UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

RAFAEL BALDASSO AUDIBERT

On the Evolution of AI and Machine Learning: Analyses of Impact, Leadership and Influence over the Last Decades

Work presented in partial fulfillment of the requirements for the degree of Bachelor in Computer Science

Advisor: Prof. Dr. Anderson Rocha Tavares
Coadvisor: Prof. Dr. Luís da Cunha Lamb

Porto Alegre
May 2022

“Machines take me by surprise with great frequency.”

— ALAN MATHISON TURING, 1950

ACKNOWLEDGMENTS

This Bachelor Thesis is not a one-man's work. I wouldn't have written the first character without the help of those whom I gladly share my life with it.

Thank you, Lili, my love, for being the most genuine symbol of empathy, love-sharing, and understanding. This thesis is only been written because of decisions we took together, and that changed my life forever. Every small piece of life I have now is due to having met and fallen in love with you. From the deepest of my heart, my life couldn't be better than it is, and it's all because of you. Nós contra o mundo, pra todo sempre.

Thank you, Mom, Dad, and Vini. Y'all are my life. Mom, thank you for being the most caring person ever. Dad, thank you for being the most supportive person ever. Vini, thank you for being the biggest zoomer ever, and for always sharing your wins with me. I love the three of you in every single beat of my heart.

Grazie, granma Inês and granpa Izalino. You guys were always everything I needed. Thanks for always being there cheering for me, and never letting me down. I miss you, grandpa, so much, every single day. I stole your old ragged deck of cards to myself, hope you don't mind.

Thanks to all the friends who made me arrive where I am now. Thanks, Ana, Andrei, Bardini, and Teo for being there from the beginning, bearing the same hard assignments, sleeping in the same boring classes, and living the same hard college student life. Thanks, Vini, for bringing me some peace of mind when I had to play LOL together with those silvers. Thanks, Andy, for being the most annoying friend and coworker ever, you always bring me joy.

Special thanks to those who introduced me to the academia, and provided me a way to tolerate those boring days while these work experiments were running: Henrique, Pedro, and Marcelo. This work has a lot of you in it.

Thanks, Iron, Julio, Emily, and Giovanna for having provided me with the best exchange experience it could have been. You guys were always there in the moments I felt the loneliest, and the furthest from those I loved. I thank y'all for every Simpel beer, every trip to Heidelberg, every freezing bike (or roller) ride at 1 AM on a Thursday, and for those weird but terrifically tasty coxinhas. You were responsible for making this experience unforgettable. Liebe euch alle.

Danke, Jonas, dass du in diesen 10 Monaten der verständlichste Forschungskoordinator warst, den ich mir hätte wünschen können. Ich weiß, dass ich die Erwartungen nicht

erfüllt habe, aber ich hoffe, Sie wissen, dass Sie meine Austauschzeit viel überschaubarer gemacht haben. Hoffentlich können wir uns wiedersehen.

Thanks, as well, to my employer LeadSimple, in special Daniel, Chris, and Jordan. Having the stability and freedom to be able to both work in an admirable company and live a balanced life, with my work, studies, hobbies, and travels is everything I've ever wanted.

Some experiments in this work used the PCAD infrastructure, at INF/UFRGS (<http://gppd-hpc.inf.ufrgs.br>). I am very thankful for their help in our work by allowing us to run these experiments, especially Prof. Lucas Melo Schnorr for always being extremely solicitous and helpful, even when I didn't know how SSH keys worked properly yet – I probably still don't know – and got them mixed up several times in a row.

At last, I thank every single person who has helped me throughout my path at INF, both on those cloudy winter Mondays between the buildings, and on those sunny evenings when Ana would do her daily photosynthesis. Thanks, INF for providing me the best of everything, even in the darkest times to the Brazilian science landscape, especially all the professors that crossed my path and enlightened me, in no particular order: Prof. Luis da Cunha Lamb, Prof. Anderson Rocha Tavares, Prof. Rodrigo Machado, and Prof. Valter Roesler. Thanks, UFRGS for having been everything I needed, so that I could become everything I wanted to be.

ABSTRACT

Artificial Intelligence has changed the world we live in. It all started in the academia where we had some seminal works in the area such as (MCCULLOCH; PITTS, 1943), (TURING, 1950), (MINSKY, 1961), and (VALIANT, 1984). It is a fact, however, that AI ended up becoming much more than a research topic explored only in universities: AI has led to uncountable articles by large enterprises like (GOMEZ-URIBE; HUNT, 2016), (RAMESH et al., 2021), and (MAHABADI et al., 2022). Even though the area has seen a big evolution since it started, there hasn't been much exploration about the evolution of it, and the dynamics involved in the process of transforming it into a quite well-established computer science area. This work, therefore, intends to shed some light on the history of Artificial Intelligence, exploring the dynamics involved in its evolution through the lenses of the papers published in AI conferences since the first IJCAI conference in 1969. We achieve so by creating comprehensive citation/collaboration paper/author datasets and computing its centralities looking for insights on how the area has reached its current state. Throughout the process, we correlate these datasets with the Turing Award winners, and the two winters the AI has field has gone through already, also looking at self-citation trends and new authors' behaviors. Finally, we also present a novel way to infer the country of affiliation of a paper from its organization. This work, therefore, provides a deep analysis of the Artificial Intelligence history, from the most diverse points of view, enabling insights that, to the best of our knowledge, weren't studied before.

Keywords: Machine learning. artificial intelligence. data analysis. graph. centrality measures. conferences. influence. ethics. turing award. affiliation country.

Sobre a Evolução da IA e Aprendizado de Máquina: Análises do Impacto, Liderança e Influência nas Últimas Décadas

RESUMO

A Inteligência Artificial mudou completamente o mundo em que vivemos. Tudo começou na academia onde tivemos obras seminais como (MCCULLOCH; PITTS, 1943), (TURING, 1950), (MINSKY, 1961), e (VALIANT, 1984). A IA, porém, se tornou muito mais do que um tópico de pesquisa existente somente em universidades: a IA gerou também incontáveis trabalhos de grandes corporações como (GOMEZ-URIBE; HUNT, 2016), (RAMESH et al., 2021), and (MAHABADI et al., 2022). Embora a area tenha tido uma grande evolução desde o seu início, não houveram muita explorações sobre sua evolução, e as dinâmicas envolvidas no processo que transformou a área em uma das mais estabelecidas da Ciência da Computação. Esse trabalho, portanto, pretende colocar um holofote sobre a história da Inteligência Artificial, exporando as dinâmicas de sua evolução utilizando como lente os trabalhos publicados em conferências da área de IA desde o primeiro IJCAI em 1969. Nós conseguimos isso criando *datasets* (conjuntos de dados) completos e compreensivos sobre citações/colaboração entre autores/trabalhos, computando suas centralidades e buscando possíveis instrospecções de como a área atingiu o seu estado atual. Durante esse processo nós correlacionamos esses conjuntos de dados com os ganhadores da Turing Award, e com os dois "AI Winters" (Invernos da IA) que a área já enfrentou, olhando também para tendências de citações próprias e comportamento de novos autores. Finalizando, apresentamos também um novo método para inferir o país de afiliação de um trabalho a partir de sua organzação. Esse trabalho, portanto, provÊ uma análise profunda da história da Inteligência Artificial, a partir dos mais diversos pontos de vistas, permitindo análises e percepções que, até onde o nosso conhecimento permite, nunca haviam sido estudadas antes.

Palavras-chave: aprendizado de máquina, inteligência artificial, análise de dados, grafos, medidas de centralidade, conferências, influência, ética, turing award, país de afiliação, conjunto de dados.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

e.g.	*exemplum gratia* (en: for example)
et al.	*et alia* (en: and others)
i.e.	*id est* (en: that is)
AI	Artificial Intelligence
ML	Machine Learning
DL	Deep Learning
NLP	Neural Language Processing
KD	Knowledge Discovery
IR	Item Retrieval
AAAI	Association for the Advancement of Artificial Intelligence Conference
ACL	Association for Computational Linguistics Conference
CVPR	Conference on Computer Vision and Pattern Recognition
ECCV	European Conference on Computer Vision
EMNLP	Conference on Empirical Methods in Natural Language Processing
ICCV	International Conference on Computer Vision
IJCAI	International Joint Conference on Artificial Intelligence
KDD	SIGKDD Conference on Knowledge Discovery and Data Mining
NAACL	North American Chapter of the Association for Computational Linguistics Conference
NIPS	Old abbreviation for Conference on Neural Information Processing Systems
NeurIPS	Conference on Neural Information Processing Systems
SIGIR	Special Interest Group on Information Retrieval Conference
WWW	International World Wide Web Conference
ACM	Association for Computing Machinery
XML	Extensible Markup Language

JSON	JavaScript Object Notation
CERN	European Organization for Nuclear Research
DARPA	American Defense Advanced Research Projects Agency
FGCP	Japanese Fifth Generation Computer Project
USA	United States of America
MAG	Microsoft Academic Graph
UUID	Universally unique identifier

LIST OF SYMBOLS

$e_{u,v}$	Graph edge e from node u to node v
k_u	Degree of a node u
$\partial_{s,t}$	Number of shortest paths between nodes s and t
$\partial_{s,t}(u)$	Number of shortest paths between nodes s and t through node u
$d(u,v)$	Distance between nodes u and v
$d_2(v)$	Number of neighbors of v plus the sum of the number of neighbors for every v's neighbor

LIST OF ALGORITHMS

CONTENTS

1 INTRODUCTION

Artificial Intelligence has changed the world we live in. It all started in the academia where we had some seminal works in the area such as (MCCULLOCH; PITTS, 1943), (TURING, 1950), (MINSKY, 1961), and (VALIANT, 1984), but AI ended up becoming much more than a research topic explored only in universities. AI has led to uncountable articles by large enterprises like (GOMEZ-URIBE; HUNT, 2016), (RAMESH et al., 2021), and (MAHABADI et al., 2022). There are surveys[1] that show that AI is being used in 37% of the organizations, at least to some extent. For over half a century – the first comprehensive AI conference, IJCAI, started in 1969 –, Artificial Intelligence has broken barriers and surprised many who doubted it could not achieve groundbreaking results. In the 1990s, Deep Blue (CAMPBELL; HOANE; HSU, 2002) became the first computer to win a chess match against the then reigning chess world champion, Garry Gasparov, under tournament conditions.

Later, AI would eventually be able to reach even higher grounds in a wide number of applications: AlphaGo (SILVER et al., 2016) won a series of matches against Go world champions, Brown et al. (2020) can generate texts that resemble human-like competence, Cobbe et al. (2021)'s work was shown to solve math word problems, Jumper et al. (2021) can accurately predict protein structure folding, Park et al. (2019) can render real life-like images from segmentation sketches, to name a few.

