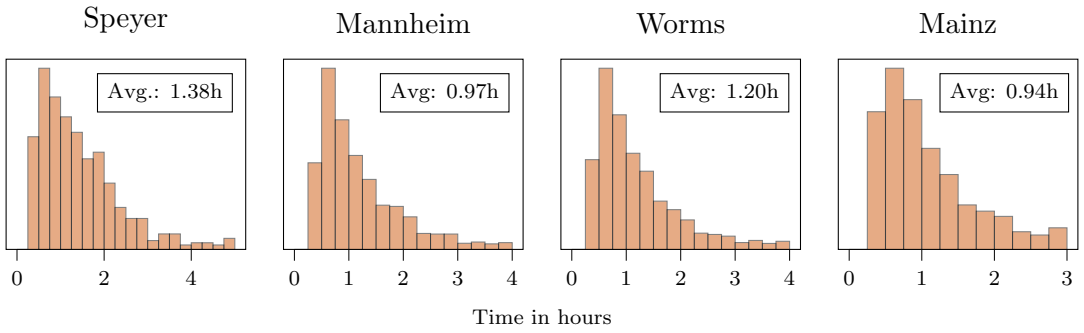


Figure 4.5: Distribution of inter-event times in the Rhine dataset. To make the illustration more understandable, outliers are excluded. The overall average is 1.1 hours.

$$\delta_0 = 15 \text{ min}, \delta_1 = 2 \text{ hours}, \delta_2 = 2 \text{ hours}, \delta_3 = 2 \text{ hours}$$
$$\delta_0 = 1/4\bar{t}, \delta_1 = 2\bar{t}, \delta_2 = 2\bar{t}, \delta_3 = 2\bar{t}$$
$$sim_1 = 0.89, sim_2 = 1.96$$

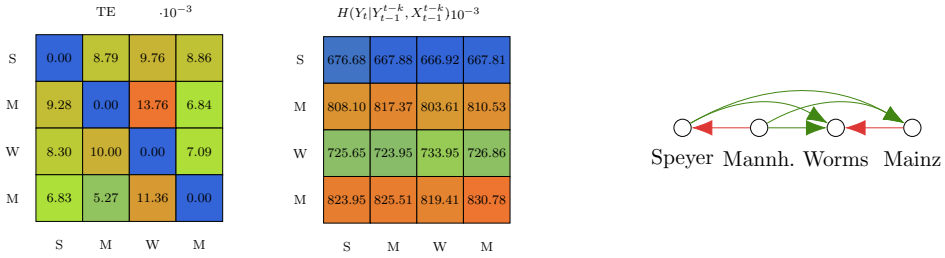


Figure 4.6: Test results for the Rhine river using bins $\delta_0 = \bar{t}/5, \delta_1 = 2/5\bar{t}, \delta_2 = 4/5\bar{t}$. The first heat map shows the transfer entropy matrix between the cities. The second heat map shows $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k})$. Note that if $X = Y$ then $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k}) = H(Y_t|Y_{t-1}^{t-k})$. The row refers to Y, the column refers to X. The graph shows the edges between the cities based on the transfer entropy matrix. A red edge means an incorrect edge, a green edge means a correct edge. We compute the similarity between the correct adjacency matrix and the transfer entropy matrix.

which is an upper triangular matrix. The similarity values do not indicate correct results very well. For example, the similarity value in the fourth experiment is higher than in the second experiment, which means "less similar". The reason is that in the fourth experiment, the link between Fremersdorf and Sanktarnautal is less clear than in the second experiment. The difference between the absolute transfer entropy values is higher.

4.1.5 Results for Rhine river

The Rhine river originates in Switzerland, and flows through Germany to the North Sea. It is 1200km long. The minimal time unit for Saar data is 15 minutes. The inter-event distribution for all measurement stations is shown in Figure 4.5. As described in the previous section, outliers are removed for the diagram using the Tukey test with $k = 3$. Speyer has the largest average inter-event time with 1.38 hours, followed by Worms with 1.2 hours and Mannheim and Mainz with around 0.95 hours. The total overall average

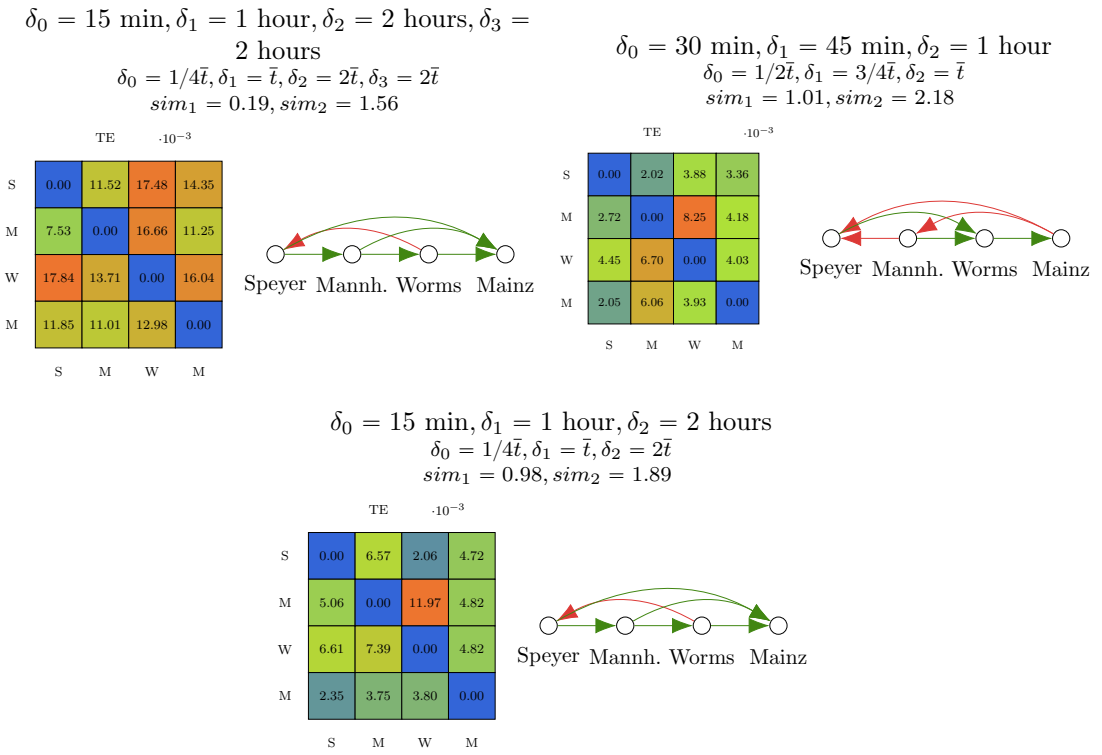


Figure 4.7: Test results for the Rhine river using different configurations of bin widths. The first heat map shows the transfer entropy matrix between the cities. The graph shows the edges between the cities based on the transfer entropy matrix. A red edge means an incorrect edge. A green edge means a correct edge. We compute the similarity between the correct adjacency matrix and the transfer entropy matrix.

inter-event time is $\bar{t} = 1.1$ hours. Again, we round this to $\bar{t} = 1$ hour.

The results of the first experiment are shown in Figure 4.6. We observe a strong correlation between the inter-event times and the entropy $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k})$, like in the previous example. Speyer has the lowest entropy and the highest average inter-event time. Mainz and Mannheim both have similar average inter-event times and a similar, higher entropy. The link between Speyer and Mannheim, as well as Worms and Mainz, is detected incorrectly. This might be a hint that the bin widths are too wide because we fail to detect one-hop links, but can easily detect multi-hop links.

The results for the following experiments are shown in Figure 4.7. In the second experiment, we want to investigate what happens if we exchange the second bin, δ_1, with a smaller bin (1 hour). All one-hop links are detected correctly, but now we fail to recognize the multi-hop link between Worms and Speyer. This confirms our assumption about too wide bin widths of the first last experiment.

The third experiment is done with bins of width $(\bar{t}/2, 3/4\bar{t}, \bar{t})$. Two multi-hop links are not recognized as well as one one-hop link. This means that bin widths are chosen entirely

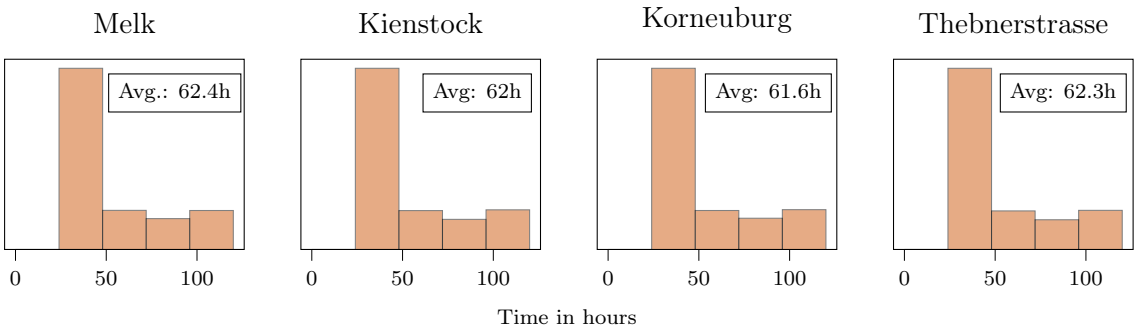


Figure 4.8: Distribution of inter-event times in the Danube dataset. To make the illustration more understandable, outliers are excluded. The overall average is 62.06 hours.

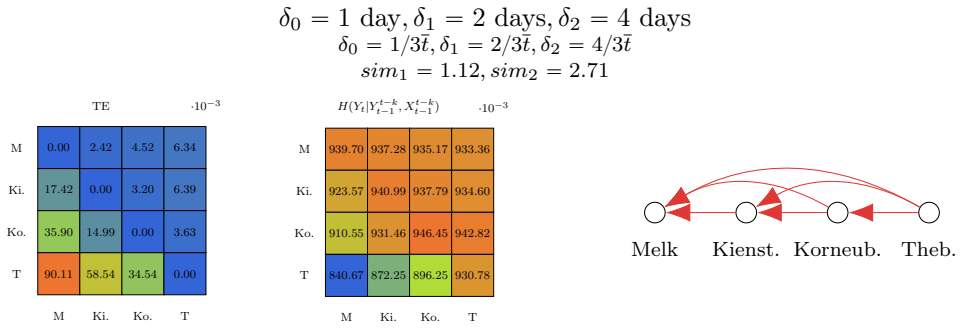


Figure 4.9: Test results for the Danube river using bins $\delta_0 = \bar{t}/5, \delta_1 = 2/5\bar{t}, \delta_2 = 4/5\bar{t}$. The first heat map shows the transfer entropy matrix between the cities. The second heat map shows $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k})$. Note that if $X = Y$ then $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k}) = H(Y_t|Y_{t-1}^{t-k})$. The row refers to Y, the column refers to X. The graph shows the edges between the cities based on the transfer entropy matrix. A red edge means an incorrect edge, a green edge means a correct edge. We compute the similarity between the correct adjacency matrix and the transfer entropy matrix.

wrong since the binning process cannot capture the history of any city sufficiently. The first bin is too wide, which is why Speyer is involved in two wrong links. Overall, the bins do not observe enough time at once, which is why the multi-hop link between Mainz and Mannheim is wrong.

In the last experiment, we choose the same bins as in the second experiment but remove the last bin. This experiment gives an interesting insight as it shows the same relations but in general, a lower transfer entropy. This confirms the theoretical finding we had about bin widths in the first section.

4.1.6 Results for Danube river

The Danube is the second-longest river in Europe and flows through 10 countries. The Danube dataset is different from the Saar and Rhine dataset because the measurement data is not delivered every 15 minutes but every 24 hours. The minimal time unit is

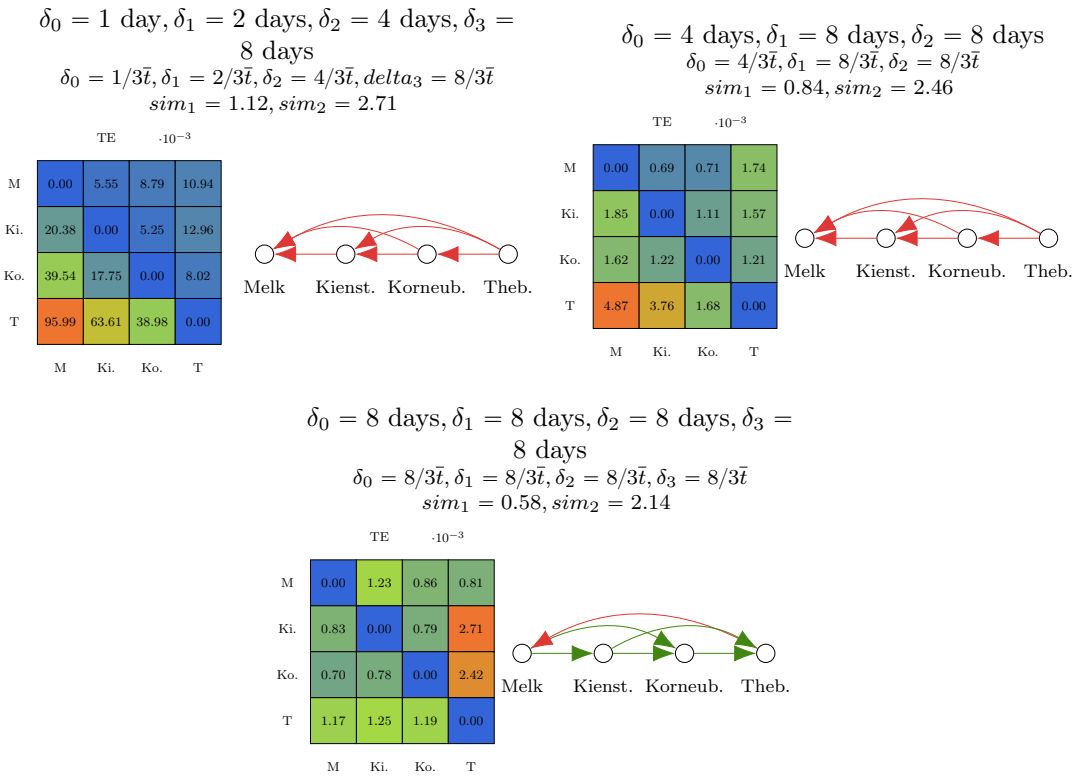


Figure 4.10: Test results for the Danube river using different configurations of bin widths. The first heat map shows the transfer entropy matrix between the cities. The graph shows the edges between the cities based on the transfer entropy matrix. A red edge means an incorrect edge. A green edge means a correct edge. We compute the similarity between the correct adjacency matrix and the transfer entropy matrix.

therefore 24 hours. The average inter-event times are much higher than in the first two datasets. Additionally, we expect that detecting correct flow directions will be more difficult since the information we have is less detailed. Figure 4.8 shows the inter-event time distribution for all cities along the Danube river. Outliers are excluded using the Tukey method with $k = 2$. The total average inter-event time is 62.06 hours, which we again round to $\bar{t}$ =72 hours (3 days). Melk has the largest inter-event time with 62 hours.

