Kritik: *Uncovering the Temporal Dynamics of Diffusion Networks*, Manuel Gomez Rodriguez, David Balduzzi, Bernhard Schölkopf, ICML 2011

Felix Gessert, 15.04.2012, Seminar maschinelles Lernen (Uni Hamburg)

Autoren

Das Paper nennt drei Autoren: Manuel Gomez Rodriguez, David Balduzzi und Bernhard Schölkopf. Ihr Hintergrund und Forschungsschwerpunkt, sowie eine Einschätzung ihrer thematischen Kompetenz wird im folgenden Abschnitt diskutiert.

Manuel Gomez Rodriguez

Person	Ph.D. Student an der Stanford Universität (Kalifornien) im Department of Electrical Engineering, MS Electrical Engineering
Forschungs-Schwerpunkt	Diffusion Networks, Konvexe Optimierung, Brain-Machine Interfaces
Publikationen	2 Journal Artikel, 4 Konferenz Paper, 3 Workshop Paper, insgesamt 3 Publikationen zum Thema Diffusion Networks
Zitationen	106 (nach Google Scholar)
Awards	Mehrere (offenbar eher unbedeutende), z.B. KDD Best Research Paper Award Honorable Mention (2010), ICML Student Travel Scholarship (2011)

Rodriguez scheint als Doktorand einige Erfahrung mit dem Thema Diffusion Networks gesammelt zu haben, kann aber aufgrund seiner kurzen Wirkungsdauer nicht als erfahrener oder gar renommierter Wissenschaftler auf diesem Gebiet bezeichnet werden. Er arbeitete bei der Publikation des Papers mit dem Max-Planck-Institut (MPI) Tübingen zusammen, dessen Direktor und Koautor des Papers Bernhard Schölkopf einer der Betreuer seiner in Arbeit befindlichen Dissertation ist.

David Balduzzi

Person	Post-Doc am MPI Tübingen im Bereich Empirische Inferenz, Ph.D. in algebraischer Geometrie (2006, University of Chiacago), MS Mathematik
Forschungs-Schwerpunkt	Theoretische Neurowissenschaft, verteiltes Lernen, neuronale Netze
Publikationen	12 (veröffentlichte) Publikationen auf Konferenzen und Journals zu Biologie, Angewandter Mathematik und Machine Learning (u.a. NIPS, ICML)
Zitationen	118 (nach Google Scholar)
Awards	NRF Prestigious Scholarship (2001), South African Mathematical Society Bronze Medal (1999)

Als angewandter Mathematiker auf dem Gebiet der neuronalen Netze liegt das Paper im Kernbereich der Forschung von Balduzzi: dem Anwenden mathematischer Methoden auf große Netzwerke mit zeitlicher Dimension. Seit 2010 arbeitet er am MPI Tübingen im Bereich Intelligent Systems (Leitung: Bernhard Schölkopf). Seine Publikationen wurden bisher – gemessen an der Dauer der Forschungsaktivität - nicht auffallend häufig zitiert, was ein Hinweis auf wenig relevante Forschung oder einen unzugänglichen Schreibstil sein könnte.

Bernhard Schölkopf

Person	Direktor des MPI im Bereich Intelligente Systeme, MS Mathematik, Diplom Physik, Forschung an zahlreichen Universitäten und Einrichtungen
Forschungs-Schwerpunkt	Inferenz aus empirischen Daten, Kernel -Methoden für hochdimensionale Daten
Publikationen	Sehr viele, 8 Bücher, 97 Journal Artikel, 181 Konferenz Paper
Zitationen	44592 insgesamt, 29790 davon seit 2007 (nach Google Scholar)
Awards	Sehr viele, zuletzt der Max-Planck Forschungspreis 2011, der mit 750.000 Euro dotiert ist