Even though the area has seen a big evolution since it started, there hasn't been much exploration about the evolution of it, and the dynamics involved in the process of transforming it into a quite well-established computer science area. Some influential (and maybe polarizing) researchers such as Gary Marcus have discussed the developments that happened in the area in recent years in (MARCUS, 2018) or even wondered what's to come in the next decade in (MARCUS, 2020). On the other hand, in this work, we look a bit further back in Artificial Science's history, and explore deeper the processes that caused it to become the academic and business success it is today.

In this work, we will explore how the collaboration and citation networks of researchers evolved since 1969, within the three flagship AI conferences – IJCAI, AAAI, and NeurIPS – together with some flagship conferences of research areas impacted and influenced by AI, namely CVPR, ECCV, ICCV, ICML, KDD, ACL, EMNLP, NAACL, SIGIR, and WWW. Even though not all of these conferences had a vast number of AI-

[1] <https://www.gartner.com/en/newsroom/press-releases/2019-01-21-gartner-survey-shows-37-percent-of-organizations-have>

related papers published in their early years, we add them to our work to compose a “big picture” of how AI has not only grown itself but also gradually started to influence other subjects, such as computer vision, natural language processing, and information retrieval.

We achieve so by exploring, and enhancing, a big dataset of papers published in Computer Science venues since 1969, the v11 Arnet dataset (TANG et al., 2008). We use version v11 from this data dataset, containing data originating from DBLP with further disambiguation in regards to paper authorship, spanning from 1969 to 2019. There are versions v12 and v13 available with data until 2021, but the data for the recent years is pretty degraded in these more recent datasets, thus rendering the statistical analysis on them useless (See Section 3.1 to understand our trade-offs on using v11 instead of v13). This dataset is then used to create a dataset of our own, modeled in several different graph representations of the same data, allowing us to explore them in a true network fashion. These graphs – with their centralities already computed – are made available for future research, as the process to generate them involves the use of supercomputers with amounts of memory and processing not easily found outside big universities or companies.

Our analyses then make use of these centralities to rank both papers and authors over time using citation and collaboration networks. We then correlate these rankings to external factors, such as conferences loaction, or the Turing Award – the most coveted award in Computer Science. This will allow us to explore what/who were/are the influential papers/authors in every area/venue. Additionally, we will also explore the dynamics of where all this research is being produced, trying to understand the recent shift of production from the United States to China.

In these analyses (Chapter 4), we show how authors do not keep influence in an area for a long period, with the trend not being confirmed if we rank papers by importance, as they have the ability to be respected/important for a longer period of time. We also show how the average number of authors per paper is increasing in the researched venues, as well as the number of self-citations. Furthermore, we also take a peek at the authors who introduce most people to these conferences, by checking how many papers an author is where that paper is the "first paper" for one of the authors. We also show the dynamics behind citations between conferences, showing how some conferences work better together than others.

Because of the nature of our work – converting huge amounts of unstructured data into a structured data format – we also generate some side contributions besides our main work: a new and efficient Python library to convert XML to JSON that uses file streams

instead of loading the whole data in memory; a parallel Python implementation to compute some centrality measures for graphs, using all physical threads available in a machine; a novel structure to avoid reprocessing data already processed when its underlying structure is a graph.

Our work is organized as follows: Chapter 2 provides analyses of the history behind Artificial Intelligence, and some background information on the analyzed computer science conferences, the Turing Award, and a review of graphs in general; Chapter 3 elaborates on the methodology used to fulfill this work, including information about the underlying dataset and the process behind the generation of the graphs/charts used throughout this work; Chapter 4 presents and discusses the analyses of the aforementioned data under various perspectives; Chapter 5 concludes our work with some other insights, shedding some light over what was seen in the work, and listing possible suggestions for future work using this new dataset; The Appendix brings some tables and figures that do not properly fit the main part of this work.

2 BACKGROUND

2.1 Artificial Intelligence History

The Artificial Intelligence History can be defined in terms of five main time periods: three important ones where the field grew stronger and stronger, interluded by two periods where the area was discredited and thought to be of little real-world impact, aptly named "AI Winters".

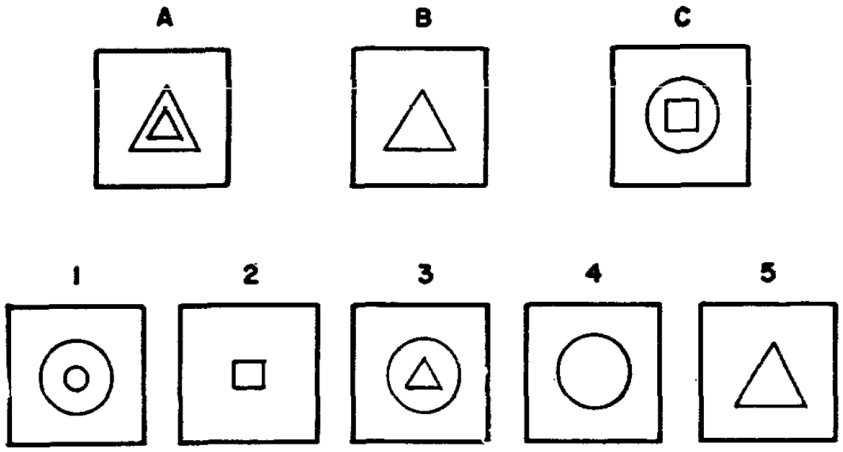
2.1.1 The dawn of AI *(1940-1974)*

Although debatable, the first modern artificial intelligence papers were published in the 1940s. One of the first artificial neural networks-related papers arguably dates back to 1943, when Warren McCulloch and Walter Pitts formally defined how simple logical operations from propositional logic could be computed in a connectionist setting (MCCULLOCH; PITTS, 1943). Later, in 1950, Alan Turing published the widely cited "Computing Machinery and Intelligence" paper (TURING, 1950), the first philosophical paper related to AI. In this paper, Turing reflects if machines are able to think and also proposes some sort of "imitation game" (now widely known as the Turing Test) in order to verify the reasoning and thinking capabilities of computing machines. Nevertheless, it was in 1956 that the term Artificial Intelligence (AI) was coined by John McCarthy during the Dartmouth Summer Research Project on Artificial Intelligence workshop. From the workshop onward, A.I. rapidly evolved into a potentially world-changing research field – at that time, especially focusing on the symbolic paradigm, revolving around rule-based systems. In 1963, the first collection of AI articles would be published in (FEIGENBAUM; FELDMAN, 1963).

A great example of these primitive rule-based systems is Eliza (WEIZENBAUM, 1966), the first-ever chatbot, created in 1964, by Joseph Wiezenbaum at the Artificial Intelligence Laboratory at MIT. It is undeniable how huge is the chatbot market in our age, powering huge multi-million dollar company's revenues like Intercom[1] or Drift[2]. Eliza was created to be an automated psychiatrist, as if the human was talking to someone who understood their problems, although the system worked in this rule-based format, replying

[1] <https://www.intercom.com/>
[2] <https://www.drift.com/>

Figure 2.1 – Geometry Analogy Problem example



Source: (EVANS, 1964)'s Figure 1

to the user with pre-fed answers. Besides the main artificial intelligence approach, we can already see how related areas are easily influenced with a chatbot clearly involving natural language processing as well.

It would also be in 1964 that (EVANS, 1964) would show that a computer could solve what they described as "Geometry Analogy Problems", which correlates with the problems usually displayed in IQ tests where one needs to solve a question in the format "figure A is to figure B as figure C is to which of the given answer figures?" such as the one represented in Figure 2.1

Important research would also vouch in favor of the area, causing DARPA (the American Defense Advanced Research Projects Agency) to fund several different AI-related projects from the mid-'60s onwards, especially at MIT.

This era was marked by the extreme optimism in the speeches of the area practitioners. Marvin Minsky said in a 1970s Life magazine interview – one year after receiving the Turing Award (See Section 2.2) – that "from 3-8 years we will have a machine with the general intelligence of a human being". He would also, in the same interview, boldly claim that "If we're lucky, they might decide to keep us as pets.". Science Fiction fully adopted the Artificial Intelligence utopic future theme, with the release of famous movies like the french "Alphaville" in 1965 by Jean-Luc Godard, and "2001: A Space Odyssey" by Stanley Kubrick and Arthur Clarke in 1968.

Prior to its first fall into oblivion, however, AI had drawn enough attention to be the main theme of an international conference, the First International Joint Conference on Artificial Intelligence (IJCAI), held at Stanford, in 1969. In it, out of the 63 published

papers, we have some of notice such as Stanford's work in a "system capable of interesting perceptual-motor behavior" (FELDMAN et al., 1969), Nilsson (1969)'s Mobius automation tool, and Green (1969)'s QA3 computer program that can write other computer programs and solve practical problems for a simple robot.

It was also before the first winter that Alain Colmerauer would develop Prolog, one of the most famous programming languages responsible for powering most of the early AI algorithms. The feature that makes Prolog stand out among other languages is the fact that it is mostly a declarative language: the program logic is expressed in terms of relations, represented as facts and rules. A computation is initiated by running a query over these relations (LLOYD, 1984). Prolog would become the programming language chosen to build Watson[3], IBM's question-answering computer system.

2.1.2 The first winter *(1974-1980)*

The first winter was defined by the hindrances found by the researchers while trying to develop anything related to artificial intelligence. The biggest of which was the computing power needed by the artificial intelligence algorithms, which simply did not exist at the time. Computers did not have enough memory to store the overwhelming amount of data required to build these complex rule-based systems, or just did not have enough computational power to solve problems fast enough.

Minsky and Papert (1969) may have played a huge part in this process. Strong critics of the "perceptron" (a machine learning algorithm used in binary classifiers) by the already Turing Award winner can have caused strong influence on the Artificial Intelligence to avoid researching deeper into neural networks – the state of the art in most areas today – and instead focus on the already declining symbolic methods.

The foundations of what we today call NP-Complete problems established by Cook (1971) and Karp (1972) would also help in the decline of the area by showing that many problems can only be solved in exponential time. This posed a risk to AI because it meant that some of the basic problems being solved by the era models would probably never be used in real-life data, where data is not represented by just a few data points.

[3] <https://www.ibm.com/watson>

2.1.3 The revival *(1980-1987)*

AI had been brought to life again in the early 1980s, mainly due to an increase of commercial interest in *expert systems* and to the agenda of the newly created Association for the Advancement of Artificial Intelligence conference (AAAI). Besides that, the funds that had gone missing in the first AI winter would also be back on the table, with the Japanese government funding AI research as part of their Fifth Generation Computer Project (FGCP). Some other countries would also restart their funding projects, like UK's Alvey project and DARPA's Strategic Computing Initiative.

After Minsky's criticism, connectionism would have a comeback in the early 1980s. Hopfield (1982) proved that what we today call a "Hopfield network" could learn in a different way than what it was being done before with perceptrons and simple artificial neural networks. Also, at the same time, Rumelhart, Hinton and Williams (1986) would popularize "Backpropagation": a new method to easily train and "backpropagate" the gradient in machine learning models.

2.1.4 The second winter *(1987-2000)*

Criticisms over the deployment of expert systems in real-world applications, however, may have caused the second AI winter (from the late 1980s to the early 2000s), which ended up ceasing AI research funding.