In the first experiment, we chose the bins $(1/3\bar{t}, 2/3\bar{t}, 4/3\bar{t})$. The results can be seen in Figure 4.9. Not a single link is detected correctly, and the transfer entropy matrices, which should be an upper triangular matrix, is a lower triangular matrix. The transfer entropy matrix shows a high information flow from the last station, Thebnerstrasse to Melk, a lower to Kienstock, and an even lower to Korneuburg. The matrix showing $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k})$ looks different from what we saw in the last section since the average inter-event times are almost equal for all cities. The highest transfer entropy appears at the point where we have the least uncertainty, between Thebnerstrasse and Melk.

The results for the following experiments are shown in Figure 4.10. The second exper-

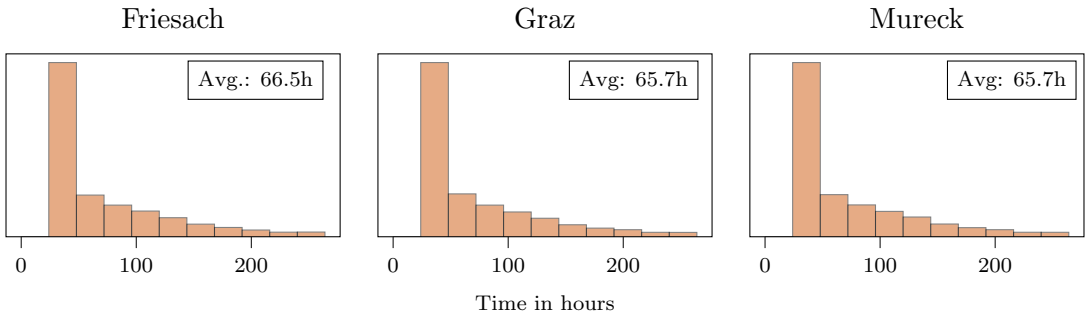


Figure 4.11: Distribution of inter-event times in the Mur dataset. To make the illustration more understandable, outliers are excluded. The overall average is 66.26 hours.

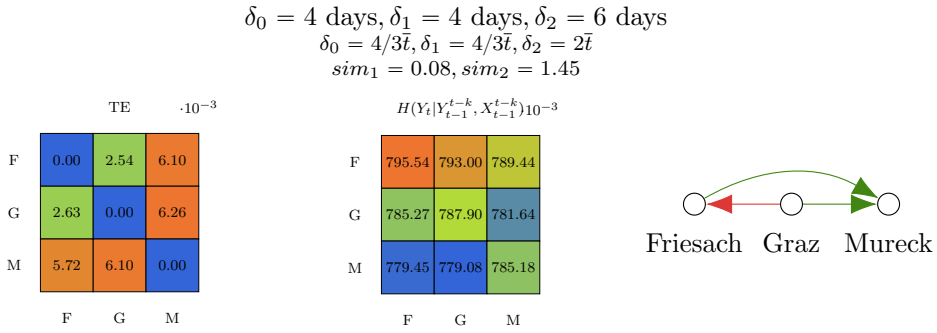


Figure 4.12: Test results for the Mur river using bins $\delta_0 = \bar{t}/5, \delta_1 = 2/5\bar{t}, \delta_2 = 4/5\bar{t}$. The first heat map shows the transfer entropy matrix between the cities. The second heat map shows $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k})$. Note that if $X = Y$ then $H(Y_t|Y_{t-1}^{t-k}, X_{t-1}^{t-k}) = H(Y_t|Y_{t-1}^{t-k})$. The row refers to Y, the column refers to X. The graph shows the edges between the cities based on the transfer entropy matrix. A red edge means an incorrect edge, a green edge means a correct edge. We compute the similarity between the correct adjacency matrix and the transfer entropy matrix.

iment repeats the first, but with one more bin. The results are the same as in the first experiment. We assume that bins are not wide enough. The third experiment works with bins $(4/3\bar{t}, 8/3\bar{t}, 8/3\bar{t})$, and still, no links are detected correctly. However, we observe that the transfer entropy above the transfer entropy matrix' diagonal increases.

As expected, we need to choose wider bins. In the fourth experiment, we work with bins of size $(8/3\bar{t}, 8/3\bar{t}, 8/3\bar{t})$. Now, all the links are detected correctly except one. Notably, information flow to Thebnerstrasse is very high compared to the other values in the transfer entropy matrix. In the previous scenarios, we saw that information flow *from* Thebnerstrasse was very high.

4.1.7 Results for Mur river

The Mur river is 450km long and runs through Austria and Slovenia. The minimal time unit for is 24 hours. Figure 4.11 shows the inter-event time distribution for all cities.

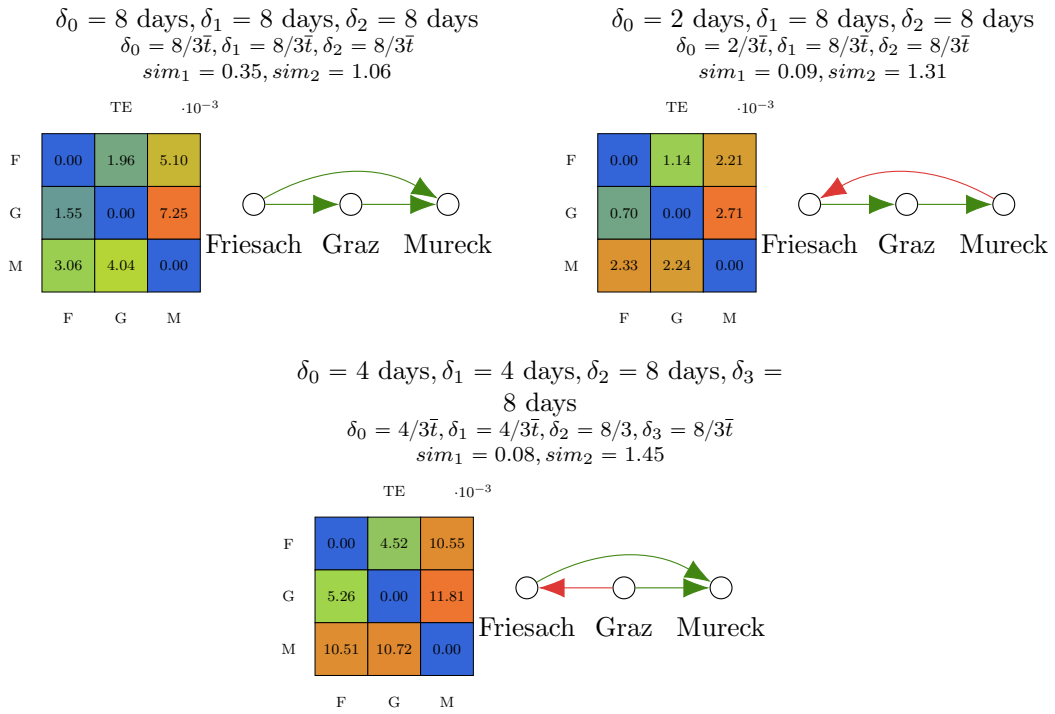


Figure 4.13: Results for test runs on the Mur river for different bin widths. A green arrow means a correct edge while a red arrow denotes an incorrect one.

Outliers are removed by setting $k = 3$ in the Tukey test. Friesach has the largest average inter-event time, followed by Graz and Mureck. The overall average inter-event time is 66.26 hours. We round this to $\bar{t} = 72$ h $= 3$ days.

The first experiment is done with bins $(4/3\bar{t}, 4/3\bar{t}, 4/3\bar{t}$. The results are shown in Figure 4.12. Friesach has the highest uncertainty in its history, which might be related to the slightly higher average inter-event time. Interestingly, the information flow to Mureck is higher than the information flow to Graz. We already saw in previous experiments with networks. Actors who are influenced by multiple actors accumulate their information transfer. Information flow to Mureck is very high because it "includes" the information flow from Friesach to Mureck and from Graz to Mureck.

The results for the following experiments are shown in Figure 4.13. In the second experiment, we use wider bins of equal width of 8 days. All links are detected correctly, and we still see a big difference between the transfer entropy to Graz and the transfer entropy to Mureck.

In the third experiment, we narrow down the first bin to 2 days. The transfer entropy matrix fails to detect the multi-hop link between Friesach and Mureck. We hypothesize that this happens because, in total, the computation window of the binned random variable is reduced to 18 days instead of 24 days and that therefore, the window is too small to analyze the two-hop connection in a meaningful way.

The last experiment works with four bins of size $(4/3\bar{t}, 4/3\bar{t}, 8/3\bar{t}, 8/3\bar{t})$. Just like in the first experiment, the link between Friesach and Graz is not detected correctly. This might hint that bins are chosen too narrow.

4.1.8 Summary

The experiment with the river dataset allows us to get a first estimation of how we can use transfer entropy in order to extract useful information from a tiny dataset. The cities can be mapped to actors or stochastic processes. In the beginning, we define the event *Water level rises* and compute the timestamps on which the event occurs.

This makes it possible to compute the average inter-event time, which we use to choose appropriate bin widths. Then, we compute transfer entropy between the cities along a river.

In general, determining the correct flow direction works better for Rhine and Saar than for Danube and Mur. The reason is that we get measurement data for the first two every 15 minutes and the last two every 24 hours.

However, several other factors affect the quality of the results, including:

- Obstacles along the river: Smaller distances make it easier to detect the correct flow directions because there are fewer obstacles that can change the water level, for example, embankment dams or reservoirs.
- Tributaries: Rivers which lead into a larger river can influence downstream cities. For example, the river Sulm ends up in the Mur between Graz and Mureck. If the water level of the Sulm rises, this might also trigger events in Mureck, which are not caused by Graz.
- Forks: Similar to tributaries, a river can fork and continue as a smaller side river and a bigger main river.

These experiments should give first insights into the topic of transfer entropy in empirical studies. In the next section, we have a look at a larger dataset.

4.2 Last.fm

In the following we analyze the meaning and behavior of transfer entropy using a larger dataset, the LFM-1b dataset. It contains listening events of users on the platform Last.fm. First, we examine the information flow between users in certain countries who listen to a specific artist. Second, we investigate whether users who listen to Country, R&B/Hip-Hop or Rock influence each other.

4.2.1 Description of the dataset

The LFM-1b dataset [70] contains data retrieved from Last.fm. Last.fm is a website which offers an online music database and a sophisticated recommendation system for their users. Users can share which music they listen to or meet other people with the same music taste. Last.fm does not offer music streaming. It only tracks which music users listen to. Data is sent to Last.fm when users listen to music using certain apps on their smartphones, e.g., Spotify or Apple Music, or certain programs on their desktop devices [9].

Last.fm offers an API for researchers to build their own datasets [10]. Schedl [70] worked with *tags* in order to retrieve meaningful data from Last.fm to build the LFM-1b dataset. On the platform, users can tag music. This means they assign terms to music which describe the track, album or artist. For example, they might assign *rock* to a song by Queen. Schedl focused on those tags which were used by users most. Each song has an artist associated with it, and each artist has fans. The users represented by the dataset is a randomly chosen subset of these fans, yielding around 120 000 users in total.

Table 4.2 gives an overview of all the objects and fields to describe the objects in the dataset. Each user is characterized by a unique identifier, country, age and gender. As shown in [70], less than 50% of users specify their country or gender. Only about 40% of all users provide their exact age. If a user listens to a song, a listening event is created. A listening event is described by a timestamp, the user, the track, the artist, and the album. Furthermore, for each artist and album, there is a unique identifier and a name.

Object	Fields	Number
User	`(userid, country, age, gender)`	120.322
Artist	`(artistid, artistname)`	3.190.371
Album	`(albumid, albumname)`	15.991.038
Track	`(trackid, trackname, artistid)`	32.291.134
Listening event	`(userid, artistid, albumid, trackid, timestamp)`	61.534.450

Table 4.2: Description of objects in the LFM-1b dataset, the fields associated with them and the number of data items per object.

Tracks are characterized by a name and the identifier of the respective artist.

The timestamps of listening events cover a range of 20 months, beginning in January 2013 and ending in August 2014. The dataset contains 61 million listening events, 32 million tracks, 15 million albums and 3 million artists.

4.2.2 Countries as stochastic processes

In the first experiment we evaluate how users of different countries influence each other. We focus on the five countries with the most users, as listed in [70]. The group of users which does not specify their country (N/A) is excluded. The countries can be described as a stochastic process with certain properties, just like the actors in the synthetic analysis. The list of timestamps is created by taking all listening events of users of a particular country.

We execute the experiment with five different artists, Adele, Rihanna, Bruno Mars, Taylor Swift, and Drake. These artists are selected based on the rating of the Billboard magazine. The Billboard magazine is a famous music magazine in the United States which publishes charts, rating music artists in different genres. The artists are the five top artists listed in the Billboard Year-End Charts 2012 and 2013 [7, 8]. We select all listening events created by users living in a certain country, for example, Germany, who listen to Bruno Mars. The result is a list of timestamps per country that can directly be binned and then be used in the transfer entropy computation.