Professor Bernhard Schölkopf ist ohne Frage eine der führenden Figuren im Bereich Machine Learning. Als Direktor des MPI Bereichs „Intelligent Systems“, ehemaliger Program Chair namhafter Konferenzen (z.B. NIPS, COLT) und Editor (z.B. JMLR), sowie mit Forschungserfahrung bei mehreren renommierten Wissenschaftseinrichtungen (z.B. AT&T Bell Labs, Microsoft Research) hat Schölkopf sich als Wissenschaftsschwergewicht mit großem Einfluss etabliert. Das Erscheinen seines Namens auf dem Paper sollte jedoch nicht überbewertet werden: der Umstand das sein Name zuletzt genannt wird und Balduzzi und Rodriguez in seiner Abteilung forschen, lässt darauf schließen, dass er an der Entstehung des Papers nur peripher beteiligt war. Möglicherweise ist sogar das Erscheinen des Artikels auf der angesehenen ICML mehr seinem gewichtigen Namen denn der Qualität der präsentierten Forschungsergebnisse geschuldet.

Der Inhalt des Papers

Das Paper stellt den „NetRate“ Algorithmus vor, der es erlaubt, aus gegebenen Infektions-Daten das zugrundliegende Diffusionsnetzwerk des Verbreitungsprozesses abzuleiten. Das Modell ist dabei folgendes: eine Menge von Knoten (z.B. Blogger) ist an dem Diffusionsprozess (z.B. dem Ausbreiten eines Internet-Mems) beteiligt, der über die Kanten zwischen den Knoten vermittelt wird. Dabei wird ein probablistisches generatives Modell zugrunde gelegt: jeder infizierte Knoten (z.B. ein Twitter-Stream der ein Mem erwähnt) gibt gemäß einer zeitabhängigen Wahrscheinlichkeitsverteilung seine Infektion an einen benachbarten Knoten (z.B. einen Follower) weiter. Die stochastische Unabhängigkeit der Ausbreitung über zwei Kanten wird dabei (plausiblerweise) vorausgesetzt.

Der vorgestellte Algorithmus ermittelt nun diejenigen Transmissionsraten (ein Parameter der zeitabhängigen Wahrscheinlichkeitsverteilung zwischen zwei Knoten) unter denen die Likelihood-Funktion für die gegebenen Trainingsdaten (z.B. Erwähnungszeitpunkte von

Memen Twitter-Streams für eine Anzahl von Memen) maximiert wird. Das dabei zu lösende Optimierungsproblem ist konvex und kann mit vorhandenen Algorithmen gelöst werden. Es kann zusätzlich dadurch beschleunigt werden, dass die Berechnungen der Transmissionsraten für jeden Ausgangsknoten parallel ermittelt werden. Eine Implementierung des NetRate-Algorithmus auf Basis eines Solvers für konvexe Probleme wurde als MatLab-Implementierung im Web veröffentlicht [1]. Die unterstützten Typen von Wahrscheinlichkeitsverteilungen, deren Transmissionsrate die Ausbreitung zwischen zwei Knoten charakterisiert sind Exponentialverteilung, Potenzgesetz und Rayleighverteilung (illustriert in Abbildung 1). Der Typ der Wahrscheinlichkeitsverteilung ist ein fester Parameter des Verfahrens und wird nicht durch die Optimierung ermittelt. In diesem Sinne kann der Algorithmus also als ein Maximum-Likelihood Parameter-Schätzer bezeichnet werden. Das so bestimmte Diffusionsnetzwerk besteht aus allen gegebenen Knoten und enthält eine Kante für jedes Knotentupel dessen Transmissionsrate größer null ist.