Hans Moravec wrote in (MORAVEC, 1988) that "it is comparatively easy to make computers exhibit adult level performance on intelligence tests or playing checkers, and difficult or impossible to give them the skills of a one-year-old when it comes to perception and mobility". This, with some contributions from Rodney Brooks, and Marvin Minsky, would emphasize what is now known as Moravec's Paradox: the idea that reasoning *per se* does not require much computation power, and can easily be thought/learned to/by a machine, but building an intelligent machine able to do what is "below conscience level for humans", i.e. motor skills, is what actually required enough computation power that did not yet exist at the time.

It is naive, however, to assume that nothing happened in this era. Campbell, Hoane and Hsu (2002) Deep Blue's greatest achievement – winning a Chess match with tournament rules against the then-reigning Chess champion Garry Gasparov – happened in 1997. Previously, in 1994, TD-GAMMON (TESAURO, 1994) program would show the

power Reinforcement Learning has by creating a self-teaching backgammon program able to play it at a master-like level. Also, although self-driving cars are usually considered recent technology, the ground for it was laid in this era, with Ernst Dickmmans's "dynamic vision" concept in (DICKMANNS, 1988; THOMANEK; DICKMANNS, 1995) where they had a manned car riding in a Paris' 3-lane highway with normal traffic at speeds of up to 130 km/h.

The late 1990s would also see an increase of research in information retrieval with the World Wide Web's boom, with research in web scrapers and AI-based information retrieval/extraction tools (FREITAG, 2000).

2.1.5 The groundbreaking present *(2000-present)*

The 2000s present us with AI's Renaissance, especially if we look at the impact of a specific AI subarea: Machine Learning (ML), but even more specifically its Deep Learning (DL) subfield. It was in this context that NeurIPS (at the time, NIPS) arose, again, as perhaps the most prominent AI conference, where several groundbreaking DL papers have been published, featuring convolutional neural networks, graph neural networks, adversarial networks, and other (deep) connectionist architectures.

In the early 2000's we would see AI reaching the end customer in most developed countries. iRobot[4] introduced its Roomba Robot Vacuum in 2002. Apple, Google, Amazon, Microsoft, and Samsung released Siri, Google Assistant, Alexa, Cortana, and Bixby, respectively, AI-based personal assistants capable of understanding natural language and executing a wide variety of tasks – admittedly, it did not work that well at the beginning, circa 2010, but it does work now.

Most achievements in the area since 2000 are related to DL, basing itself in the Artificial Neural Network (ANN) concept, a system that tries mimicking the way our brain cells work. This is not a new concept, as it was described in 1943 in (MCCULLOCH; PITTS, 1943), however, the immense computing power we have now allowed us to stack several layers of "neurons" one after the other – thus "deep" neural networks – and compute the results extremely fast. Also, given the natural parallelism of the process, the advent of GPUs created the necessary fertile ground for the explosion in deep models we have now.

Some of the most incredible recent achievements base themselves on something called Generative Adversarial Networks (GANs), a "framework for estimating generative

[4] <https://www.irobot.com/>

models via an adversarial process, in which we simultaneously train two models: a generative model G that captures the data distribution, and a discriminative model D that estimates the probability that a sample came from the training data rather than G" (GOODFELLOW et al., 2014). This framework is responsible for a wave of photo-realistic procedurally-generated content at the likes of <https://this-person-does-not-exist.com/en>, <https://thiscatdoesnotexist.com/>, <http://thiscitydoesnotexist.com/>, or the recursive <https://thisaidoesnotexist.com/>.

GANs are responsible for what we colloquially call "deepfakes" – a mash of "deep learning" with "fake". They work by superimposing one's face by another face, through the use of a machine learning model. Some more recent deepfakes can also alter the subject's voice, improving the experience. These are especially bad from an ethics standpoint when one imagines that these can be used to fake audio and images of influential people (HWANG, 2020). A thorough review of the area can be found in (NGUYEN et al., 2019).

A lot of the work produced in the Deep Learning field generated fruits, with 3 Turing Awards (for 5 different people) going to Artificial Intelligence researchers from 2010 onwards. Leslie Valiant won it in 2010, although his main articles date back to the '80s and '90s with his most important work "A theory of the learnable" (VALIANT, 1984) being published in 1984. Judea Pearl won it in the following year, 2011, for his "contributions [...] through the development of a calculus for probabilistic and causal reasoning" present in (PEARL, 1988) and (PEARL, 2009). The other three winners received their prizes in 2018: Hinton is most famously known for his Imagenet-winning CNN (KRIZHEVSKY; SUTSKEVER; HINTON, 2012), LeCun is also known for his work in text recognition in (LECUN et al., 1998) and (LECUN et al., 1989), while Bengio is mostly known for his collective work with LeCun and his recent works in GANs (GOODFELLOW et al., 2014) and neural translation (BAHDANAU; CHO; BENGIO, 2014). They are collectively known for (LECUN; BENGIO; HINTON, 2015).

Talking about games/e-sports, Google's AlphaGo won against the Chinese Go grandmaster Ke Jie in 2017[5], after having already won 4 out of 5 matches against the famous Go player Lee Sedol in 2016[6]. Also in 2017, OpenAI's Dota 2 bot[7] won a 1v1 demonstration match against the Ukrainian pro player Dendi, a huge demonstration of power in a game with imperfect information, with almost infinite possible future states. Later, in 2019, a new version of the same bot, called OpenAI Five, wins back-to-back

[5] <https://www.wired.com/2017/05/googles-alphago-continues-dominance-second-win-china/>
[6] <https://www.bbc.co.uk/news/technology-35797102>
[7] <https://openai.com/blog/dota-2/>

5v5 games against the then-world-champion Dota team, OG[8]. Also in 2019 DeepMind's AlphaStar bot reaches the biggest possible tier in Starcraft II[9].

It is impossible to talk about AI in recent years and not talk about the striking growth in submitted, and accepted, papers in the three biggest AI-related venues. Figure 3.4 shows we have over 1500+ papers in these conferences in recent years. For exact numbers, please check Table A.4. By checking the figure above it is also important noticing how Computer Vision arguably became the most important of the related areas, with CVPR having the biggest quantity of papers in their proceedings, thanks to the boom in image recognition and self-driving cars. We give more details of AI-related publications in Chapter 4.

2.2 The Turing Award

The annual ACM A.M. Turing Award is regarded as the highest prize a computer scientist can earn. It is conceded by the Association for Computing Machinery (ACM) to people with outstanding and lasting contributions to computer science and computing in general.

The prize was introduced in 1966 and named after the British mathematician Alan Turing. Turing influenced several different areas of computer science, formalizing the concepts of algorithms and computation with the Turing Machine (TURING, 1936). Turing is also considered by most as Artificial Intelligence's father after having proposed the Turing test to decide if a machine is "intelligent" or not (TURING, 1950). He is also known for his work in the Second World War, helping the British to decode the Nazi German Enigma machine with his *Bombe* machine, named after the Polish *bomba kryptologiczna* decoding machine. He died at age 41 from cyanide poisoning.

The prize was accompanied by a US$250,000 prize from 2007 to 2013, with financial support provided by Intel and Google (STAFF, 2007). Since 2014, however, the winners receive US$1 million, financed by Google (STAFF, 2014) for their exceptional achievement.

The prize has already been given to 62 different researchers in the most diverse areas of computer science research, both from a hardware and a software perspective. It has not ever been given to someone posthumously.

[8] <https://openai.com/five/>

[9] <https://www.theguardian.com/technology/2019/oct/30/ai-becomes-grandmaster-in-fiendishly-complex-starcraft-ii>

It is worth noting the bias present in the prize, however. The winners are chosen mostly from a US-centric view, as only 37%[10] of the winners were not born in the United States - and only 27%[11] of them credit a country different than the United States as the country where they did their main job. Also, the first woman to receive the prize, Elizabeth Allen, received the prize for her work on IBM's STRETCH computer only in 2006, 40 years after the first prize. Only two other women would eventually receive the prize.

For our work, the most interesting Turing Award Winners are those who had important contributions to the Artificial Intelligence field. The annual ACM A.M. Turing Award has given, since 1966, seven prizes to 11 different researchers due to their efforts in AI and related areas:

- **Marvin Minsky** (1969): *For his central role in creating, shaping, promoting, and advancing the field of Artificial Intelligence;*[12]
- **John McCarthy** (1971): *Dr. McCarthy's lecture "The Present State of Research on Artificial Intelligence" is a topic that covers the area in which he has achieved considerable recognition for his work;*[13]
- **Herbert Simon** and **Allen Newell** (1975): *In joint scientific efforts extending over twenty years, initially in collaboration with J. C. Shaw at the RAND Corporation, and subsequentially with numerous faculty and student collegues at Carnegie-Mellon University, they made basic contributions to artificial intelligence, the psychology of human cognition, and list processing;*[14]
- **Edward Feigenbaum** and **Raj Reddy** (1994): *For pioneering the design and construction of large scale artificial intelligence systems, demonstrating the practical importance and potential commercial impact of artificial intelligence technology;*[15]
- **Leslie Valiant** (2010): *For transformative contributions to the theory of computation, including the theory of probably approximately correct (PAC) learning, the complexity of enumeration and of algebraic computation, and the theory of parallel and distributed computing;*[16]

[10] Table G.1 lists every Turing Award winner correlating them with their country of birth

[11] See every author page in their ACM Turing Award website: <https://amturing.acm.org/byyear.cfm>

[12] Extracted from <https://amturing.acm.org/award_winners/minsky_7440781.cfm>

[13] Extracted from <https://amturing.acm.org/award_winners/mccarthy_1118322.cfm>

[14] Extracted from <https://amturing.acm.org/award_winners/simon_1031467.cfm>, and <https://amturing.acm.org/award_winners/newell_3167755.cfm>, respectively.

[15] Extracted from <https://amturing.acm.org/award_winners/feigenbaum_4167235.cfm>, and <https://amturing.acm.org/award_winners/reddy_9634208.cfm>, respectively.

[16] Extracted from <https://amturing.acm.org/award_winners/valiant_2612174.cfm>

- **Judea Pearl** (2011): *For fundamental contributions to artificial intelligence through the development of a calculus for probabilistic and causal reasoning;*[17]

- **Geoffrey Hinton**, **Yann LeCun**, and **Yoshua Bengio** (2018): *For conceptual and engineering breakthroughs that have made deep neural networks a critical component of computing.*[18]

The rationale for each prize above was transcribed from ACM's special Turing Award website[19].

2.3 CS Conferences

There are several conferences in the Computer Science field, but we had to narrow them down to the ones considered the most important to be able to properly analyze them. CSRankings is a metrics-based ranking of top computer science institutions around the world[20], which separates the works from each institution for each venue. They have a selection of conferences that they consider the important few: in this work, we will only ever focus on institutions available in their "AI" category, which are briefly described below.

Some of the abbreviations are actually from associations that happen to have a conference that, colloquially, received the same abbreviation.