The first step is to determine the bin width for the analysis. Therefore, the average and median inter-event time in minutes is computed for each artist and each country, as shown in Table 4.3. The second to last column of the table shows the total average and median inter-event times. In the last column, we write down the bin widths which are used for the experiment.

We choose the first bin according to the median inter-event time. For the second bin, we use $3\bar{t}$, which results in a time between 2 hours and 5 hours. For the third bin, we choose $16\bar{t}$, which results in a time between 12 hours and 25 hours. One should keep in

Artist	**US**		**RU**		**DE**		**UK**		**PL**		**Total**		**Bins**
	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	Avg.	Med.	(in minutes)
Adele	31	4	80	5	82	5	79	5	25	4	46	4	$\delta_0 = 4, \delta_1 = 46 \cdot 3, \delta_2 = 46 \cdot 16$
Rihanna	40	5	97	5	106	14	76	7	41	4	60	5	$\delta_0 = 5, \delta_1 = 60 \cdot 3, \delta_2 = 60 \cdot 16$
Bruno Mars	62	121	252	18	140	14	108	7	63	4	95	7	$\delta_0 = 7, \delta_1 = 95 \cdot 3, \delta_2 = 95 \cdot 16$
Taylor Swift	36	5	266	5	249	7	74	5	200	5	86	5	$\delta_0 = 5, \delta_1 = 86 \cdot 3, \delta_2 = 86 \cdot 16$
Drake	25	4	351	5	271	13	66	5	141	5	65	4	$\delta_0 = 4, \delta_1 = 65 \cdot 3, \delta_2 = 65 \cdot 16$

Table 4.3: Inter-event times in minutes for each country per artist. *Avg.* denotes the mean, *Med.* denotes the median of the raw data. In the last column we show the bins we choose for the experiment. The first bin refers to the overall median, the second and third bin to the average inter-event time.

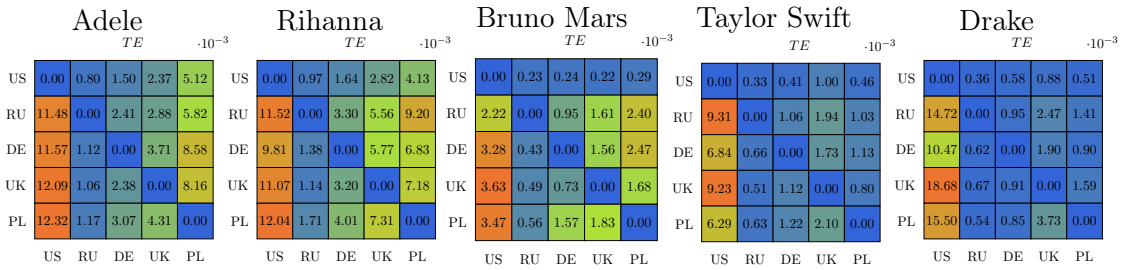


Figure 4.14: Transfer entropy between users from different countries listening to a specific artist. We used raw (unfiltered) data for the analysis, but we should have removed too active countries. The reason for the high amount of transfer entropy towards the US is the time difference.

mind that by choosing the first bin not equal to the minimal time unit (1 minute), we will not be able to capture the exact occurrence of every single event.

The total number of listening events per country per artist is sown in Table 4.4. The US has the most listening events for each artist, except for Adele, where Poland has more than the US. Since all artists are English speaking artists from either the US, UK or Canada, we expect that the US and possibly the UK have a strong influence on the other countries.

We evaluate the transfer entropy between the five countries for each artist. First, we compute the raw transfer entropy on unfiltered data. The results are shown in Figure 4.14.

First, we observe that the influence of all countries to the US is very strong. This does not confirm our initial assumption that the US would be the country with the strongest influence. The reason for this is the time difference between the US and all other European countries. The dataset specifies timestamps in UTC format. Assume that Adele releases her new album *21* on January 24, 2011, at 1 PM GMT (Greenwich Mean Time). In London, people start to listen to the music immediately, and there are a lot of listening events. The UTC time is be January 24, 2011, at 1 PM. In Los Angeles, it is 5 AM local time, and many people will still be asleep, then go to work and then start to listen to Adele - much later than in the UK.

Artist	US	RU	DE	UK	PL
Adele	124.363	43.298	41.661	49.445	137.584
Rihanna	116.829	41.453	44.684	63.637	116.186
Bruno Mars	36.753	8.218	16.087	21.643	33.877
Taylor Swift	112.397	12.252	13.296	50.416	16.265
Drake	143.220	7.560	9.621	46.618	19.754

Table 4.4: Number of listening events per country per artist.

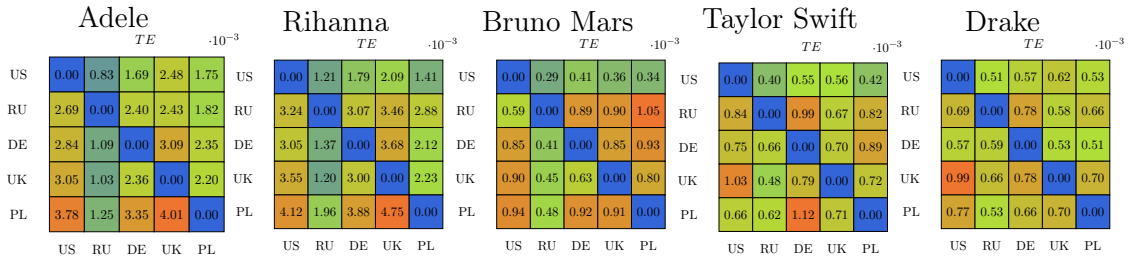


Figure 4.15: Transfer entropy between users from different countries listening to a specific artist. We used filtered data by choosing random subsets of pairwise equal length for each computation.

The problem is that most listening events will occur at different times of the day in the US and in Europe, which makes it hard to choose bins which appropriately cover this behavior. In both countries, users might listen to music before going to bed. Therefore, most listening events happen at around 8 PM GMT in Europe but at 3 AM UTC in the US.

For Adele and Rihanna, we observe a high information transfer towards Poland. Poland has a huge number of listening events in these cases. Since we are working with raw, unfiltered data, this result is not very surprising. The synthetic analysis suggests to filter out too active and inactive actors.

In the second part of the experiment, we apply this filtering. We cannot exclude actors from the analysis since actors are countries in this case, and we want to analyze all five countries. Therefore, we apply a filter that works with random subsets of the data. We choose the country with the smallest number of timestamps. For example, for the artist Rihanna this would be Russia with 41453 timestamps. We select 41453 timestamps from all other countries randomly. This forms the new, filtered history of the countries. Then, we can compute the transfer entropy between them and ensure that all actors are equally active. We repeat this random selection 100 times and then report the average transfer entropy. The results can be seen in Figure 4.15.

We can extract similarity patterns of influence from these five cases. For example, Poland has a high influence on the UK when listening to Adele, Rihanna, or Bruno Mars. Russia has a high influence on Poland when listening to Bruno Mars or Taylor Swift. The US seems to influence the UK most. However, randomly shuffling data does not solve the timezone problem, which is why we do not see transfer entropy between the US and other countries at all.

4.2.3 Genres as stochastic processes

In the second experiment, we evaluate how users of a specific genre influence each other. As genres we select R&B/Hip-Hop, Country and Rock, three top genres suggested by the Billboard magazine. Each genre is represented by ten tracks. Half of it were the

top 5 tracks in the respective genre in the Year-End charts of 2012, the other half in the Year-End charts of 2013 [1, 2, 3, 4, 5, 6]:

- R&B/Hip-Hop: *Thrift Shop - Macklemore*, *Blurred Lines - Robin Thicke*, *Can't Hold Us - Macklemore*, *Suit & Tie - Justin Timberlake*, *Holy Grail - Jay Z*, *Love On Top - Beyonce*, *Climax - Usher*, *Lotus Flower Bomb - Wale*, *The Motto - Drake*, *Mercy - Kanye West*
- Country: *Cruise - Florida Georgia Line, Wagon Wheel - Darius Rucker, Boys 'round Here - Blake Shelton, Crash My Party - Luke Bryan, I Want Crazy - Hunter Hayes, Time Is Love - Josh Turner, You Don't Know Her Like I Do - Brantley Gilbert, Why Ya Wanna - Jana Kramer, Cowboys And Angels - Dustin Lynch, (Kissed You) Good Night - Gloriana*
- Rock: *Radioactive - Imagine Dragons, Ho Hey - The Lumineers, Royals - Lorde, Safe And Sound - Capital Cities, Sail - AWOLNATION, Lonely Boy - The Black Keys, These Days - Foo Fighters, Burn It Down - Linkin Park, Somebody That I Used To Know - Gotye, Gold On The Ceiling - The Black Keys*

In other worlds, if a user listens to one of the ten tracks of one genre, he or she listens to the specific genre. For example, if a user listens to *Royals* by Lorde, he or she listens to Rock. Only users from the US who listen to all three genres are considered. We analyze how many users who listen to a specific genre are inspired to also listen to another genre. For example, we analyze how likely it is that users who listen to rock will, therefore, also listen to Country.

Table 4.5 shows the inter-event times for each genre. Rock has the highest average inter-event time with 618 minutes, which is about 10 hours, and the most listening events, followed by R&B/Hip-Hop and Country. For our computations we can now derive the bin widths $\delta_0 = 154$ minutes, $\delta_1 = 556/2$ minutes, $\delta_2 = 556 \cdot 2$ minutes.

Figure 4.16 shows the results of the experiment. The first transfer entropy matrix shows the raw transfer entropy. We observe that R&B/Hip-Hop users and Rock users influence each other and also have a strong influence on Country users. However, Country users do not influence Rock and R&B/Hip-Hop users that much. This might be because Country music is much less popular. According to a study executed by Nielsen Music in 2015 [11], the most famous music genres, according to sold CDs, vinyl, and online streams, are Rock (30%) followed by R&B/Hip-Hop (22%). Additionally, from a very subjective point of

Genre	Average	Median	Number of listening events
Country	528	133	2518
R&B/Hip-Hop	514	161	2955
Rock	618	164	3200
Total	556	154	

Table 4.5: Average and median inter-event times in minutes for each genre. The last column shows the number of listening events per genre. We choose the bin widths $\delta_0 = 154, \delta_1 = 556/2, \delta_2 = 556 \cdot 2$ according to median and average.

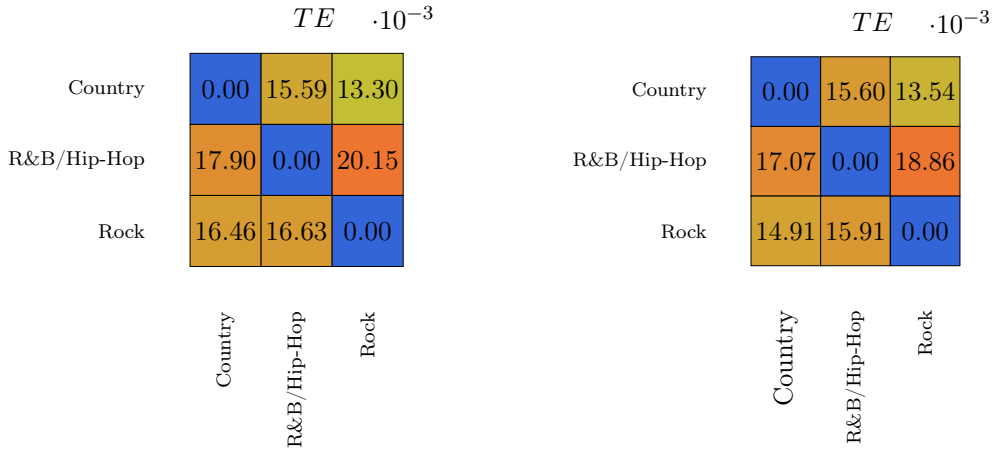


Figure 4.16: Influence of US-users listening to music from a specific genre. The first heat map shows the raw transfer entropy computed between the genres. The second heat map shows the transfer entropy computed between random subsets of the data of equal length. The influence of users listening to Country is not that big.

view, a large fraction of Rock and R&B/Hip-Hop music can also be classified as Pop music, which has a popularity of 19% too. Country only has a popularity of 10%, and most Country songs can be clearly classified as Country. The border to other genres is more clear.

Figure 4.16 also shows the results of the tests executed with random subsets of equal length. The genre with the least amount of timestamps is Country. We randomly select 2518 timestamps from genres Rock and R&B/Hip Hop, compute the pairwise transfer entropy and report the average. In general, the outcome is similar to the first case. The transfer entropy from R&B/Hip-Hop to Rock is even stronger.

4.2.4 Summary

The LFM-1b experiments allow us to test transfer entropy in a larger dataset to analyze the behavior of users in a social network, Last.fm. In the first part, we view countries as stochastic processes and investigate how users from the US, Russia, Germany, the UK, and Poland influence each other. The main problems in this analysis were time zone differences in the data and massive differences in the number of available listening events per country. In the second part, we view music genres as stochastic processes. A genre is defined by a set of 10 songs. We take users from the US who listen to all three genres and examine their influence. The outcome of the experiment showed that users from the Country genre follow users from the other two genres.

Another issue of the analysis is the choice of bin widths. We choose the widths in ascending order, starting with the median inter-event time. The median inter-event time is not the smallest time unit in the analysis. In a situation where a lot of listening events happen within a small time frame, we are not able to detect all the listening events. In the next section we show how the result changes when using the smallest possible or the median inter-event time for the first bin width.

4.3 Twitter trolls

In this section, we conduct an analysis of tweets posted by Russian and Iranian trolls on Twitter using transfer entropy. The central question is if there is influence between hashtags, in other words, if users using hashtag A can influence users to use hashtag B. In the first experiment, we analyze the 100 most used hashtags and their influence. In the second experiment, we filter out hashtags that are used too often and not often enough and analyze their influence.