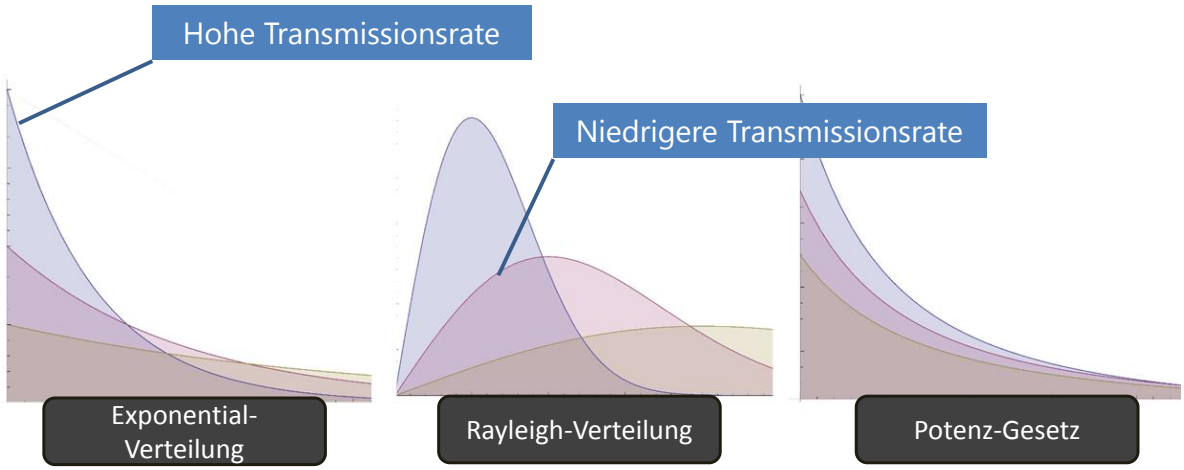

Abbildung 1 Unterstützte Wahrscheinlichkeitsverteilungen

Die Autoren vergleichen ihren Algorithmus mit den Algorithmen „NetInf“ und „ConNie“ bezüglich der Gütekriterien Precision, Recall und Accuracy auf synthetischen und echten Daten (aus dem Meme Tracker Dataset). Im Unterschied zu den beiden anderen Algorithmen erzielt NetRate ein eindeutiges Ergebnis statt einer Lösungsmenge, weshalb der Vergleich nicht offensichtlich ist. Die Autoren stellen jedoch fest, dass in den beschriebenen Fällen NetRate mindestens ebenbürtig ist. Die Innovation ihres Ansatzes sehen die Autoren in dem Umstand, dass NetRate als erster Algorithmus die zeitliche Dimension des Diffusionsprozesses auf Basis von Knotenpaaren betrachtet und keine fachlichen Annahmen über die Natur des Diffusionsprozesses macht (z.B. Bloggermentalitäten bei einem Internet-Mem).

Einordnung nach Qualität, Klarheit, Originalität und Signifikanz

Qualität

Die mathematische Notation des Artikels ist sehr sauber, was sicherlich dem mathematischen Hintergrund des Zweitautors geschuldet ist. Die Beweise – obwohl anspruchsvoll – sind relativ kompakt und nachvollziehbar. Der vergleichende Abschnitt des Papers hat deutliche Schwächen: es wird nicht sauber deklariert, in welcher Weise sich die Ergebnisse der drei

Algorithmen unterscheiden, d.h. welche manuellen Parameter verwendet wurden. Auch der konzeptuelle Unterschied der Algorithmen wird nicht präzise erläutert. Die Laufzeit und Skalierbarkeit von NetRate wird nicht in Beziehung zu den anderen Algorithmen gesetzt. Dies scheint selbst für eine theoretisch fokussierte Arbeit unsauber. Die Motivation für die Nutzung eines künstlichen Diffusionsnetzwerkes, auf dem NetRate - nicht überraschend – gut funktioniert wird nicht dargelegt. Der Vergleich auf realen Daten fällt demgegenüber sehr kurz und unergiebig aus, vermutlich nicht zuletzt deshalb, weil NetRate dort keine hohe Güte zeigt und nicht auf die Größe des ungekürzten Datensets zu skalieren scheint. Der gesamte Vergleich wirkt damit insgesamt unaufrichtig und unbefriedigend. Den Autoren gelingt es allerdings schon einführend zu erklären, wie sich ihre Arbeit prinzipiell von vorangehenden Arbeiten abhebt.

Klarheit

Der theoretische Teil des Papers kommt mit sehr stringenten Definitionen und kurzen Beweisen aus. Die Klarheit des Artikels hätte jedoch deutlich durch ein illustratives einführendes Beispiel (z.B. Diffusion in sozialen Netzwerken) verbessert werden können. Auch auf die Visualisierung der Netzwerke und Ergebnisse wurde verzichtet. Dies kommt zwar der Kürze des Papers zugute, schmälert aber seine Eingängigkeit. Sprachlich kann das Paper (bis auf wenige Rechtschreibfehler) überzeugen und der Schreibstil eloquent und sachlich. Die Autoren versäumen es allerdings im Paper auf die Website mit der MatLab-Implementierung zu verweisen.