Most of the credits for the research in the section below, more specifically the "most influential papers in the recent years" snippet, is due to <https://www.paperdigest.org/>.

2.3.1 IJCAI

The International Joint Conferences on Artificial Intelligence (IJCAI) was founded in California in 1969, being the first comprehensive AI-related conference to exist. The conference was held in odd-numbered years, but since 2016 the conference happens annually. It has already been held in 15 different countries, while the 2 most recent ones

[17] Extracted from <https://amturing.acm.org/award_winners/pearl_2658896.cfm>

[18] Extracted from <https://amturing.acm.org/award_winners/hinton_4791679.cfm>, <https://amturing.acm.org/award_winners/lecun_6017366.cfm>, and <https://amturing.acm.org/award_winners/bengio_3406375.cfm>, respectively.

[19] <https://amturing.acm.org/byyear.cfm>

[20] <http://csrankings.org>

were in Virtual Japan and Montreal-themed based. The next ones will be held in Austria (2022), South Africa (2023), and China (2024), increasing the number of countries that hosted the conference to 17 – China has already hosted it before.

Similar to AAAI, IJCAI is a comprehensive AI conference, encompassing all areas, with some recent publications in novel areas such as GNNs (ABBOUD et al., 2020) and transformers (CLARK; TAFJORD; RICHARDSON, 2020).

IJCAI has, over the year, published important papers from Turing Award winners such as (AVIN; SHPITSER; PEARL, 2005) and (VERMA et al., 2019).

2.3.2 AAAI

The Association for the Advancement of Artificial Intelligence (AAAI – pronounced *"Triple AI"*) was founded in 1979 under the name of American Association for Artificial Intelligence. This association is responsible for promoting one of the most important conferences in the AI field since 1980: the AAAI Conference on Artificial Intelligence. The conference used to be held once every 1 or 2 years, with a lack of conferences between 1990 and 1996, but it is been held yearly since 2010. It is worthy of note that although the conference has removed the "American" bit from its name, it has actually only been held in the USA and Canada (and remotely in 2021).

The conference has a pretty broad focus on AI without an outstanding subarea, so it has a lot of important papers published in the most recent years such as (BORDES et al., 2011) in Knowledge Bases, (XIA et al., 2014) in Information Retrieval, (HASSELT; GUEZ; SILVER, 2015) in Reinforcement Learning, and (LI et al., 2019) in Computer Vision and NLP.

2.3.3 NeurIPS (formerly NIPS)

The Conference and Workshop on Neural Information Processing Systems (NeurIPS) is a machine learning and computational neuroscience conference held every December, since 1987. It was already held in the USA, Canada, and Spain.

The Conference was once abbreviated as NIPS, but because of it being controversial and after the accusations of it being a hostile environment by some women attendees, their

board decided to change it to NeurIPS [21] in 2018.

CSRankings defines it as a "Machine Learning & Data Mining" conference, containing some important papers for the area, recently featuring GPT-3 (BROWN et al., 2020) and PyTorch's technical paper (PASZKE et al., 2019), which curiously have 31 and 21 authors, respectively. The sheer size of the company is incredible, with 2,334 papers accepted in 2021, outnumbering every other conference studied in this work.

2.3.4 CVPR

The Conference on Computer Vision and Pattern Recognition (CVPR) is an annual conference on Computer Vision and Pattern Recognition, regarded as one of the most important conferences in its field, with 1,294 accepted papers in 2019. It will in 2023, for the first time, be organized outside the United States in Vancouver, Canada. Of course, CVPR 1997 was held in Puerto Rico, an American ultramarine territory. It was first held in 1983 and has since 1985 been sponsored by IEEE, and since 2012 by the Computer Vision Foundation, responsible for providing open access to every paper published in the conference.

Being one of the most important Computer Vision venues it has seen some groundbreaking work in the past with research in novel areas such as Siamese Representation Learning (CHEN; HE, 2020), GANs (KARRAS; LAINE; AILA, 2018), and Dual Attention Networks (FU et al., 2018).

Turing Award Yann LeCun is a historical participant of the conference with works published in it on several occasions, e.g. (BOUREAU et al., 2010), (LECUN; HUANG; BOTTOU, 2004).

2.3.5 ECCV

ECCV stands for European Conference on Computer Vision, being CVPR's European arm – even though ECCV 2022 is actually going to be held in Tel Aviv - Israel, and not in Europe. It is held biennially every even-numbered year since 1990, when it was held in Antibes, France.

Even though it is considered CVPR's small sister, it had 1,360 accepted papers in

[21] <https://www.nature.com/articles/d41586-018-07476-w>

2019, also heavily focusing on Computer Vision with some publications of note such as RAFT (TEED; DENG, 2020), a model able to segment and predict image depth with high accuracy.

2.3.6 ICCV

Similar to ECCV, the International Conference on Computer Vision is CVPR's International arm, being held every odd-numbered year since 1987, when it was held in London, United Kingdom, been held in 14 other countries ever since.

1,077 papers made the cut in 2019, such as (SHAHAM; DEKEL; MICHAELI, 2019) who won the 2019's best paper award.

2.3.7 ACL

ACL is the Association for Computational Linguistics's conference held yearly since 2002, having surprisingly been held in 15 different countries in the last 20 years.

They define themselves on their website as "the premier international scientific and professional society for people working on computational problems involving human language, a field often referred to as either computational linguistics or natural language processing (NLP). The association was founded in 1962, originally named the Association for Machine Translation and Computational Linguistics (AMTCL), and became the ACL in 1968."

Commonly referred to as an NLP-related conference, it has some amazing work in recent years such as (STRUBELL; GANESH; MCCALLUM, 2019)'s work in investigating the environmental effects of creating these huge language models we have seen recently - such as (BROWN et al., 2020).

2.3.8 NAACL

NAACL is the conference held by the North American Chapter of the Association for Computational Linguistics, therefore also referred to as an NLP conference. The conference is actually named NAACL-HLT (or HLT-NAACL, sometimes) – North American Chapter of the Association for Computational Linguistics: Human Language Technologies.

It has been held since 2003, and it was co-located with ACL on the few occasions when ACL happened in North America. One of the most important papers using transformers in recent years was published there: the BERT model (DEVLIN et al., 2018).

2.3.9 EMNLP

EMNLP stands for Empirical Methods in Natural Language Processing. The conference started in 1996 in the US based on an earlier conference series called Workshop on Very Large Corpora (WVLC) and has been held yearly since then.

The recent conferences are marked by works trying to improve the BERT model (DEVLIN et al., 2018) already explained above, such as (JIAO et al., 2019), (FENG et al., 2020) and (BELTAGY; LO; COHAN, 2019) – the latter has also been published in ACL.

2.3.10 ICML

ICML is the International Conference on Machine Learning, the leading international academic conference focused on machine learning. The conference is held yearly since 1987, with the first one being held in 1980 in Pittsburg, USA. The first few conferences were all held in the United States, but the 9th conference, in 1992, was held in Aberdeen, Scotland, United Kingdom. Since then it has been held in 10 other countries, and twice virtually because of the COVID-19 pandemic.

It contains some seminal papers in Machine Learning from PASCANU; MIKOLOV; BENGIO and IOFFE; SZEGEDY, and some more recent excellent research like (ZHANG et al., 2018) and (CHEN et al., 2020). Besides Bengio's aforementioned seminal paper, his Turing Award co-winner Hinton also published important papers in ICML (NAIR; HINTON, 2010).

2.3.11 KDD

The SIGKDD Conference on Knowledge Discovery and Data Mining is an annual conference hosted by ACM, which had its first conference in 1989's Detroit. Although it is usually held in the United States, it has already been hosted by a few other countries, namely Canada, China, France, and United Kingdom.

It is the most important conference encompassing the Knowledge Discovery and Data Mining field, with 394 accepted papers in 2021: a smaller number if we compare with the other conferences we investigate in this paper, but a huge achievement nonetheless.

The conference recent years have seen a lot of presence of Artificial Intelligence, mostly defined by Graph Neural Networks (GNNs), with works from QIU et al., WU et al., JIN et al., and LIU; GAO; JI, all of them accepted in 2020's SIGKDD. It is interesting to note how all of these authors with influential papers in this 2020 conference are from Chinese territory, a preview of what's to come – see some insights about it in Section 4.6.

2.3.12 SIGIR

SIGIR stands for Special Interest Group on Information Retrieval, an ACM group. It has its own annual conference that started in 1978 and has happened every single year since then. It is considered to be the most important conference in the Information Retrieval (how to acquire useful and organized information from raw, unorganized, and unstructured data) area.

After 43 editions, it has been hosted in 21 different countries. It used to alternate between the USA and a different country, but this rule does not follow anymore, with only one conference in the US in the last 8 years.

A lot of the work in recent years is focused on recommender systems such as (HE et al., 2020), (WU et al., 2021), and (WANG et al., 2020).

2.3.13 WWW

The Web Conference (WWW) is one of the top internet conferences in the world. It "brings together key researchers, innovators, decision-makers, technologists, businesses, and standards bodies working to shape the Web"[22]. It is a yearly event that started, obviously, at CERN in Geneva, Switzerland, in 1994.

The conference heavily focuses on Semantic web and Data mining with some important results in recommender systems as well.

[22]<https://dl.acm.org/conference/www>

2.4 Graphs and their centralities

A graph G is represented by a tuple $G = (V, E)$, where V is a set of nodes (vertexes) and E a set of edges $e_{u,v}$ connecting nodes u to v where $u, v \in V$. These edges can be directed or undirected – thus making us able to differentiate between directed and undirected graphs. In the directed case of $e_{u,v}$ we call u as being the source node and v the destination node. We will always use n to represent the number of nodes in a graph, and m to represent the number of edges in it.

Also, a pair of nodes (u, v) might have more than one edge connecting them: in this case, we call the graph a multigraph. Similarly, these edges might have a weight w making the graph a weighted graph.

Furthermore, we can also have labeled graphs, where nodes and edges can be of different types. These are useful in knowledge representation systems, such as the graph built in Section 3.3.4.

We call $p = u_1, u_2, ..., u_p$ a path between u_1 and u_p in G if $\exists\, e_{u_i,u_i+1}\, \forall\, 1 <= i <= p - 1$. Basically, we have a path if we can go from node u_1 to u_p through a sequence of connected edges. We can also define a shortest path between a pair of nodes (u, v) as the path with the minimum possible quantity of intermediate nodes – note, however, that we can have more than one shortest path between any pair of nodes (u, v).

2.4.1 Centralities

The existence and interest in Graph centrality measures date back to the 1940s, but it was more formally incorporated into graph theory in the 1970s (WAN et al., 2021; FREEMAN, 1978). A fundamental motivation for the study of centrality is the belief that one's position in the network impacts their access to information, status, power, prestige, and influence (WAN et al., 2021). Therefore, throughout this work when we want to identify the above concepts we will use graph centralities for the different networks we built.