The second goal of this analysis is to determine how important the choice of the first bin width is. In the previous sections, we have already shown that the bin widths considerably affects the result, and that the average inter-event time is a good indicator of the ideal bin widths. The Twitter study conducted by Ver Steeg et al. [80] mentions that a possible strategy is to use wider bin widths for less recent history. They also mention using the finest resolution for the first bin. In the following, we want to evaluate the results we get when using bins according to the average inter-event time. In the first run of each experiment, we set $\delta_0 = 1$ minute, the finest resolution. In the second run of each experiment, we set δ_0 to the median of inter-event times.

4.3.1 Description of the dataset

In 2018, Twitter released 9 million tweets, which were posted by around 3800 Russian troll accounts between 2013 and 2018 [36]. Internet trolls are persons who post rude and offensive messages on social media sites. Very often, the messages are provocative or aim at starting arguments with other users [30]. Russia allegedly uses troll farms or troll factories to conduct political campaigns on Twitter, who post fake news and try to manipulate the opinion of other users [55, 75]. The accounts used by trolls are very often fake accounts. For example, in 2016, Russian trolls were suspected to be employed in the US elections to manipulate people's opinions. Other cases include the Brexit referendum in 2016, the elections in France 2017, and the 2017 Catalan independence referendum [34, 39, 45, 76].

In total, the dataset comprises 9 million tweets posted by 3841 accounts. The data is given in CSV format. Each line represents a tweet, and each tweet is described by the time when it was posted, the list of hashtags used, and several other attributes. Since we focus on the hashtags, we filter out all tweets which do not use hashtags. This results in 2 million tweets referring to 152892 different hashtags.

The further preprocessing steps, which are applied to the data, are very similar to the ones applied in the river dataset, as described in Section 4.1.2. In general, we start with a list of tweets using certain hashtags. This list is transformed to a list of hashtags. Each hashtag list is associated with a list of tweet timestamps. The timestamp mapping, as it is described in 4.1.2, is also applied in this case since tweets. The minimal time unit is one minute, since the tweet timestamps are also given in minutes. We do not know about the exact second when a tweet was posted. Generally, one can view the hashtag as a stochastic process and the tweet times associated with the hashtag as the history of

the process.

4.3.2 Experiment design

The experiment design is also very similar to the one applied when working with the river dataset (see Section 4.1.3). In this case, we only have two experiments. The first experiment aims at analyzing the influence of the 100 most used hashtags. The second experiment aims at analyzing the influence of hashtags which are used not too rarely and not too often. The experiments are designed in 5 phases:

1. **Preprocessing:** We apply the preprocessing steps as described in the previous section to the test data.
2. **Filtering:** In the first experiment, we exclude hashtags that are not among the 100 most used. In the second experiment, we first compute the average number of tweets per hashtag. Then, we exclude all hashtags which have fewer tweets then the average number. In the next step, we compute the first and third quartile, $q1$, and $q3$ from the number of tweets per hashtag. We exclude all hashtags which have less than $q1 + 0.6 \cdot (q3 - q1)$ or more than $q3$ associated tweets.
3. **Choice of bins:** The average inter-event time is computed for the resulting datasets in order to indicate the size of bin widths.

   In the first run of each experiment, we set the first bin width δ_0 to 1 minute. In the second run of each experiment, we choose the first bin width according to the median inter-event time. The second and third bins are always chosen according to the average inter-event time.
4. **Computation of TE:** We compute the transfer entropy between all hashtags.
5. **Discussion:** We provide a detailed discussion of the results by listing the hashtags with the most information outflow and the most information inflow. Additionally, we list the links with the most transfer entropy.

4.3.3 100 most used hashtags

In the first experiment, we investigate the 100 most used hashtags, *i.e.*, the 100 hashtags, which have the longest timestamp list.

In order to determine the bin-widths, the average-inter event time needs to be computed. Figure 4.17 shows the distribution of inter-event times excluding outliers. Most tweets have a very short average inter-event time. The median is 18 minutes. However, there are some extreme outliers. The average inter-event time is 190 minutes. We select $\delta_1 = 190$ and $\delta_2 = 190 \cdot 2$ for the widths of the second and third bin. In the first part we use $\delta_0 = 1$ and in the second part we use $\delta_0 = 18$.

The distribution of the number of timestamps per hashtag after selecting the 100 most used hashtags is shown in Figure 4.18. Most of the hashtags have a low number of timestamps. The most used hashtag has a timestamp list of length 198000, *i.e.*, occurs

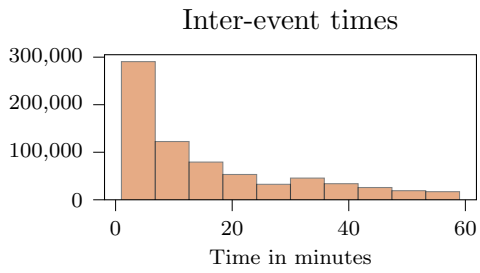


Figure 4.17: Distribution of inter-event times of all Tweets of the 100 most used hashtags. For the purpose of easier illustration, outliers not shown. The total average inter-event time is 190 minutes, the median inter-event time is 18 minutes.

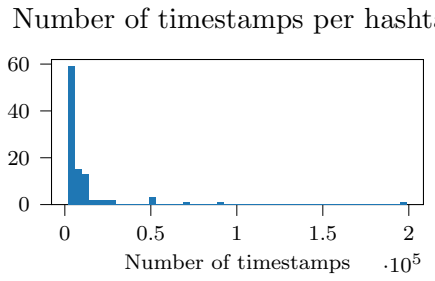


Figure 4.18: Number of tweets/timestamps per hashtags for the 100 most used hashtags. Outliers are not excluded. The most used hashtag appears 198684 in a Tweet, the 100th most used hashtag 2088 times. On average, there are 11533 tweets per hashtag.

in 198000 tweets. The least used hashtag, which is the 100th most used in total, occurs in 2088 tweets. On average, one hashtag has a timestamp list length of 11500.

Results for $(\delta_0 = 1, \delta_1 = 190, \delta_2 = 190 \cdot 2)$. In Table 4.6 we report the top 10 hashtags according to the total outdegree TE. These can be seen as the hashtags which influence others the most. Apparently, the most influential hashtags are Украина, which means *Ukraine* in English, политика, *politics*, and политика, *news*, which all indicate political content. The only hashtag which is not directly related to politics is Футбол, *football*.

100 most used hashtags $(\delta_0 = 1, \delta_1 = 190, \delta_2 = 190 \cdot 2)$						
Hashtag	**Total Out-degree** TE	**Hashtag**	**Total In-degree** TE	**Hashtag Src.**	**Hashtag Dst.**	TE
Украина[1]	0.02265	новости[2]	0.10653	Футбол[3]	новости[2]	0.00486
политика[4]	0.02225	USA	0.06675	культура[6]	новости[2]	0.00474
Футбол[3]	0.02205	local	0.05469	Зенит[7]	новости[2]	0.00454
РФ[8]	0.02100	sports	0.05296	авто[5]	новости[2]	0.00429
Образ России[9]	0.02085	Россия[10]	0.04588	россия[10]	новости[2]	0.00415
world	0.02082	love	0.04433	games	новости[2]	0.00413
crime	0.02071	politics	0.04339	Украина[1]	новости[2]	0.00337
культура[6]	0.02048	America	0.04104	true	USA	0.00329
Россия[10]	0.02043	Украина[1]	0.03841	ОбразРоссии[9]	новости[2]	0.00313
celebs	0.01980	НевскиеНовости[11]	0.03411	РФ[8]	новости[2]	0.00308

Table 4.6: Top 10 of hashtags among the 100 most used hashtags. We use the bins $\delta_0 = 1, \delta_1 = 190, \delta_2 = 190 \cdot 2$ for our computations. The first two columns show the 10 hashtags with the largest cumulated outgoing TE, the second two columns show the largest cumulated ingoing TE, the last three columns show the links with the largest TE.

The hashtags with the most indegree TE are those users felt most inspired to use. They are also all very political. The top hashtag is новости, *news*. Interestingly, the most influential links, which are shown in the right part of the table, all lead to the hashtag *news*, except one.

In the Appendix, in Figures A.2 and A.3, we can see the influence network extracted from the total in- and outdegree TE. The top hashtags are marked in orange. In both cases, we observe a slight clustered structure. It is noticeable that the hashtag language in one such cluster is always either English or Russian, which indicates that trolls rarely mix languages in their tweets. In both figures, see one cluster using Russian hashtags related to politics and one cluster using English hashtags seemingly related to sports and celebrities.

Results for ($\delta_0 = 18, \delta_1 = 190, \delta_2 = 190 \cdot 2$). In Table 4.7 we present the 10 hashtags with the highest outdegree TE. More or less the same hashtags as in the previous section are reported, only the ordering is slightly different. *TopNews*, *entertainment* and *SanJose* are new hashtags. The hashtags with the most indegree influence are also very similar. We see the same result about the links with the most TE. The destination hashtag is always новости, *news*.

In the Appendix, in Figures A.4 and A.5 we can see a similar structure as before. However, the graphs appear to be less clustered.

100 most used hashtags ($\delta_0 = 18, \delta_1 = 190, \delta_2 = 190 \cdot 2$)

Hashtag	**Total Out-degree** TE	**Hashtag**	**Total In-degree** TE	**Hashtag Src.**	**Hashtag Dst.**	TE
Россия[10]	0.24725	новости[2]	0.73928	Футбол[3]	новости[2]	0.03087
Невские[11] Новости	0.24061	local	0.44543	культура[6]	новости[2]	0.03078
crime	0.22715	USA	0.37958	Зенит[7]	новости[2]	0.02932
Украина[1]	0.22574	Невские[11] Новости	0.33501	games	новости[2]	0.02852
политика[4]	0.22528	politics	0.32874	авто[5]	новости[2]	0.02590
РФ[8]	0.22070	sports	0.32237	россия[10]	новости[2]	0.02332
world	0.21893	America	0.32041	Украина[1]	новости[2]	0.02099
TopNews	0.21827	Россия[10]	0.30895	ОбразРоссии[9]	новости[2]	0.01988
entertainment	0.21757	кино[12]	0.26901	Россия[10]	новости[2]	0.01919
SanJose	0.21722	Украина[1]	0.26509	музыка[13]	новости[2]	0.01784

Table 4.7: Top 10 of hashtags among the 100 most used hashtags. We use the bins $\delta_0 = 18, \delta_1 = 190, \delta_2 = 190 \cdot 2$ for our computations. The first two columns show the 10 hashtags with the largest cumulated outgoing TE, the second two columns show the largest cumulated ingoing TE, the last three columns show the links with the largest TE.

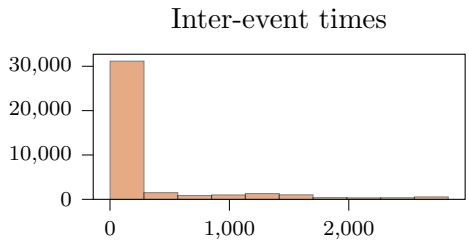


Figure 4.19: Distribution of inter-event times of all Tweets of the quartile-filtered hashtags. For the purpose of easier illustration, outliers not shown. The total average inter-event time is 10638 minutes, the median inter-event time is 44 minutes.

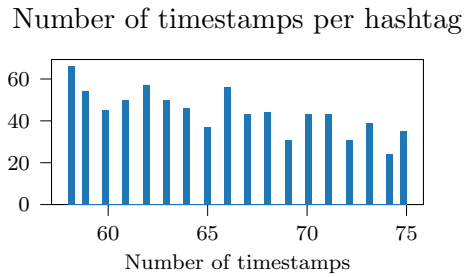


Figure 4.20: Number of tweets/timestamps per hashtags for the quartile-filtered hashtags. Outliers are not excluded. The most used hashtag appears 76 times in a Tweet, the least used hashtag 57 times. On average, there are 65 tweets per hashtag. In total, we analyze 794 hashtags.

Summary. In general, we can argue that changing the bin widths of the first bin does not have a crucial influence on the result. In both cases, we see more or less the same results, and it is easy to identify very influential hashtags.

However, if one wants to know which one of two hashtags is more influential, the task becomes harder. For example, the first results suggest that Украина is more influential than Россия, but the second results suggest the opposite. The question is also if there influence is different in reality. Therefore, if the goal of the experiment is to compare the influence of two hashtags, we suggest to repeat the experiment with several different bin widths and then combine the results.

4.3.4 Quartile-filtered hashtags

In the second experiment, we investigate hashtags that were filtered by the number of tweets they occur in, which is the same as the number of timestamps associated. First, we compute the average number of timestamps per hashtag and exclude all hashtags which have fewer timestamps than that. Then, we compute the first and third quartile and exclude all hashtags which have less than $q1+0.6\cdot(q3-q1)$ and more than $q3$ timestamps. We needed to increase the lower boundary from simply $q1$. This is because further investigations showed that $q1$ would allow hashtags which have not enough timestamps to provide meaningful analysis results.

In order to determine the bin widths, we need to investigate the average inter-event time. Figure 4.19 shows the distribution of inter-event times excluding outliers. The distribution indicates that the vast majority of tweets has a short average inter-event time. The distribution is less balanced than it was in the first experiment. The reason for this is that now we allow hashtags which have a shorter number of timestamps. A small number of timestamps in a certain time frame always lead to a longer average

inter-event time than a large number of timestamps. The total average inter-event time is 10638 minutes. The median is 44 minutes.

Figure 4.20 shows the number of timestamps per hashtag. Quartile-filtering of data leads to the lower bound of 57 timestamps and the upper bound of 76 timestamps. For our experiment we choose the bin widths ($\delta_0 = 1, \delta_1 = 10638/2, \delta_2 = 10638$) and ($\delta_0 = 44, \delta_1 = 10638/2, \delta_2 = 10638$). In total, we analyze 794 different hashtags.

Results for ($\delta_0 = 1, \delta_1 = 10638/2, \delta_2 = 10638$). The most influential hashtags by means of outgoing transfer entropy are shown in Table 4.8. We see that most of the hashtags are now English hashtags because, apparently, Russian trolls prefer to post in Russian. Most of the topics are political, for example *corruption* and *motivation*. Others might indicate propaganda or controversial topics, for example *abortion* or *blackgirlmagic*. *BBC*, a big news company, is the second most influential hashtag.