Originalität

Die Autoren geben einen klaren Überblick über die verwandten Arbeiten und grenzen ihren Ansatz klar davon ab: als erste seien sie auch in der Lage die temporale Dynamik in Diffusionsnetzwerken aus beobachteten Daten abzuleiten und in ihrem Modell zu fassen. Der differenzierte Vergleich mit den Algorithmen NetInf und ConNie fehlt jedoch. Zwar werden die entsprechenden Publikationen referenziert aber eine klare Abgrenzung der Algorithmen gegeneinander wird nicht gegeben. Unklar bleibt während der Herleitung, welche Teile von vorangehenden Publikationen übernommen oder inspiriert sind und welche originär sind.

Signifikanz

Der Inhalt des Papers scheint von hoher Signifikanz zu sein. Die Untersuchung und Visualisierung der Dynamik sozialer Netzwerke erfreut sich zunehmender Beliebtheit [2]. Ich halte es jedoch für wahrscheinlich, dass der Artikel in der bisherigen Form ein zu enges Publikum adressiert. Das Interesse an der Verarbeitung großer Datenmengen mit Web-Bezug scheint derzeit besonders im Umfeld der Cloud-Computing-, Big-Data- und NoSQL-Community angesiedelt zu sein. Hier verpassen die Autoren die Gelegenheit darauf hinzuweisen, dass eine Implementierung auf Basis aktueller NoSQL Graphen-Datenbanken wie Neo4J oder Wide-Column-Stores wie HBase kombiniert mit einem Framework für skalierbare, verteilte Algorithmen wie Hadoop von großem Vorteil wäre. Die Erwähnung solcher Skalierungs- und Anwendungsperspektiven hätte dem Paper sicherlich Aufmerksamkeit verschafft, die über den Kreis der theoretischen Machine-Learning Community hinausgeht.

Zusammenfassung

Es werden nun abschließend jeweils drei Stärken und Schwächen des Papers genannt.

Stärken des Papers

- Ein neuer Ansatz, der ein aktuelles und interessantes Thema adressiert und eine frei verfügbare Implementierung bereitstellt.
- Das generative probabilistische Modell der Diffusion in Netzwerken auf Basis von Wahrscheinlichkeitsverteilungen ist intuitiv und leicht verständlich.
- Die mathematischen Schritte sind präzise und konsistent formuliert.

Schwächen des Papers

- Der NetRate Algorithmus ist ein Maximum-Likelihood Schätzer, der das Produkt der Stichprobenwahrscheinlichkeiten als Likelihood verwendet. Das impliziert, dass die Trainingsdaten unabhängig und identisch verteilt sein müssen (iid). Ich denke, dass diese Annahme in sehr vielen Fällen nicht gerechtfertigt ist (u.a. in dem Meme Tracker Dataset): das Auftauchen eines Mems z.B. begünstigt das anschließende Auftauchen eines ähnliches Mems mit Bezug zum ersten Mem (z.B. aufgrund geänderter Follower-Beziehungen). Die Unabhängigkeit der Stichproben ist dann nicht länger gegeben. Die gleiche Verteilung jeder Stichprobe ist ebenfalls eine zu starke Annahme: die Autoren nennen z.B. Rayleigh-Verteilungen als passend für die Diffusion bei kurzlebigen Trends („Fads“) und Potenzgesetze als passender für langlebigere Trends. Zwei Stichproben kann also eine unterschiedliche Verteilung zugrunde liegen, wodurch die Annahme der gleichen Art von Wahrscheinlichkeitsverteilungen für alle Stichproben ebenfalls ungerechtfertigt ist.
- Die Vergleiche mit den Algorithmen NetInf und ConNie wirken aufgrund des künstlichen Datensets und der fehlenden Abgrenzung unaufrichtig.
- Es wird vollständig auf erläuternde Visualisierungen verzichtet.

Referenzen

[1] Implementierung des NetRate Algorithmus in MatLab: http://www.stanford.edu/~manuelgr/netrate/

[2] Strata Conference on Big Data. http://lanyrd.com/2011/stratany