Sections 2.4.1.1 to 2.4.1.4 describe the most important graph centralities in the literature which are used throughout this work. Then in Section 2.4.1.5 we go over some other centrality measures for completeness' sake. Although we did not use these, they have some merit and have the ability to provide interesting insights into this and are, therefore, present as options for future work.

2.4.1.1 Degree Centrality

We represent the degree of a node u as k_u meaning the number of other nodes connected to this node. In a directed graph we can further split this metric into two: k_u^{in} is the in-degree, representing the number of nodes $v \in V$ that have an edge $e_{v,u}$ with v as source and u as destination (i.e. number of nodes with an edge pointing to u); the opposite metric k_u^{out} is the out-degree, representing the number of nodes $v \in V$ that have an edge $E_{u,v}$.

Therefore, it is easy to extend this metric to a centrality called **Degree Centrality** defined as:

$$\mathcal{C}_{Dg}(u) = \frac{k_u}{n-1}, \tag{2.1}$$

where n represents the number of nodes V in the graph G.

Also, the same way we have in-degree and out-degree metrics, we can extend Equation 2.1 and define **In-Degree Centrality** and **Out-Degree Centrality**, respectively:

$$\mathcal{C}_{Dg_{in}}(u) = \frac{k_u^{in}}{n-1} \tag{2.2}$$

$$\mathcal{C}_{Dg_{out}}(u) = \frac{k_u^{out}}{n-1} \tag{2.3}$$

These degree metrics are used to identify how well a node is directly connected to other nodes, without considering the influence a node can pass to its neighbors.

2.4.1.2 Betweenness Centrality

The **Betweenness Centrality** was defined in (FREEMAN, 1977), and its measure of importance of a node u is how many shortest paths in the graph go through u. It is defined as

$$\mathcal{C}_B(u) = \frac{\sum_{s \neq u \neq t} \frac{\partial_{s,t}(u)}{\partial_{s,t}}}{(n-1)(n-2)/2} \qquad \forall s, u, t \in V, \exists\, e_{s,t} \tag{2.4}$$

where $\partial_{s,t}(u)$ is the number of shortest paths between s and t that go through u, and $\partial_{s,t}$ is simply the number of shortest paths between s and t. Note that we are only ever counting paths between the pair (s,t) if there is a path between (s,t).

Betweenness is related to the notion of connectivity, where a node with a bigger betweenness actually means that it is a point of connection between several nodes. In a graph with a single connected component, a node can have the highest betweenness if it works as a bridge between two individually disconnected components. It is regarded as a measure of a node's control over communication flow (FREEMAN, 1978), (CARDENTE, 2012).

2.4.1.3 Closeness Centrality

Closeness Centrality was created in (SABIDUSSI, 1966) representing the closeness of a node with every other node in the graph. It is the inverse of the farness which in turn is the sum of distances with all other nodes (SAXENA; IYENGAR, 2020). It is defined by

$$\mathcal{C}_C(u) = \frac{n-1}{\sum_{v \neq u} d(u,v)} \qquad \forall u, v \in V \tag{2.5}$$

where $d(u,v)$ is the distance between the nodes u and v. This distance is simply the number of edges in the shortest path p between the pair (u,v) if the graph is unweighted, while it is the sum of every edge in the path in case the graph is unweighted.

Note that because distance is not defined between every pair of nodes in disconnected graphs (a graph where not every node can be reached from another node) we can't compute closeness for disconnected graphs.

A node with a higher closeness indicates that the node is in the middle of a hub of other nodes. It also means that a node with big closeness values is "closer", on average, to the other nodes, hence closeness. It represents the node's level of communication independence (FREEMAN, 1978), (CARDENTE, 2012).

2.4.1.4 PageRank Centrailty

Pagerank is a global centrality measure that needs the entire network to measure the importance of one node. It measures the importance of one node based on the importance

of its neighbors. (SAXENA; IYENGAR, 2020). It was developed by BRIN; PAGE when they were creating Google, and it is the underlying method behind their search engine.

To understand Pagerank, we need to understand that its main idea is to understand how important a web page is in the middle of all the other millions of pages on the world wide web. The main idea behind it is that we are considering a web page important if other important web pages link to it.

Think about it as if we had a webcrawler randomly exploring the web and increasing a counter every time we enter into a specific page. Then, when you are on a page you either have the option to click on one of the links on the page or go to a random page on the web with probability $0 <= q <= 1$ – this is useful both to model real-life where we simply go to random websites and also to mimic pages without any out link. The usual value for q, also called teleportation or damping factor, is 0.15, as defined in the original paper. Therefore, with this thought in mind, we can define **Pagerank** as

$$\mathcal{C}_{PR}(u) = \frac{q}{n} + (1-q)\sum_{v} \frac{\mathcal{C}_{PR}(v)}{k_v^{out}} \qquad \exists e_{u,v} \in E \tag{2.6}$$

The equation above illustrates how this process is iterative because we depend on the Pagerank of every neighbor to be able to compute our own Pagerank. The process usually converges or can be stopped after a certain number of iterations.

2.4.1.5 Other centralities

There are other useful centralities present in the literature. As explained before they were not used in our work, but they would ideally be used in future work using the dataset created. Most of these were extracted from similar lists in (SAXENA; IYENGAR, 2020) and (WAN et al., 2021).

- **Semi-Local centrality** (CHEN et al., 2012) defines a metric similar to the degree centrality where we expand it to 2 levels of neighbours.

$$\mathcal{C}_{SL}(u) = \sum_{v \in N(u)} \sum_{w \in N(v)} d_2(w), \tag{2.7}$$

where $\mathrm{d}_2(w)$ is the number of neighbors plus the number of neighbors for every

neighbor of w – basically how many nodes you can reach in two steps.

- **Volume Centrality** (WEHMUTH; ZIVIANI, 2013) is a kind of generalization from the above centrality parameterizing how far a node influence can reach and is defined by

$$\mathcal{C}_V(u) = \sum_{v \in \tilde{N}_h(u)} k_v, \tag{2.8}$$

where $N_h(u)$ is the set of neighbors within a distance h of u, and $\tilde{N}_h(u) = N_h(u) \cup \{u\}$. WEHMUTH; ZIVIANI demonstrated that $h = 2$ results in a good trade-off of identifying nodes with important relations and the cost of computing this relationship.

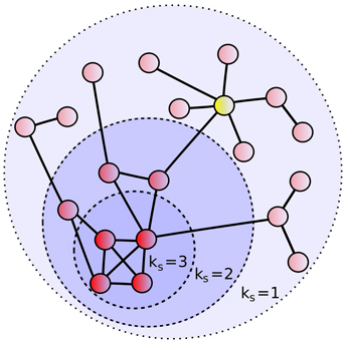
- **H-index** (HIRSCH, 2005) is a well-known statistic in the research world, being exhibited as a statistic in most research-aggregator portals such as Google Scholar and DBLP. HIRSCH defined that h is the highest integer value for which the author has h papers with at least h citations.

- **Coreness Centrality** (KITSAK et al., 2010) represents the idea that the important nodes are at the core of a graph. It can be determined by the process of assigning each node an index (or a positive integer) value derived from the k-shell decomposition. The decomposition and assignment are as follows: Nodes with degree k= 1 are successively removed from the network until all remaining nodes have a degree strictly greater than 1. All the removed nodes at this stage are assigned to be part of the $k$shell of the network with index k_S= 1 or the 1-shell. This is repeated with the increment of $k$to assign each node to distinct k-shells(WAN et al., 2021). See Figure 2.2 to see an example of definition of k-shells. Then, we can mathematically define this centrality as

$$\mathcal{C}_k(u) = \max\{k | u \in H_k \subset G\}, \tag{2.9}$$

where H_k is the maximal subgraph of G with all nodes having a degree of at least k in H(WAN et al., 2021).

Figure 2.2 – Example of a k-shell assignment



Source: (TANASE et al., 2015)

Some more complex centralities mostly use the fact that we can define a graph by its adjacency matrix $\mathbf{A}$ and its corresponding eigenvalues and eigenvectors. Because this work is not a literature review we will not include them.

2.5 Related Work

One of the main motivations behind this work is the fact that the history of artificial intelligence and its dynamic evolution has not been researched enough: (XU et al., 2019) focused specifically on "explainable AI" evolution (or de-evolution, in this case); (OKE, 2008) does deepen its work in several different AI areas, with a thorough review of each area, but it does not go back in history further than the mid-1990s; (MIJWIL; ABTTAN, 2021) does a great job of explaining recent research on AI in general and tries predicting what we can expect from it in the next few years, similar to (MARCUS, 2020)'s look at the future.

There are also similar approaches to investigate author citation/collaboration networks such as (DING et al., 2010; GUNS; LIU; MAHBUBA, 2011; ABBASI; HOSSAIN; LEYDESDORFF, 2012; CARDENTE, 2012; WU et al., 2019), mostly focusing in the betweenness centrality. Wartburg, Rost and Teichert (2022) use closeness to analyze patent networks. Also, Krebs (2002) show how centrality measures can be used to identify proeminent actors from the 2001 Twin Tower's terrorist attackers network.

In the country affiliation in papers, Grubbs, Glass and Kilmarx (2019) investigated coauthor country affiliation in Health research funded by the US National Institute of Health; Michalska-Smith et al. (2014) go further by trying to correlate country of affiliation with the acceptance rate in journals and conferences; Yu et al. (2021) studied how one

can infer the country of affiliation of a paper from its data in WoS[23]; Hottenrott, Rose and Lawson (2019) investigates the rise on multi-country affiliations in articles as well.

[23]<https://www.webofknowledge.com/>

3 METHODOLOGY

This chapter explains how our work was developed, explaining where our underlying data was extracted from in Section 3.1, throughly describing the artifacts generated in this work along with the algorithms and data structures used to build them in Section **??**.

3.1 Underlying Dataset

The most extensive bibliography of computer science publications is the DBLP Database (DBLP, 2019), available at <https://dblp.uni-trier.de/>, recently (in February 2022), it surpassed the 6 million publications mark (See Figure 3.1), containing works from almost 3 million different authors. Figure 3.2 shows how large is the increase in publications in the recent years, per DBLP's statistics page[1]. They provide a downloadable 664MB GZipped version of their dataset in XML format[2]. Recently (after this work had already been started and was past the dataset choosing process), DBLP has also released its dataset in RDF format[3]. However, because their dataset has its pitfalls, such as duplicated authors and/or incorrectly merged authors, we opted to not use their dataset directly.