The hashtags with the most ingoing transfer entropy are also political in the first place. We assume that *MerylStreep* was used by trolls in a political discussion, too. Streep, who is an actress, gave a speech at the 2016 Democratic National Convention to support Hillary Clinton [12]. *Nowplaying* is used in a tweet by a user to announce which music the user is currently listening to.

The links with the most transfer entropy give very interesting insights. Two of them originate at ЦРУ, *CIA*. The first destination hashtag is *ShockingmurderinAtlanta*. Several sources claim that this hashtag indicates fake news spread by Russian trolls about a black,

Quartile-filtered hashtags ($\delta_0 = 1, \delta_1 = 10638/2, \delta_2 = 10638$)

Hashtag	**Total Out-degree** TE	**Hashtag**	**Total In-degree** TE	**Hashtag Src.**	**Hashtag Dst.**	TE
Israeli	0.00548	LiberalCrybabies	0.00707	Meet	Nowplaying	0.00167
BBC	0.00545	ПутинНаш$_{w14}$ Президент	0.00683	NoJustice NoPeace	HowToSpot ASocialist	0.00012
abortion	0.00537	TrumpWasRight	0.00679	StopKillingUs	MomIn5Words	0.00011
LGBTQ	0.00533	Nowplaying	0.00678	UltimOra	MomIn5Words	0.00011
motivation	0.00533	PrayForFlorida	0.00666	ЦРУ^{15}	Shockingmurder inAtlanta	0.00011
ЦРУ^{15}	0.00525	Stockholm	0.00666	ЦРУ^{15}	AskNacks	0.00011
women	0.00518	MerylStreep	0.00648	Еврови$_{16}$ дение	RahmEmanuelOut	0.00011
blackgirl magic	0.00512	blackart	0.00640	ShutItDown	HowToSpot ASocialist	0.00011
Corruption	0.00511	writingtips	0.00632	sport	MomIn5Words	0.00011
corruption	0.00511	Article50	0.00623	JayZ	ItsUnacceptableTo	0.00011

Table 4.8: Top 10 of hashtags among quartile-filtered hashtags. We use the bins $\delta_0 = 1, \delta_1 = 10638/2, \delta_2 = 10638$ for our computations. The first two columns show the 10 hashtags with the largest cumulated outgoing TE, the second two columns show the largest cumulated ingoing TE, the last three columns show the links with the largest TE.

unarmed woman who was shot by the police [23, 26, 59]. The hashtag *MomIn5Words* is used to construct a tweet consisting of five words that a mom would say to her children. In theory, it could be possible that trolls use this hashtag to make seemingly off-topic, non-political posts occasionally to lower the chance to be reported and banned from Twitter.

In the Appendix, Figures A.6 and A.7 show the influence network around the analyzed hashtags. In Figure A.6 we see a small cluster in which all influence originates at *blackgirlmagic*. This hashtag was introduced by HuffPost, an online newspaper, to "celebrate the beauty, power and resilience of Black women" [85]. In Figure A.7 we see that a lot of information flow originates and destines at the hasthag *SanctuarySewer*, which is apparently a term from a video game.

Results for ($\delta_0 = 44, \delta_1 = 10638/2, \delta_2 = 10638$). In this step, we increase the size of the first bin. The results are presented in Table 4.9. We observe that about 50% of the hashtags, which are most influential, are the same as in the first step with $\delta_0 = 1$. The most influential hashtag is seemingly *motivation*, which has a very broad meaning. The hashtag *PeoplesMonday* is used in tweets about police brutality against black people.

The top hashtag, according to indegree transfer entropy, is *Nowplaying*, but the others all indicate controversial topics. Interestingly, four out of ten hashtags refer to black people who were shot by the police between 2016 and 2017. This includes *DeborahDanner*, *JordanEdwards*, *TyreKing* and *TerenceCrutcher*. *FreeRicky* is about a black man who was mistakenly imprisoned for 39 years in the US [72].

Quartile-filtered hashtags ($\delta_0 = 44, \delta_1 = 10638/2, \delta_2 = 10638$)

Hashtag	**Total Out-degree TE**	**Hashtag**	**Total In-degree TE**	**Hashtag Src.**	**Hashtag Dst.**	TE
motivation	0.13226	Nowplaying	0.25716	Meet	Nowplaying	0.00167
chicago	0.13111	FidelCastro	0.18934	Kindle	Nowplaying	0.00165
Corruption	0.12911	DeborahDanner	0.16828	PeoplesMonday	Nowplaying	0.00158
LGBTQ	0.12775	JordanEdwards	0.16275	Красавицы_17 России	Nowplaying	0.00153
women	0.12720	OC	0.14952	Красавицы17 России	Nowplaying	0.00153
Catholic	0.12669	MerylStreep	0.14218	ПолныйКонтакт[18]	Nowplaying	0.00149
Clintons	0.12648	TyreKing	0.13793	dreams	Nowplaying	0.00144
Peoples Monday	0.12641	новосибирск[19]	0.13770	Nizza	Nowplaying	0.00143
Romance	0.12571	TerenceCrutcher	0.13747	antifa	Nowplaying	0.00143
abortion	0.12518	FreeRicky	0.13720	Антимасонское20 движение	Nowplaying	0.00140

Table 4.9: Top 10 of hashtags among quartile-filtered hashtags. We use the bins $\delta_0 = 44, \delta_1 = 10638/2, \delta_2 = 10638$ for our computations. The first two columns show the 10 hashtags with the largest cumulated outgoing TE, the second two columns show the largest cumulated ingoing TE, the last three columns show the links with the largest TE.

The links with the most information transfer are completely different from what we saw in the first test. They all lead to the hashtag *Nowplaying*. However, the link with the most information transfer is the same as in the previous example. Similar to *MomIn5Words*, we think this is again used by trolls to distract from their political messages.

Figures A.8 and A.9 show the influence network we extract from the most influential and influenced hashtags. Both show a more or less clustured structure.

Summary. In the second experiment, the results we got for both bin widths are similar but more different than in the first experiment. One reason is that we analyze more hashtags, which is to say 794 instead of 100. This means we analyze in total around 630.000 links instead of 10.000. The "chance" that we find more influential links in the second run is therefore higher. Another reason is that the inter-event times are, by far, not evenly distributed. The average is 10.000 minutes, while the median is 44 minutes. In the first experiment, this difference was smaller.

4.3.5 Summary

In the first experiment, a larger bin width for the first bin doe snot change the outcome significantly. The top hashtags according to outdegree and indegree transfer entropy are more or less the same, only the order is different. The links with the most TE all destine in the same hashtag.

In the second experiment, the results are more different which we think is due to the higher amount of analyzed hashtags and links.

We conclude that the structure and characteristics of the data affects whether the choice of bins makes a big or a small difference. Significant deviations in the average inter-event times can lead to distorted results.

Chapter 5

Conclusion

In this thesis, we investigated the use of transfer entropy to analyze social networks. We worked with datasets consisting of data about multiple actors, where each actor is described by a list of timestamps. A timestamp indicates the occurrence of an event. We showed how to identify influentials and to extract information networks from such datasets using the concept of transfer entropy.

Few researchers have addressed the study of transfer entropy applied to synthetic data in detail. In this thesis, we introduce two data generation processes, the Influencer-Follower model and the Coupled model. Both models can be tuned by several parameters, including the activity rate and strength of influencers and the fraction of events executed by followers independently. We provide an in-depth, pairwise comparison of those parameters and analyze how they affect transfer entropy. The comparison is made in situations where two actors exist as well as in networks of actors. These experiments revealed that it could be problematic to evaluate transfer entropy between two actors whose activity levels differ too much. Additionally, we found that bin widths have to be chosen with care and that the average inter-event time of actors' actions provides an informative basis for their choice.

Several normalization methods were studied in this work, including normalization by the follower's entropy and normalization by the number of bins. Additionally, we investigated a method to correct the systematic bias introduced by the estimation of entropies from probability distributions of binned random variables. However, none of these methods turned out beneficial for further computations. We also studied permutation tests and found them helpful in determining the true influence between two stochastic processes. We did not apply this approach to further experiments since it was too computationally intensive.

We applied the insights of our synthetic analysis to three different empirical datasets. The river dataset contains the water level of four European rivers, measured in different cities regularly. We showed that the flow direction of these rivers could be determined using transfer entropy. The LFM-1b dataset contains millions of listening events of users

on Last.fm. A listening event is described by a timestamp, the user, and which track the user listens to. We investigate whether users from different countries influence each other and find that different time zones can be a problem. In a further experiment, we examine the influence of users between different genres and see that users listening to Country music are often inspired by users listening to Rock or R&B/Hip-Hop. The third dataset we investigated was a dataset comprising tweets of Russian trolls. The aim of this analysis was to find out about the influence of specific hashtags. We especially focused on differences in the results when using a narrow or wide width for the first bin, which accounts for the most recent history. We deduce that the differences are negligible since the top 10 most influential hashtags were nearly the same, independent of the width of the first bin. However, when one wants to compare the influence of two hashtags directly, the choice of bin widths is important.

Bibliography

[1] *Billboard year-end charts country songs 2012.* https://www.billboard.com/charts/year-end/2012/hot-country-songs. Retrieved on Octover 11, 2019.

[2] *Billboard year-end charts country songs 2013.* https://www.billboard.com/charts/year-end/2013/hot-country-songs. Retrieved on Octover 11, 2019.

[3] *Billboard year-end charts r&b/hip-hop songs 2012.* https://www.billboard.com/charts/year-end/2012/hot-r-and-and-b-hip-hop-songs. Retrieved on Octover 11, 2019.

[4] *Billboard year-end charts r&b/hip-hop songs 2013.* https://www.billboard.com/charts/year-end/2013/hot-r-and-and-b-hip-hop-songs. Retrieved on Octover 11, 2019.

[5] *Billboard year-end charts rock songs 2012.* https://www.billboard.com/charts/year-end/2012/hot-rock-songs. Retrieved on Octover 11, 2019.

[6] *Billboard year-end charts rock songs 2013.* https://www.billboard.com/charts/year-end/2013/hot-rock-songs. Retrieved on Octover 11, 2019.

[7] *Billboard year-end charts top artists 2012.* https://www.billboard.com/charts/year-end/2012/top-artists. Retrieved on October 10, 2019.

[8] *Billboard year-end charts top artists 2012.* https://www.billboard.com/charts/year-end/2013/top-artists. Retrieved on October 10, 2019.

[9] *last.fm - frequently asked questions.* https://cbsi.secure.force.com/lastfm/articles/LastFM/What-is-scrobbling?retURL=%2Flastfm%2Fapex%2Fknowledgehome_lfm%3Freferer%3Dlastfm.com&popup=false&categories=LastFM%3ALastFM_Eng&template=template_lastfm&referer=lastfm.com&data=&cfs=LFM. Retrieved on September 24, 2019.

[10] *Last.fm web services.* https://www.last.fm/api/. Retrieved on September 24, 2019.

[11] *What are the most popular music genres in america?* https://www.digitalmusicnews.com/2016/04/07/most-popular-music-genres-america/. Retrieved on Octover 11, 2019.

[12] N. Al-Sibai, *Transcript of meryl streep's dnc speech calls for the "grit and grace" that hillary clinton embodies.* https://www.bustle.com/articles/175088-transcript-of-meryl-streeps-dnc-speech-calls-for-the-grit-and-grace-that-hillary-clinton-embodies. Retrieved on October 25, 2019.

[13] A. Baddeley, I. Bárány, and R. Schneider, *Spatial point processes and their applications*, Stochastic Geometry: Lectures Given at the CIME Summer School Held in Martina Franca, Italy, September 13–18, 2004, (2007), pp. 1–75.

[14] E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts, *Everyone's an influencer: quantifying influence on twitter*, in Proceedings of the fourth ACM international conference on Web search and data mining, ACM, 2011, pp. 65–74.

[15] A.-L. Barabási, *Network science*, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 371 (2013), p. 20120375.

[16] L. Barnett, A. B. Barrett, and A. K. Seth, *Granger causality and transfer entropy are equivalent for gaussian variables*, Physical review letters, 103 (2009), p. 238701.

[17] T. L. Bauer, R. Colbaugh, K. Glass, and D. Schnizlein, *Use of transfer entropy to infer relationships from behavior*, in Proceedings of the Eighth Annual Cyber Security and Information Intelligence Research Workshop, ACM, 2013, p. 35.

[18] J. Borge-Holthoefer, N. Perra, B. Gonçalves, S. González-Bailón, A. Arenas, Y. Moreno, and A. Vespignani, *The dynamics of information-driven coordination phenomena: A transfer entropy analysis*, Science advances, 2 (2016), p. e1501158.

[19] A. Bray and F. P. Schoenberg, *Assessment of point process models for earthquake forecasting*, Statistical science, (2013), pp. 510–520.

[20] K. Budhathoki and J. Vreeken, *Causal inference on event sequences*, in Proceedings of the 2018 SIAM International Conference on Data Mining, SIAM, 2018, pp. 55–63.

[21] A. Caliò and A. Tagarelli, *Complex influence propagation based on trust-aware dynamic linear threshold models*, Applied Network Science, 4 (2019), p. 14.

[22] M. Cha, H. Haddadi, F. Benevenuto, and K. P. Gummadi, *Measuring user influence in twitter: The million follower fallacy*, in fourth international AAAI conference on weblogs and social media, 2010.

[23] A. Chen, *The agency*. `https://www.stopfake.org/en/the-agency/`. Retrieved on October 25, 2019.

[24] J. N. Cooper and C. D. Edgar, *Transfer entropy in continuous time*, arXiv preprint arXiv:1905.06406, (2019).

[25] T. M. Cover and J. A. Thomas, *Elements of information theory*, John Wiley & Sons, 2012.

[26] A. K. Cronin, *Power to the People: How Open Technological Innovation is Arming Tomorrow's Terrorists*, Oxford University Press, 2019.

[27] J. M. de Assis and F. M. de Assis, *Estimation of transfer entropy between discrete and continuous random processes*, Journal of Communication and Information Systems, 33 (2018).

[28] P. O. S. V. de Melo, C. Faloutsos, R. Assunção, and A. A. F. Loureiro, *The self-feeding process: a unifying model for communication dynamics in the web.*, WWW, 13 (2013), p. 22nd.

[29] M. Dhamala, *Granger causality: theory and applications to neuroscience data*. `https://event.unitn.it/brandyschool2019/Dhamala%20Tutorial.pdf`. Retrieved on October 17, 2019.

[30] C. Dictionary, *Definition of 'troll'*. `https://www.collinsdictionary.com/dictionary/english/troll`. Retrieved on October 23, 2019.

[31] R. Diestel, *Graph Theory, 4th Edition*, vol. 173 of Graduate texts in mathematics, Springer, 2012.

[32] D. Easley, J. Kleinberg, et al., *Networks, crowds, and markets*, vol. 8, Cambridge university press Cambridge, 2010.

[33] L. Egghe and R. Rousseau, *Introduction to informetrics: Quantitative methods in library, documentation and information science*, Elsevier Science Publishers, 1990.

[34] E. Ferrara, *Disinformation and social bot operations in the run up to the 2017 french presidential election*, First Monday, 22 (2017).

[35] A. Ferraz Costa, Y. Yamaguchi, A. Juci Machado Traina, C. Traina Jr, and C. Faloutsos, *Rsc: Mining and modeling temporal activity in social media*, in Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2015, pp. 269–278.

[36] V. Gadde and Y. Roth, *Enabling further research of information operations on twitter*. `https://blog.twitter.com/official/en_us/topics/company/2018/enabling-further-research-of-information-operations-on-twitter.html`. Retrieved on October 23, 2019.

[37] M. Girvan and M. E. Newman, *Community structure in social and biological networks*, Proceedings of the national academy of sciences, 99 (2002), pp. 7821–7826.

[38] J. Golbeck, *Analyzing the social web*, Newnes, 2013.

[39] Y. Gorodnichenko, T. Pham, and O. Talavera, *Social media, sentiment and public opinions: Evidence from# brexit and# uselection*, tech. rep., National Bureau of Economic Research, 2018.

[40] B. Gourévitch and J. J. Eggermont, *Evaluating information transfer between auditory cortical neurons*,

Journal of neurophysiology, 97 (2007), pp. 2533–2543.

[41] L. Guo, E. Tan, S. Chen, X. Zhang, and Y. E. Zhao, *Analyzing patterns of user content generation in online social networks*, in Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2009, pp. 369–378.

[42] M. Hamada, *Conditional and joint entropy.* `http://web-ext.u-aizu.ac.jp/~hamada/IT/L5-IT.pdf`. Retrieved on October 25, 2019.

[43] D. Helic, *Mathematics of networks.* https://courses.isds.tugraz.at/dhelic/netsci/math.pdf. Retrieved on Octover 12, 2019.

[44] K. Hlaváčková-Schindler, M. Paluš, M. Vejmelka, and J. Bhattacharya, *Causality detection based on information-theoretic approaches in time series analysis*, Physics Reports, 441 (2007), pp. 1–46.

[45] J. Im, E. Chandrasekharan, J. Sargent, P. Lighthammer, T. Denby, A. Bhargava, L. Hemphill, D. Jurgens, and E. Gilbert, *Still out there: Modeling and identifying russian troll accounts on twitter*, arXiv preprint arXiv:1901.11162, (2019).

[46] J. Kastrenakes, *Twitter keeps losing monthly users, so it's going to stop sharing how many.* `https://www.theverge.com/2019/2/7/18213567/twitter-to-stop-sharing-mau-as-users-decline-q4-2018-earnings`. Retrieved on October 18, 2019.

[47] E. Katz and F. Paul, *Lazarsfeld (1955), personal influence*, The Part Played by People in the Flow of Mass Communication. New York, (1955).

[48] H. P. Keeler, *Notes on the poisson point process*, Weierstrass Inst., Berlin, Germany, Tech. Rep, (2016).

[49] H. Kwak, C. Lee, H. Park, and S. Moon, *What is twitter, a social network or a news media?*, in Proceedings of the 19th international conference on World wide web, AcM, 2010, pp. 591–600.

[50] L. Liu, B. Qu, B. Chen, A. Hanjalic, and H. Wang, *Modelling of information diffusion on social networks with applications to wechat*, Physica A: Statistical Mechanics and its Applications, 496 (2018), pp. 318–329.

[51] X. Ma, C. Li, J. Bailey, and S. Wijewickrema, *Finding influentials in twitter: a temporal influence ranking model*, arXiv preprint arXiv:1703.01468, (2017).

[52] D. J. MacKay and D. J. Mac Kay, *Information theory, inference and learning algorithms*, Cambridge university press, 2003.

[53] R. D. Malmgren, J. M. Hofman, L. A. Amaral, and D. J. Watts, *Characterizing individual communication patterns*, in Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2009, pp. 607–616.

[54] R. D. Malmgren, D. B. Stouffer, A. E. Motter, and L. A. Amaral, *A poissonian explanation for heavy tails in e-mail communication*, Proceedings of the National Academy of Sciences, 105 (2008), pp. 18153–18158.

[55] O. Nadelnyuk, *How russian "troll factory" tried to effect on ukraine's agenda. analysis of 755 000 tweets.* `https://voxukraine.org/longreads/twitter-database/index-en.html`. Retrieved on October 23, 2019.

[56] M. G. Nejad, D. L. Sherrell, and E. Babakus, *Influentials and influence mechanisms in new product diffusion: an integrative review*, Journal of Marketing Theory and Practice, 22 (2014), pp. 185–208.

[57] M. E. Newman, *The structure and function of complex networks*, SIAM review, 45 (2003), pp. 167–256.

[58] M. E. J. Newman, *Networks: An Introduction*, 2010.

[59] A. A. N. Now, *Cop shooting, ebola scare in atlanta invented by russians: Report.* `https://www.ajc.com/news/cop-shooting-ebola-scare-atlanta-invented-russians-report/Ux6zeq8OQeU3pgHMyPKP3O/`. Retrieved on October 25, 2019.

[60] L. Page, S. Brin, R. Motwani, and T. Winograd, *The pagerank citation ranking: Bringing order to the web.*, tech. rep., Stanford InfoLab, 1999.

[61] S. Panzeri, R. Senatore, M. A. Montemurro, and R. S. Petersen, *Correcting for the sampling bias problem in spike train information measures*, Journal of neurophysiology, 98 (2007), pp. 1064–1072.

[62] S. Panzeri and A. Treves, *Analytical estimates of limited sampling biases in different information measures*, Network: Computation in neural systems, 7 (1996), pp. 87–107.

[63] M. Prokopenko and J. T. Lizier, *Transfer entropy and transient limits of computation*, Scientific reports, 4 (2014), p. 5394.

[64] Z. Qasem, M. Jansen, T. Hecking, and H. U. Hoppe, *On the detection of influential actors in social media*, in 2015 11th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), IEEE, 2015, pp. 421–427.

[65] S. Ranganathan, K. Nakai, and C. Schonbach, *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*, Elsevier, 2018.

[66] J. G. Rasmussen, *Lecture notes: Temporal point processes and the conditional intensity function*, arXiv preprint arXiv:1806.00221, (2018).

[67] E. M. Rogers, *Diffusion of innovations*, Simon and Schuster, 2010.

[68] D. M. Romero, W. Galuba, S. Asur, and B. A. Huberman, *Influence and passivity in social media*, in Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, 2011, pp. 18–33.

[69] T. Santos, S. Walk, R. Kern, M. Strohmaier, and D. Helic, *Self-and cross-excitation in stack exchange question & answer communities*, in The World Wide Web Conference, ACM, 2019, pp. 1634–1645.

[70] M. Schedl, *The lfm-1b dataset for music retrieval and recommendation*, in Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval, ACM, 2016, pp. 103–110.

[71] T. Schreiber, *Measuring information transfer*, Physical review letters, 85 (2000), p. 461.

[72] M. Shaer, *After 39 years of wrongful imprisonment, ricky jackson is finally free.* `https://www.smithsonianmag.com/history/years-wrongful-imprisonment-ricky-jackson-finally-free-180961434/`. Retrieved on October 25, 2019.

[73] C. E. Shannon, *A mathematical theory of communication*, Bell system technical journal, 27 (1948), pp. 379–423.

[74] A. Silva, S. Guimarães, W. Meira Jr, and M. Zaki, *Profilerank: finding relevant content and influential users based on information diffusion*, in Proceedings of the 7th Workshop on Social Network Mining and Analysis, ACM, 2013, p. 2.

[75] T. Starks and L. Cerulus, *Russia's manipulation of twitter was far vaster than believed.* `https://www.politico.com/story/2019/06/05/study-russia-cybersecurity-twitter-1353543`. Retrieved on October 23, 2019.

[76] M. Stella, E. Ferrara, and M. De Domenico, *Bots sustain and inflate striking opposition in online social systems*, arXiv preprint arXiv:1802.07292, (2018).

[77] G. Szabo, G. Polatkan, P. O. Boykin, and A. Chalkiopoulos, *Social media data mining and analytics*, John Wiley & Sons, 2018.

[78] J. W. Tukey, *Exploratory Data Analysis*, Addison-Wesley, 1977.

[79] A. Vázquez, J. G. Oliveira, Z. Dezsö, K.-I. Goh, I. Kondor, and A.-L. Barabási, *Modeling bursts and heavy tails in human dynamics*, Physical Review E, 73 (2006), p. 036127.

[80] G. Ver Steeg and A. Galstyan, *Information transfer in social media*, in Proceedings of the 21st international conference on World Wide Web, ACM, 2012, pp. 509–518.

[81] D. J. Watts and P. S. Dodds, *Influentials, networks, and public opinion formation*, Journal of consumer research, 34 (2007), pp. 441–458.

[82] D. J. Watts and S. H. Strogatz, *Collective dynamics of 'small-world'networks*, nature, 393 (1998), p. 440.

[83] J. Weng, E.-P. Lim, J. Jiang, and Q. He, *Twitterrank: finding topic-sensitive influential twitterers*, in Proceedings of the third ACM international conference on Web search and data mining, ACM, 2010, pp. 261–270.

[84] H. D. White, B. Wellman, and N. Nazer, *Does citation reflect social structure?: Longitudinal evidence from the "globenet" interdisciplinary research group*, Journal of the American Society for information Science and Technology, 55 (2004), pp. 111–126.

[85] J. Wilson, *The meaning of blackgirlmagic, and how you can get some of it.* `https://www.huffpost.com/`

entry/what-is-black-girl-magic-video_n_5694dad4e4b086bc1cd517f4. Retrieved on October 25, 2019.

[86] W. Woess, *Lecture notes in discrete stochastics and information theory*, March 2019.

[87] Y. Yamaguchi, T. Takahashi, T. Amagasa, and H. Kitagawa, *Turank: Twitter user ranking based on user-tweet graph analysis*, in International Conference on Web Information Systems Engineering, Springer, 2010, pp. 240–253.

[88] W. W. Zachary, *An information flow model for conflict and fission in small groups*, Journal of anthropological research, 33 (1977), pp. 452–473.

[89] J. Zhang, X. Zhu, H. Wang, and Y. Guan, *Modeling social media user content generation using interpretable point process models*, arXiv preprint arXiv:1807.10853, (2018).

Appendix

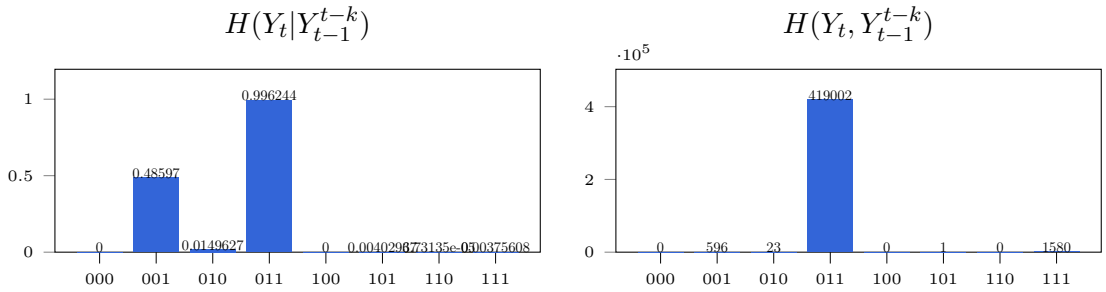


Figure A.1: Conditional and joint probability distribution when computing transfer entropy for three different bins.