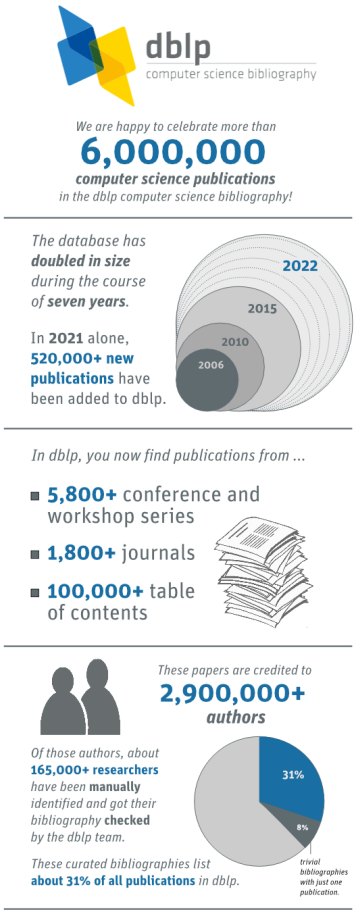
Instead, in our work, we used Arnet's (TANG et al., 2008) V11[4] paper citation network, which dates from May 2019. It contains 4,107,340 papers from DBLP, ACM, MAG (Microsoft Academic Graph), and other sources, including 36,624,464 citation relationships. This dataset contains more information than DBLP's, as they better worked on author disambiguation (merging authors DBLP considered to be different ones, or separating authors DBLP considered to be the same person), providing us the ability to generate truther collaboration/citation networks.

It is important to clarify why we are using Arnet's v11 dataset instead of one of their newer datasets, namely v12 and v13 – the latter, from May 2021, contains 5,354,309 papers and 48,227,950 citation relationships, an increase of 30.3% compared to v11. First, and foremost, this work started in 2019, when versions v12 and v13 were not available yet. Also, when these newer datasets were made available, we did try to use both of them, but we faced some problems that prompted us back to the v11 dataset:

1. v12's and v13's data format is different from v11's. The format of v12 and v13 is a

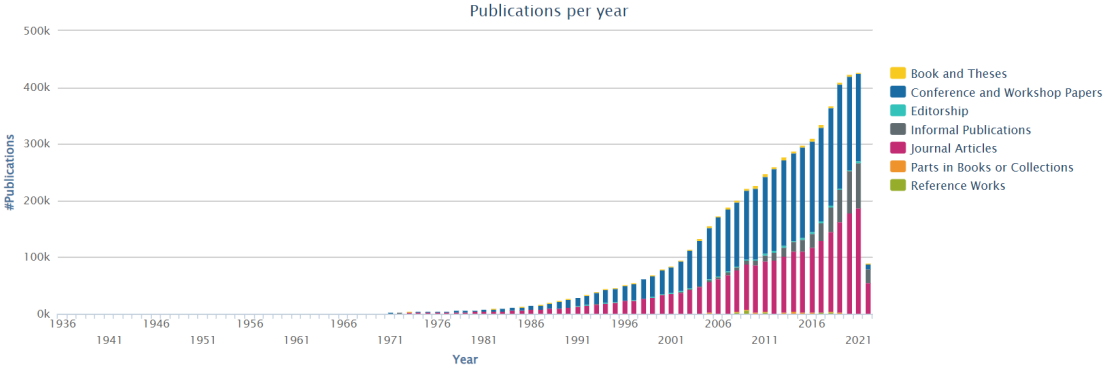
[1] <https://dblp.org/statistics/index.html>
[2] <https://dblp.org/xml/release/>
[3] <https://blog.dblp.org/2022/03/02/dblp-in-rdf/>
[4] <https://lfs.aminer.cn/misc/dblp.v11.zip>

Figure 3.1 – Excerpt of a DBLP poster acknowledging their 6 million papers mark



Source: <https://blog.dblp.org/2022/02/22/6-million-publications/>

Figure 3.2 – DBLP papers per year



Source: <https://dblp.org/statistics/publicationsperyear.html>

fully-fledged 11GB XML file, which required us to write a new Python library to convert from XML to JSON (our storage method) without loading the whole file into memory by streaming-converting it (see Section H.1). Besides the file being harder to read and handle, the new format also changed the IDs from an integer to a UUID-based value, causing us to rewrite the whole logic that was able to detect papers from the main AI conferences based on their past integer values.

2. There are fewer papers from the AI conferences of interest for this work. Even though we have 30% more papers in the most recent version, after carefully finding out which are the new IDs for the conferences, we could only find 58490 papers out of the 89102 (65%) present on version v11. As a smoke test, we did reduce our test only for the main AI conferences (AAAI, NeurIPS, and IJCAI): we could manually count 42082 papers in these 3 conferences – and this is a lower bound because we could not find the count of papers in some years for AAAI and IJCAI; v11 and v13 have 41414 and 20371 of them, respectively. We also tried finding the AI Conferences by name instead of IDs (at cost of some false positives) but it did not work, also finding only 20929 papers. This shows how we have twice the data in v11 compared to v13 instead of 30% more in v13 as expected.

3. Missing data in the most recent years. Even though v13 should have data until 2021, there are only a few hundred papers for the main AI conferences in 2019, 2020, and 2021, while in reality there should be 12559 of them.

All of the data compiled to build the points above can be seen in Table 3.1, and Figure A.4. Table A.4 has the raw data used to build Figure A.4, where "?" data points were considered to be 0 for the sake of simplicity. An interesting statistical information one might get from Figure 3.3 is the fact that even though IJCAI used to happen only in odd-numbered years, even-numbered years do not have any noticeable NeurIPS and AAAI paper acceptance rates increase.

Section 4.6.1 shows some charts where it can be seen how degraded our data looks if we had used *v13* instead of *v11*.

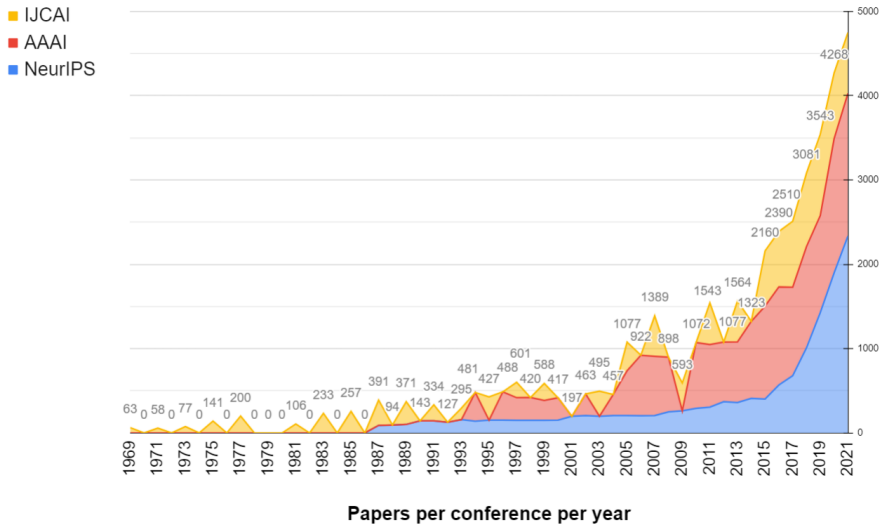
Arnet's v11 format is a variation of a JSON file with some improvements to make it easier to load the data in memory without having to load the whole file. Every line is a valid JSON object, requiring us to simply stream the file, iterating over every line, parsing the JSON file, keeping only the required information in memory, and immediately send the JSON file to be garbage collected, using no more than 8kb of memory to read the entire

Table 3.1 – Comparison of paper counts with different methods

	AI Conferences Total
Manual Count	42082
v11	41414
v13 detecting conferences by ID	20371
v13 detecting conferences by name	20929

Source: The Author

Figure 3.3 – Manual paper count per year in AAAI, NeurIPS and IJCAI



Source: The Author

file.

Every JSON object in this file follows the structure defined in Table 3.2. We, then, for most of the work, keep only the fields tagged with an asterisk (∗). Also, a question mark symbol (?) indicates the field is optional and is, sometimes, not present in the data provided by Arnet. Figure A.1 shows an example of such JSON entry, depicting (GLOROT; BENGIO, 2010)'s representation in the dataset.

Figure 3.4 shows some raw insights about this dataset, using the conferences defined in Section 2.3. It shows that all conferences have seen an increasing trend in the number of papers in the last few years, specially CVPR and AAAI.

3.2 Artifacts

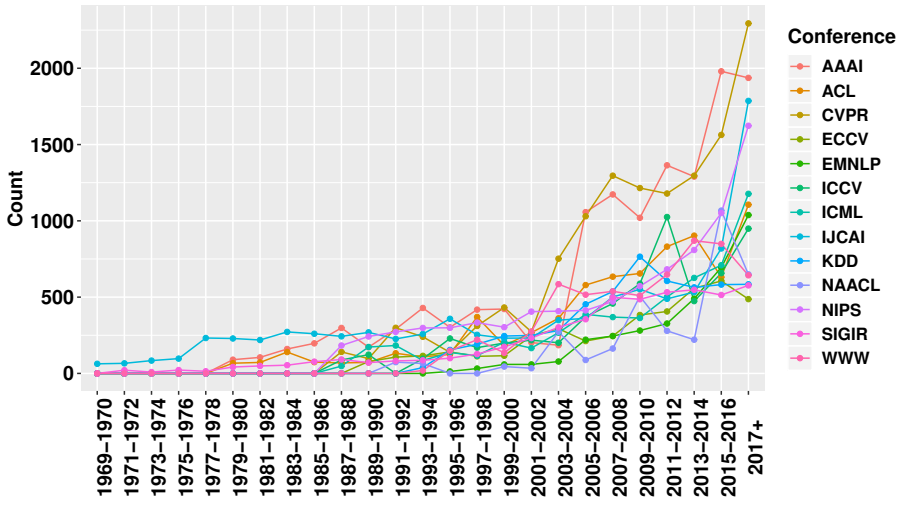
The code used to download the data, parse the dataset, and generate the graphs, analyses, and charts present in this work is available at <https://github.com/rafaeelaudibert/TCC/tree/v11> in Github. The code for this work is in branch *v11*. The *master* branch

Table 3.2 – Data structure for a single entry in the Arnet JSON dataset

Field Name	Type	Description
id*	*string*	Unique identifier for the paper
title*	*string*	Paper title
authors*	*Author[]* (See Table A.1)	List of every single author
venue*	*Venue* (See Table A.2)	Object with data about the venue
year*	*integer*	Year of publication
n_citation	*integer*	Citation number
page_start?	*string*	Paper start page in the Proceedings/Book/Journal
page_end?	*string*	Paper end page in the Proceedings/Book/Journal
doc_type	*string*	Place of publication
publisher?	*string*	Book/Journal publisher
volume?	*string*	Book volume
issue?	*string*	Journal issue
references*	*string[]*	List of ids this paper references
indexed_abstract*	*IndexedAbstract* (See Table A.3)	Inverted index holding data about the paper abstract

"" indicates the field was used in this work*
"?" indicates the field is optional
Source: The Author

Figure 3.4 – Number of papers per conference per year.



Source: The Author

contains the code used when we were trying to parse Arnet's *v13* dataset, which did not work out as explained in the previous section.

All the data analysis was built using Python, with the help of some open-source third-party libraries (See Table B.1) available in PyPi.

For the most complex plots, Python was not the right tool for the job, so they were built using R and its built-in counterparts for *matplotlib*, *numpy* and *seaborn*. Unfortunately, the code for these graphs is not available anymore because it was lost during a disk formatting procedure.