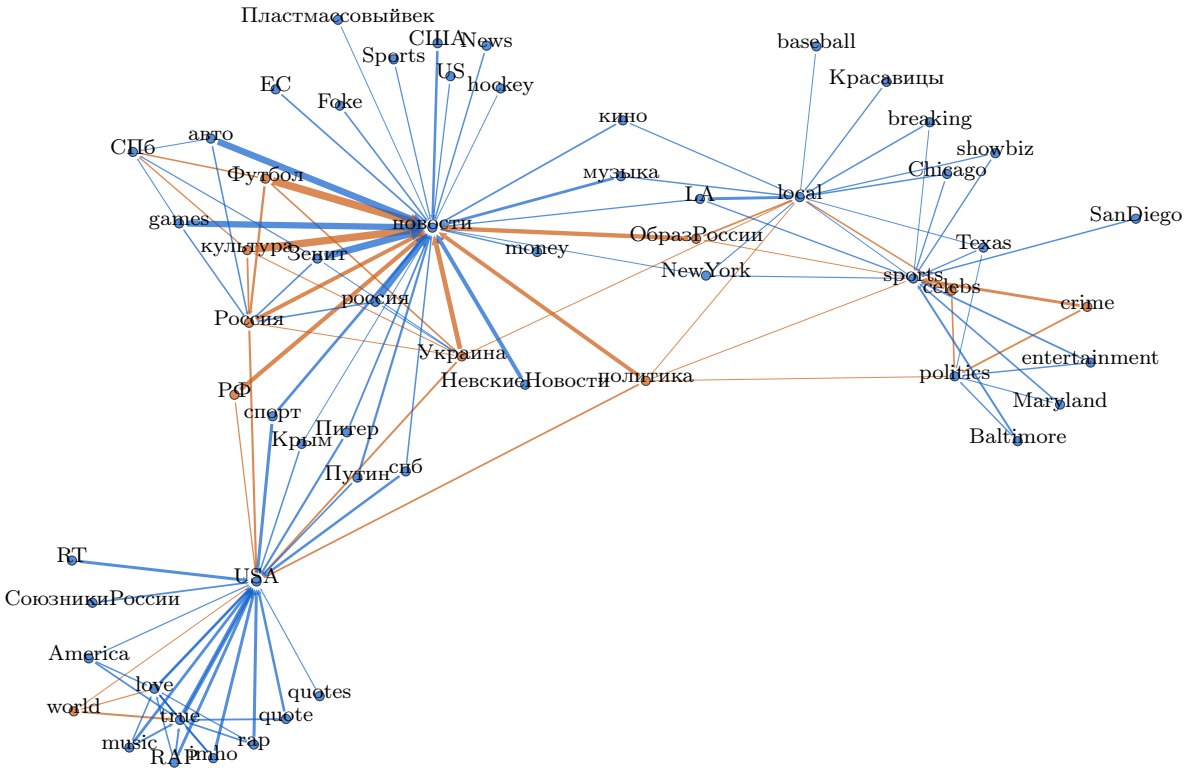


Figure A.2: 100 most used hashtags / bins: $\delta_0 = 1, \delta_1 = 192, \delta_2 = 190 \cdot 2$. Visualization of the top hashtags according to total outdegree TE. The orange nodes refer to one of the top 10 hashtags according to total outdegree TE. The orange edges mean that the source of the edge is a top 10 hashtag.

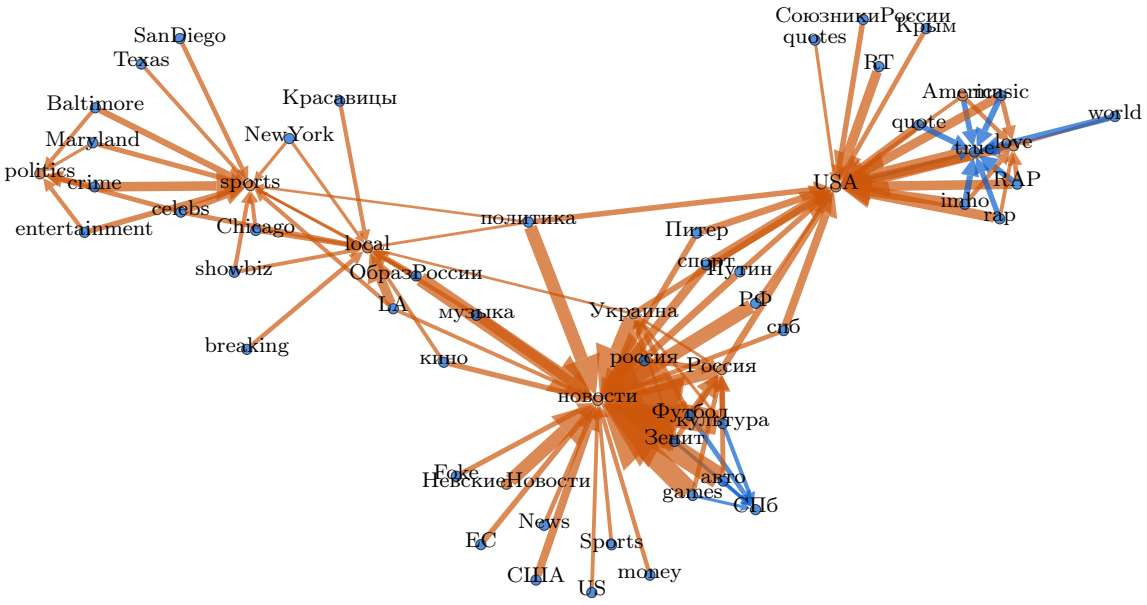


Figure A.3: 100 most used hashtags / bins: $\delta_0 = 1, \delta_1 = 192, \delta_2 = 190 \cdot 2$. Visualization of the top hashtags according to total indegree TE. The orange nodes refer to one of the top 10 hashtags according to total indegree TE. The orange edges mean that the destination of the edge is a top 10 hashtag.

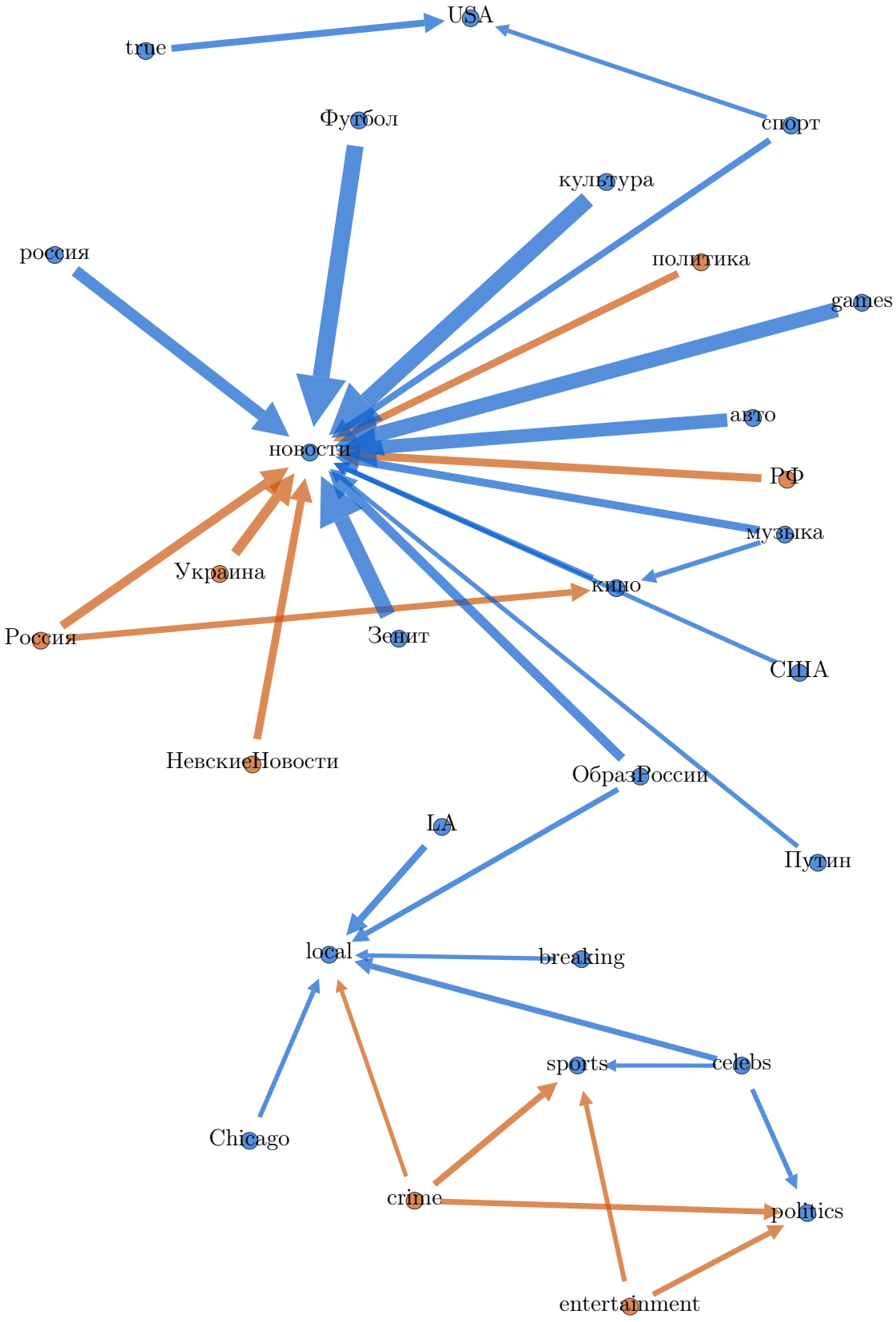


Figure A.4: 100 most used hashtags / bins: $\delta_0 = 18, \delta_1 = 192, \delta_2 = 190 \cdot 2$. Visualization of the top hashtags according to total outdegree TE. The orange nodes refer to one of the top 10 hashtags according to total outdegree TE. The orange edges mean that the source of the edge is a top 10 hashtag.

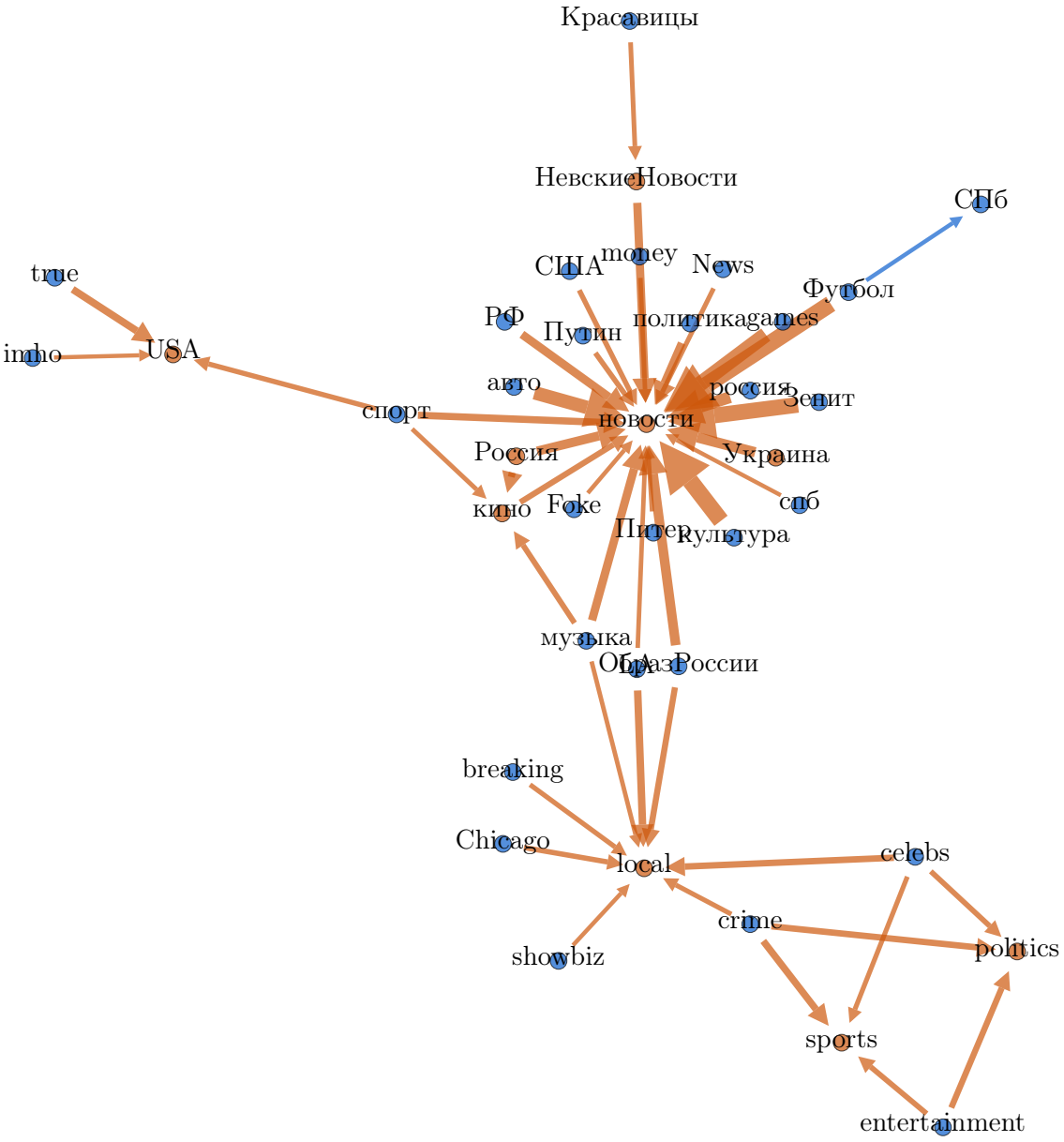


Figure A.5: 100 most used hashtags / bins: $\delta_0 = 18, \delta_1 = 192, \delta_2 = 190 \cdot 2$. Visualization of the top hashtags according to total indegree TE. The orange nodes refer to one of the top 10 hashtags according to total indegree TE. The orange edges mean that the source of the edge is a top 10 hashtag.