3.2.1 Graph Datasets

Throughout this work, we assembled 5 new datasets, modeled in a graph structure, which are briefly described below. A thorough explanation can be found in each respective section below.

Author Citation Graph (*ACi*) Directed multigraph, where every author is a node, with edges representing citations.

Author Collaboration Graph (*ACo*) Undirected graph, where every author is a node, with edges representing co-authorship

Paper Citation Graph (*PC*) Directed graph, where every paper is a node, with edges presenting citations.

Author-Paper Citation Graph (*APC*) Directed labeled graph, where nodes can be an author or a paper, and we can have edges between papers (citation) or between authors and papers (authorship).

Country Citation Graph (*CC*) Directed multigraph, where each node represents a country of origin, and edges represent citations.

As our work is focused on the flagship AI and adjacent fields conferences, we filtered their dataset to contain only the papers published in these conferences to build ours. The chosen conferences were based on CSRankings (CSRANKINGS, 2019) top-ranked AI conferences, which include the following fields: Artificial Intelligence, Computer Vision, Machine Learning & Data Mining, Natural Language Processing, and The Web & Information Retrieval. For each of the graphs explained above, we calculated the following exact centralities: degree (in and out) (Section 2.4.1.1), betweenness (Section 2.4.1.2), closeness (Section 2.4.1.3), and PageRank (Section 2.4.1.4).

For our work, we created the cumulative graph for each year from 1969 (the first IJCAI conference) until 2019, i.e. the cumulative graph for the year 2000 contains all the papers before and including 2000. A graph for each individual year from 1969 to 2019 was also created, to help with the analysis presented in the sections below. The cumulative graphs containing all the data, including exact centralities, were made available at <https://github.com/rafaeelaudibert/conferences_insights_database>. The cumulative graphs for the entire Authors Citation dataset, not restricting it by conference, were also

Table 3.3 – Graph Statistics for the cummulative data

	Graph	*Nodes*	*Edges*
CS Conferences	ACi	104179	5654596
	ACo	104179	621644
	PC	89102	486373
	APC	193281	759386
	CC	93	4776703
Full DBLP	ACi	3655049	210362459

Source: The Author

made available in the same repository, without computing the centralities. We can find the statistics for the size of each graph dataset in Table 3.3.

3.3 Types of Graphs

The graphs were built in Python using *networkx* (HAGBERG; SWART; CHULT, 2008) which provides an easy interface to build various types of graphs, including multi-graphs with directed edges, which we routinely use.

All graphs below are based on the data shown in Figure 3.5.

3.3.1 Author Citation Graph

This is a directed multi-graph, where every author is a node. An edge $e_{u,v}$ represents a paper from author u having a citing to a paper by author v. As author u can have more than one paper citing a paper by author v there might be more than one edge between the nodes, therefore we have a multi-graph. Also, authors might cite another paper from themselves, therefore we might have self-loops.

Because of the way our data is organized, when we are iterating over the papers we have only the id of the papers that were referenced, but not the ID of the authors in the other papers. So, we first create a hash table with keys as the papers IDs and the value as the authors of that paper. We use this as a lookup table to identify which authors should be connected when we are iterating over the papers. See Algorithm 2 to see how this works when building the graph.

The above means that we first need to iterate over all papers and create this huge lookup table. In practice, because you can't cite papers that haven't yet been published, we split the papers into buckets by the year they were published, and iterate in ascending years,

Figure 3.5 – Sample data for graphs

```
[
  {
   id: '1',
    title: 'Survey about Graphs',
    authors: [
      { id: '1', name: 'John Doe', org: 'MIT' }
    ],
    venue: { raw: 'Some Conference' },
    references: [],
    year: 1967
  },
  {
   id: '2',
    title: 'Survey about Bigger Graphs',
    authors: [
      { id: '2', name: 'Mary Jane', org: 'UFRGS' },
      { id: '3', name: 'Jane Carl', org: 'TU KL' },
    ],
    venue: { raw: 'Some Conference' },
    references: ['1'],
    year: 1970
  },
  {
   id: '3',
    title: 'Survey about Huge Graphs',
    authors: [
     { id: '2', name: 'Mary Jane', org: 'UFRGS' }
    ],
    venue: { raw: 'Some Conference' },
    references: ['1', '2'],
    year: 2003
  }
]
```

Source: The Author

Algorithm 1 Bucket-splitting paper per year

Require: L ▷ List of papers such as the example in Figure 3.5

papers_per_year ← empty hashtable

for year = 1969...2018 **do**
    papers_per_year[year] ← empty list
end for

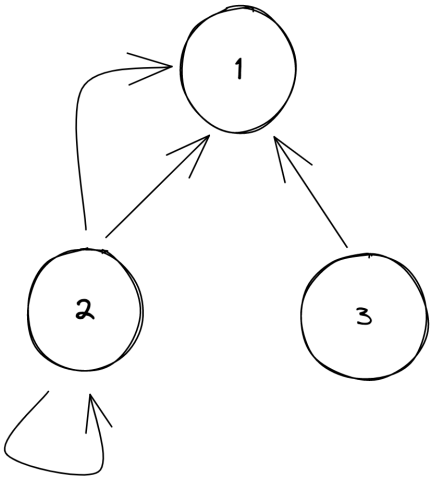
for paper ∈ L **do**
    papers_per_year[paper.year] ≪ paper ▷ ≪ means append
end for

return papers_per_year

Algorithm 2 Author Citation Graph

```
Require: papers_per_year                    ▷ Hash table as returned by Algorithm 1
  G ← new graph with empty V and E
  old_papers ← {}

  for year = 1969...2018 do
      papers ← papers_per_year[year]
      for paper ∈ papers do
          old_papers[paper.id] ← id of every author in paper.authors
      end for

      for paper ∈ papers do
          for author ∈ paper.authors do
              G.V ← G.V ∪ {author.id}
          end for

          for citation_id ∈ paper.references do
              if citation_id ∈ old_papers.keys then
                  for cited_author ∈ old_papers[citation_id] do
                      for author ∈ paper.authors do
                          G.E ← G.E ∪ { (author.id, cited_author.id) }
                      end for
                  end for
              end if
          end for
      end for
  end for

  return G
```

Figure 3.6 – Example of author citation graph



Graph generated given the input data from Figure 3.5
Source: The Author

which will make us keep only the "past" papers in this hash table. Algorithm 1 shows the year bucket-splitting algorithm and Algorithm 2 shows how we build this graph, with this more efficient hash table where at any year y we only have papers from years $i <= y$ in the hash table. Although at the end of the process the table has the same size as it would have if we had built it from the beginning, this method increases local consistency improving cache results when we are iterating over the first years making this process more efficient.

Note that we might not have data for the cited paper because we are filtering the data out for only a few conferences. In this case, we simply do not add this paper.

The most recent version of the code for this graph generation process can be found in <https://github.com/rafaeelaudibert/conferences_insights/blob/v11/graph_generation/generate_authors_citation_graph.py>.

Figure 3.6 shows an example of such graph, given the input data from Figure 3.5.

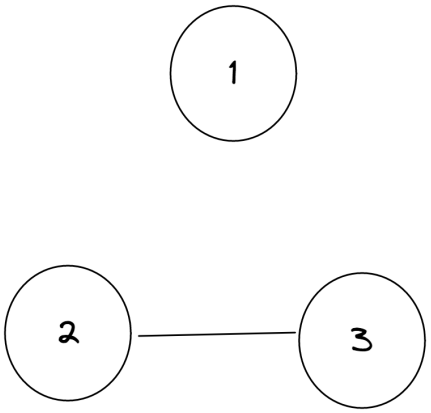
3.3.2 Author Collaboration Graph

This is an undirected graph, where every author is a node. In this graph, an edge $e_{u,v}$ represents that u and v worked together in at least one paper.

This graph is easier to generate compared to the Author Citation Graph (Section 3.3.1) because data is local and we do not need to iterate twice over the data to generate a lookup table: we can simply iterate over all papers and then connect all co-authors in a clique.

The most recent version of the code for this graph generation can be found

Figure 3.7 – Author collaboration example graph



Graph generated given the input data from Figure 3.5
Source: The Author

in <https://github.com/rafaeelaudibert/conferences_insights/blob/v11/graph_generation/generate_collaboration_graph.py>.

Figure 3.7 shows an example of such graph, given the input data from Figure 3.5.

3.3.3 Paper Citation Graph

This is a directed graph, where every paper is a node. A directed edge $e_{u,v}$ means that paper u cited paper v. Similar to the Authors Citation Graph we need to create a lookup table, using the same incremental procedure of loading in the lookup table data only for years $i <= y$ when iterating over year y.

The most recent version of the code for this graph generation can be found in <https://github.com/rafaeelaudibert/conferences_insights/blob/v11/graph_generation/generate_citation_graph.py>.

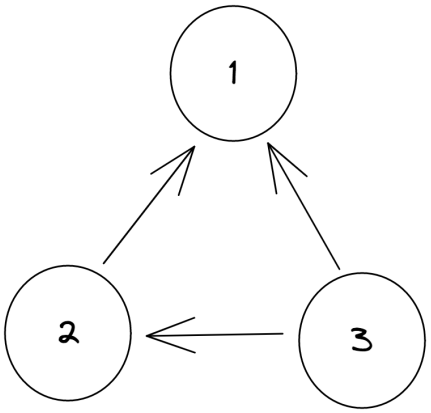
Figure 3.8 shows an example of such graph, given the input data from Figure 3.5.

3.3.4 Author-Paper Citation Graph

This is a directed labeled graph, where nodes can be either an author or a paper, and we can have edges between papers or between authors and papers, therefore this graph is more complex than the previous ones because it can represent both a paper citation network and an author citation network (through intermediate paper nodes).

This graph is built based on the *Papers Citation Graph*, with the already existent

Figure 3.8 – Paper citation example graph



Graph generated given the input data from Figure 3.5
Source: The Author

nodes being from the type *paper* V_P, and the already existent edges being from the type *citation* E_C. After, we add a node with type *author* V_A for each author, with a directed edge with type *authorship* E_A for each paper they authored.

This graph is ideal for a full picture of the data, with the possibility of inferring every possible interaction in it. Therefore, it is an ideal representation for knowledge representation tasks or even recommender systems. This is discussed in more detail in Section 5.3.

The most recent version of the code for this graph generation can be found in <https://github.com/rafaeelaudibert/conferences_insights/blob/v11/graph_generation/generate_authors_and_papers_graph.py>.