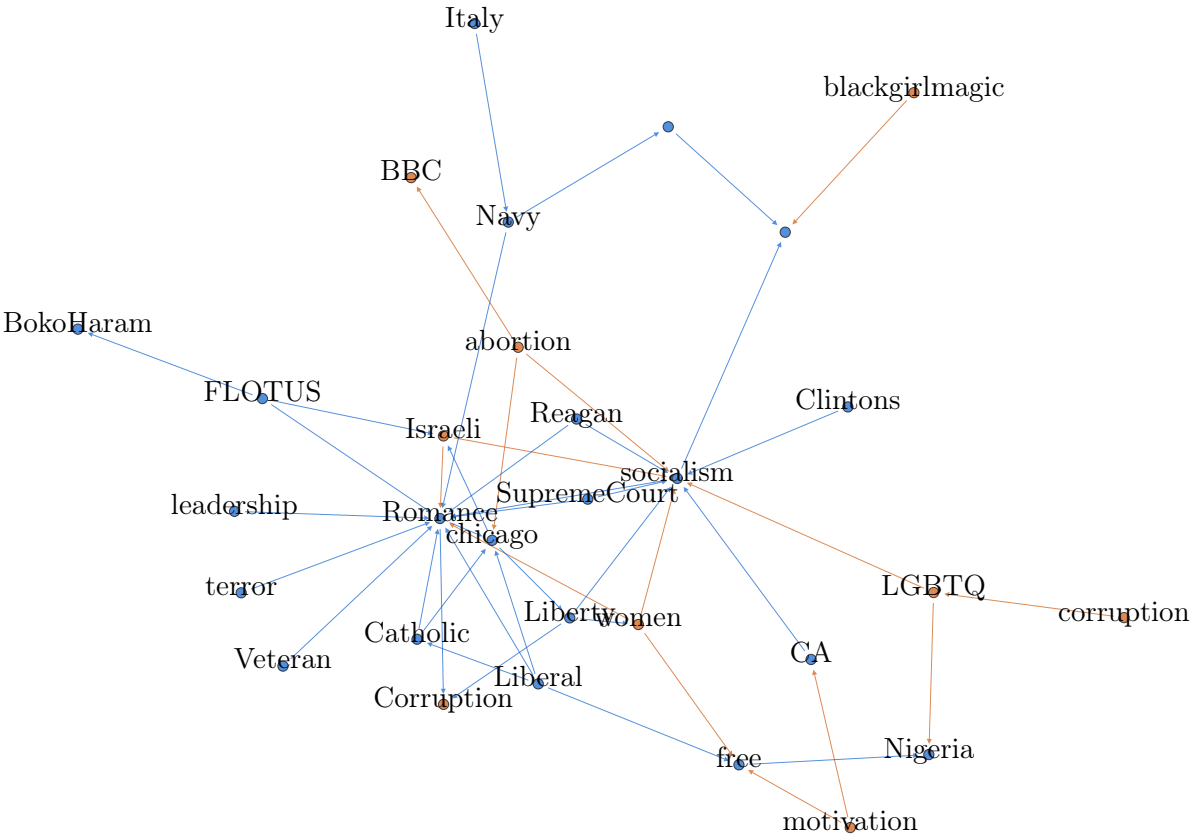


Figure A.6: Quartile-filtered hashtags / bins: $\delta_0 = 1, \delta_1 = 10638/2, \delta_2 = 10638$. Visualization of the top hashtags according to total outdegree TE. The orange nodes refer to one of the top 10 hashtags according to total outdegree TE. The orange edges mean that the source of the edge is a top 10 hashtag.

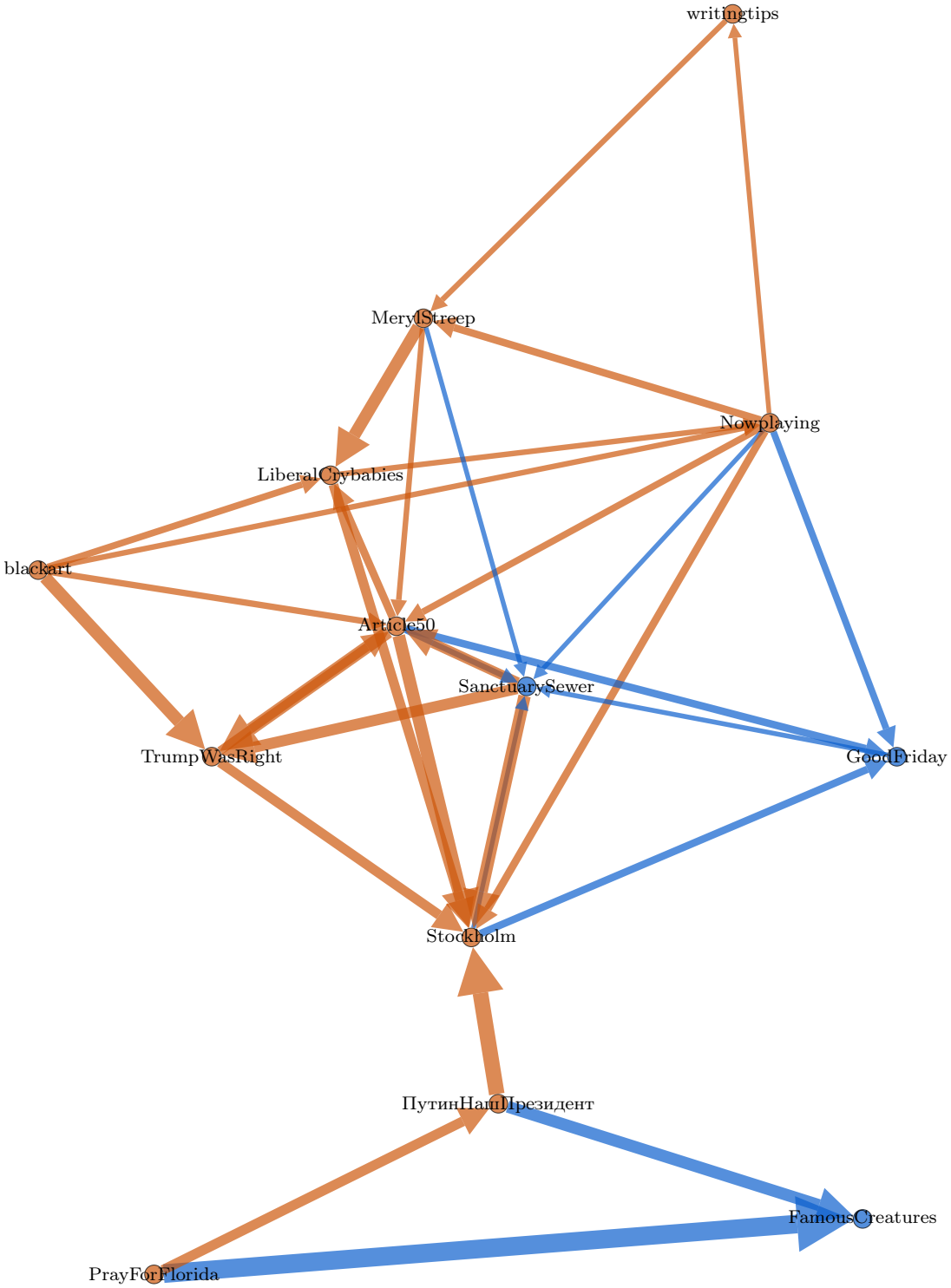


Figure A.7: Quartile-filtered hashtags / bins: $\delta_0 = 1, \delta_1 = 10638/2, \delta_2 = 10638$. Visualization of the top hashtags according to total indegree TE. The orange nodes refer to one of the top 10 hashtags according to total indegree TE. The orange edges mean that the destination of the edge is a top 10 hashtag.

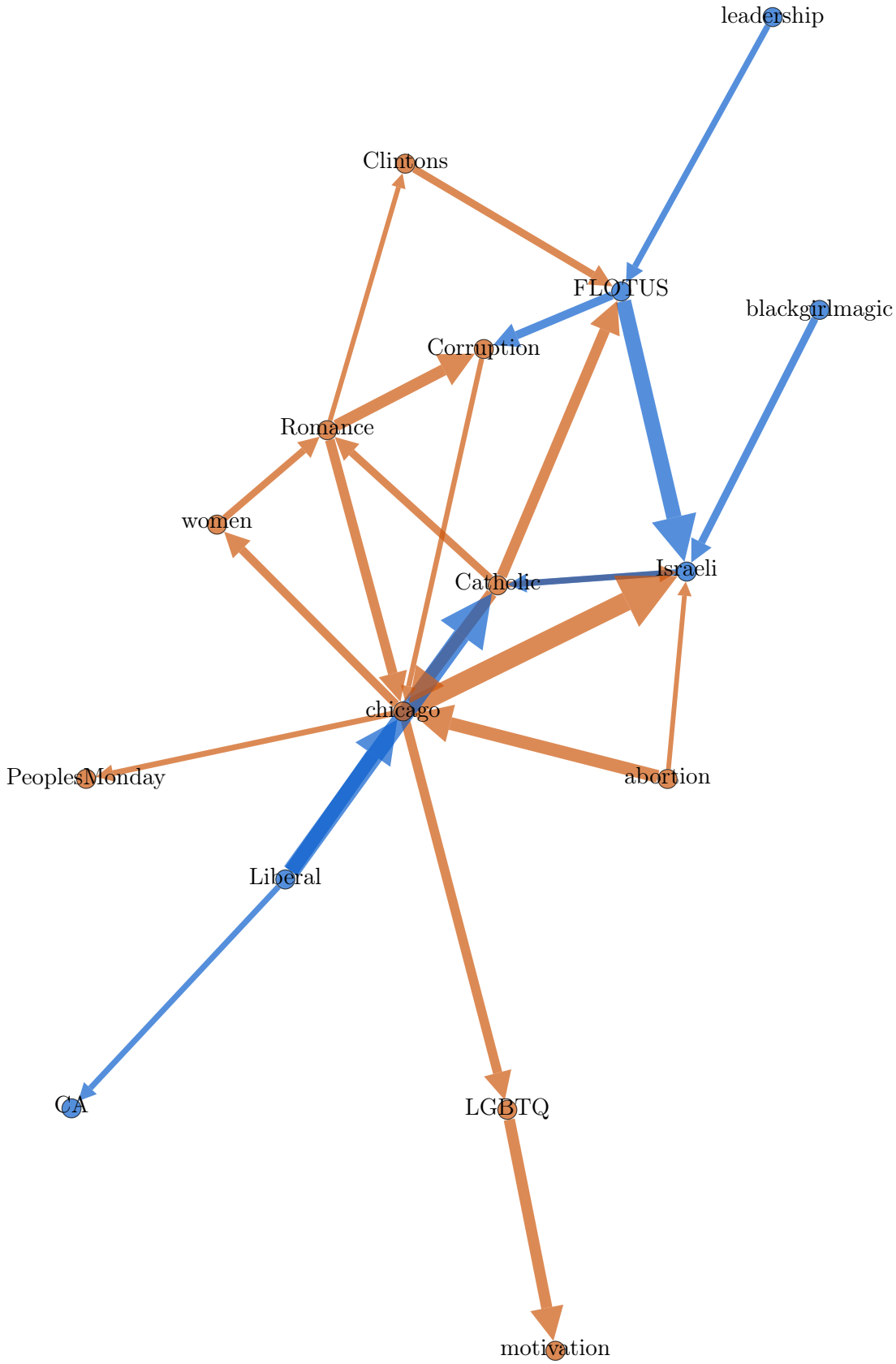


Figure A.8: Quartile-filtered hashtags / bins: $\delta_0 = 44, \delta_1 = 10638/2, \delta_2 = 10638$. Visualization of the top hashtags according to total outdegree TE. The orange nodes refer to one of the top 10 hashtags according to total outdegree TE. The orange edges mean that the source of the edge is a top 10 hashtag.

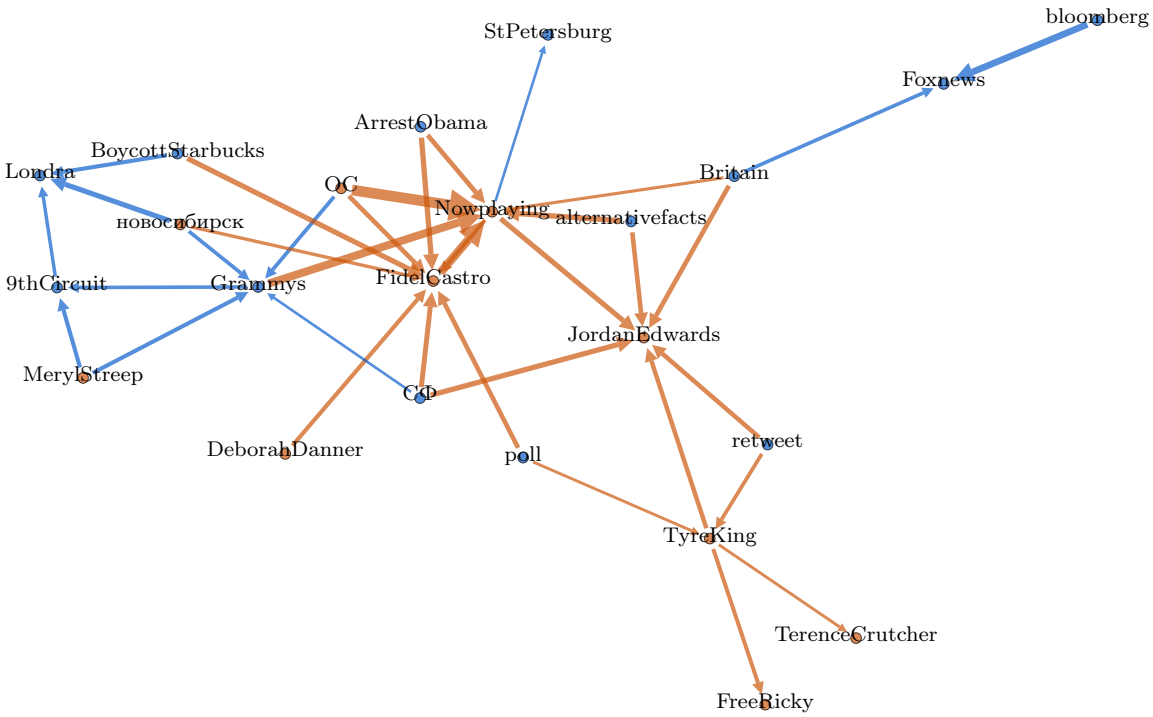


Figure A.9: Quartile-filtered hashtags / bins: $\delta_0 = 44, \delta_1 = 10638/2, \delta_2 = 10638$. Visualization of the top hashtags according to total indegree TE. The orange nodes refer to one of the top 10 hashtags according to total outdegree TE. The orange edges mean that the source of the edge is a top 10 hashtag.

1	Ukraine
2	news
3	football
4	politics
5	automatic
6	culture
7	peak
8	RF (Russian Federation)
9	image of Russia
10	Russia
11	Nevsky News
12	cinema
13	music
14	Putin our president
15	CIA (Central Intelligence Agency)
16	Eurovision
17	beauties of Russia
18	full contact
19	Novosibirsk
20	Anti-masonic movement

Table A.1: English translations of Russian Hashtags