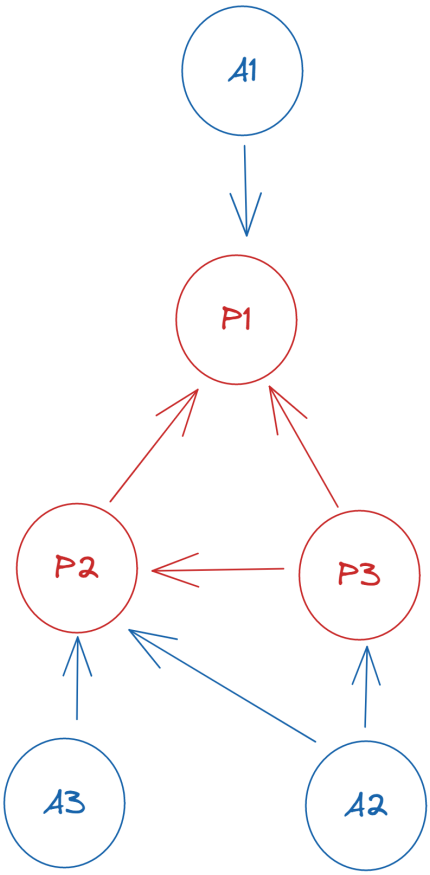
Figure 3.9 shows an example of such graph, given the input data from Figure 3.5.

3.3.5 Country Citation Graph

This is a directed multigraph, where each node represents a country, and an edge $e_{u,v}$ represents that an author from country u cited an author from country v in a paper. Because of this two nodes might have many edges between them.

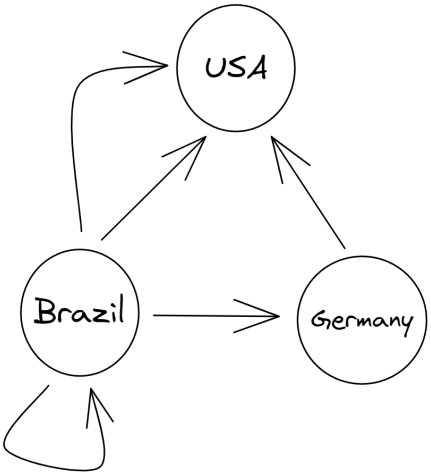
After we have figured out which country an author is from (Details in Section 3.3.5.1) we can create this graph by doing the same procedure for the citation graph. Save the papers already existing by that time in a lookup table; iterate over every paper; iterate over the citations; iterate over the current paper authors and the cited paper authors; connect the country they belong to with an edge. It is possible (and quite common) to create self-loops.

Figure 3.9 – Author Paper Citation example graph



Graph generated given the input data from Figure 3.5
Red nodes indicate they have type V_P*;*
Red edges indicate they have type E_C*;*
Blue nodes indicate they have type V_A*;*
Blue edges indicate they have type E_A*;*
Source: The Author

Figure 3.10 – Countries citation example graph



Graph generated given the input data from Figure 3.5
Source: The Author

The most recent version of the code for this graph generation can be found in <https://github.com/rafaeelaudibert/conferences_insights/blob/v11/graph_generation/generate_country_citation_graph.py>.

Figure 3.10 shows an example of such graph, given the input data from Figure 3.5, in addition to the following mapping from organizations to countries: MIT[5] → USA; UFRGS[6] → Brazil; TU KL[7] → Germany.

3.3.5.1 Affiliation x Country mapping

It is important to note that the Arnet v11 data we collected does not always provide the country of an author in its "org" field, containing only the organization they belong to – it sometimes doesn't even provide the organization – which poses a problem.

The "organization" field present in the data is in free-form format, which means that it does not have a clear structure from which we can extract the country of an author. Even worse, it might not even be a university name, as both companies and non-affiliated individuals can have papers in flagship venues. There is some structure in it for most of the data, though, so we have developed a pipeline where we iteratively try to detect an organization's country of origin.

In our pipeline, we first preprocess the organization by following Algorithm 4 removing cluttering and using only the text after the last comma – ideally where the

[5] <https://www.mit.edu/>
[6] <https://www.ufrgs.br>
[7] <https://www.uni-kl.de/>

country of affiliation should be. After, Algorithm 3 is followed. We try matching the text against a lookup table that maps organizations to countries. If there's a miss, we split the text into spaces and try matching only the first word to the table, and after only the last word. If that still does not work we try matching the text without the preprocessing step.

In the end, if everything fails, we check if we matched that author previously. That is our last resort because remember that the author might change organization (and even country) throughout their academic career, so we can't trust an author will still be in the same organization as they were the last time they published something.

Algorithm 3 Organization to Country Mapping

```
Require: raw_org                          ▷ Organization name
Require: org                              ▷ Organization name preprocessed by Algorithm 4
Require: author_id
Require: T                                ▷ Lookup table matching organization to country
Require: PT                               ▷ Past author to organization matchings
  if org ∈ T.keys then                    ▷ Check if preprocessed org is in the table
      return T[org]
  end if

  split_org ← split("org", " ")           ▷ Split the text into every space, turning it into a list
  if split_org[0] ∈ T.keys then           ▷ Check if first name in org is in the table
      return T[split_org[0]]
  end if
  if split_org[-1] ∈ T.keys then          ▷ Check if last name in org is in the table
      return T[split_org[-1]]
  end if

  if raw_org ∈ T.keys then                ▷ Check if org without preprocessing is in the table
      return T[raw_org]
  end if

  if author_id ∈ PT.keys then             ▷ Check if we have already matched this author before
      return PT[author_id]
  end if

  return ∅
```

The above process can be seen in the *infer_country_from* function in <https://github.com/rafaeelaudibert/conferences_insights/blob/v11/graph_generation/generate_country_citation_graph.py>.

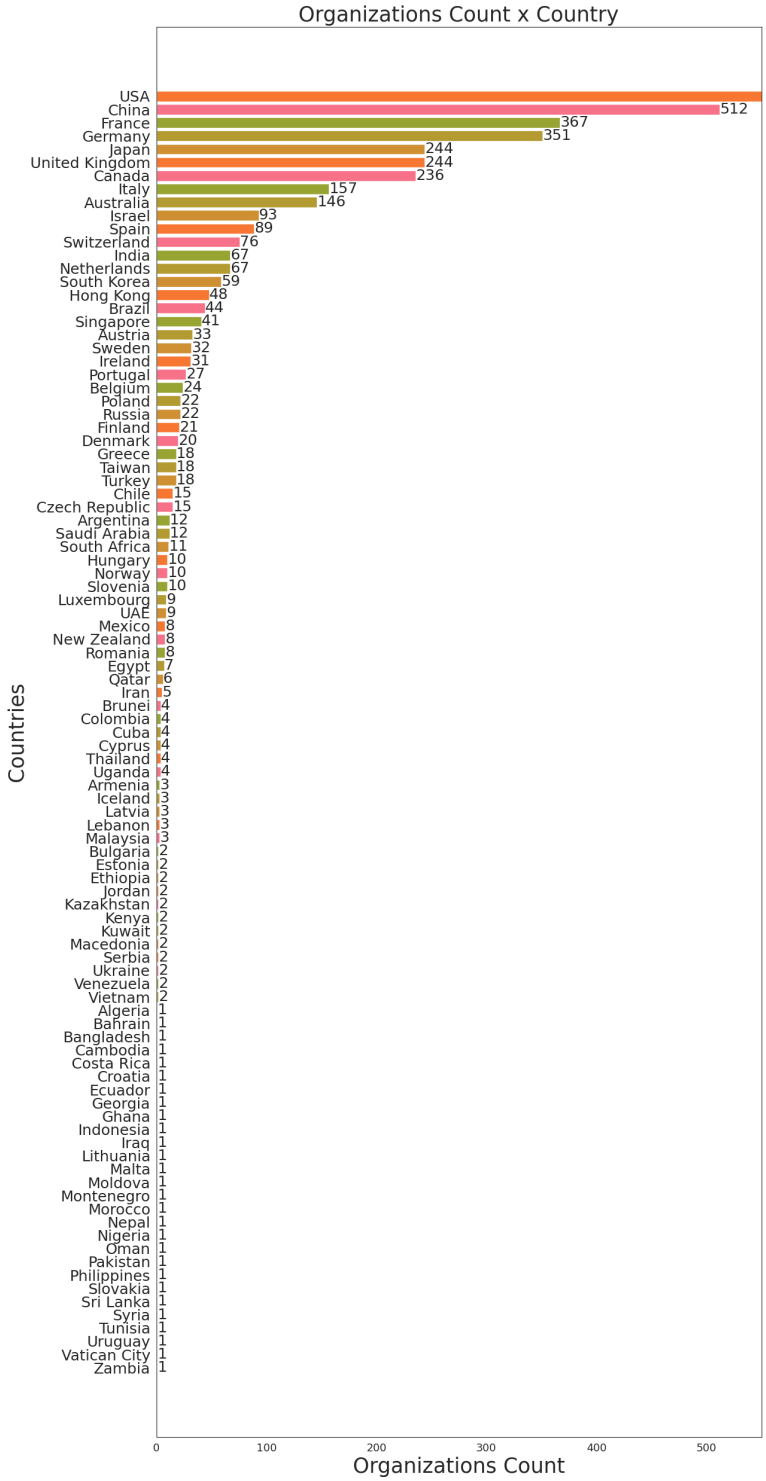
We do have another important step not fully explained in the steps above: how we created the "lookup table" to map from organizations to countries. We manually created it over the span of 2 months, through a manual iterative labor-intensive process: manually

looking at the organizations not matched using Algorithm 3 and mapping them to the countries they belong to using both our own knowledge and web searches to filter the options down. This mapping is available at <https://github.com/rafaeelaudibert/TCC/blob/v11/graph_generation/country_replacement.json>. The mapping for every organization that has ever been published in AAAI, IJCAI, and NeurIPS is complete, and the process to map this for the other conferences is still ongoing. We hope this mapping can be used in the future by other works to facilitate the inference of a country from an organization. Figure 3.11 shows how many organizations we mapped per country – the USA does not fit in the figure for scale purposes and has a value of 2163.

Additionally, there are a few authors whose “org” field is empty. For the first years of the area (1969-1979), we did not have many papers being published, so we manually looked at every single paper with an empty organization field and generated another lookup table available at <https://github.com/rafaeelaudibert/TCC/blob/v11/graph_generation/author_country_replacement.yml>. We then check this table first before attempting the above pipeline, because it is more reliable. This table was specially built in YML instead of JSON for better readability, and allows us to add comments in-between the entries.

It is notable, though, that this problem is worse in more recent years. Arnet’s data does not have organizations for most papers published from 2018 onward, so the problem is bigger in recent years. For example, in Figure 4.23 the "None" stacked part is bigger in recent years.

Figure 3.11 – Quantity of **mapped** different organizations per country that appeared in our data.



This does not reflect the true count of different organizations per country, because some of them could be easily identified by their country, and did not need any special treatment. The USA does not fit in the figure for scale purposes, and has a value of 2163.

Source: The Author

4 DATA ANALYSIS

This chapter presents the main analyses and insights performed on our datasets described in Chapter 3. We present initial statistics in Section 4.1, then analyse each graph (Sections 4.2 to 4.6). We then investigate the research impact of Turing Award winners in Section 4.7.

As already stated before, the full code for both the data generations and data analysis was made publicly available at <https://github.com/rafaeelaudibert/TCC/tree/v11>. The main code is in the branch *v11* because of the aforementioned problems with Arnet's V13.

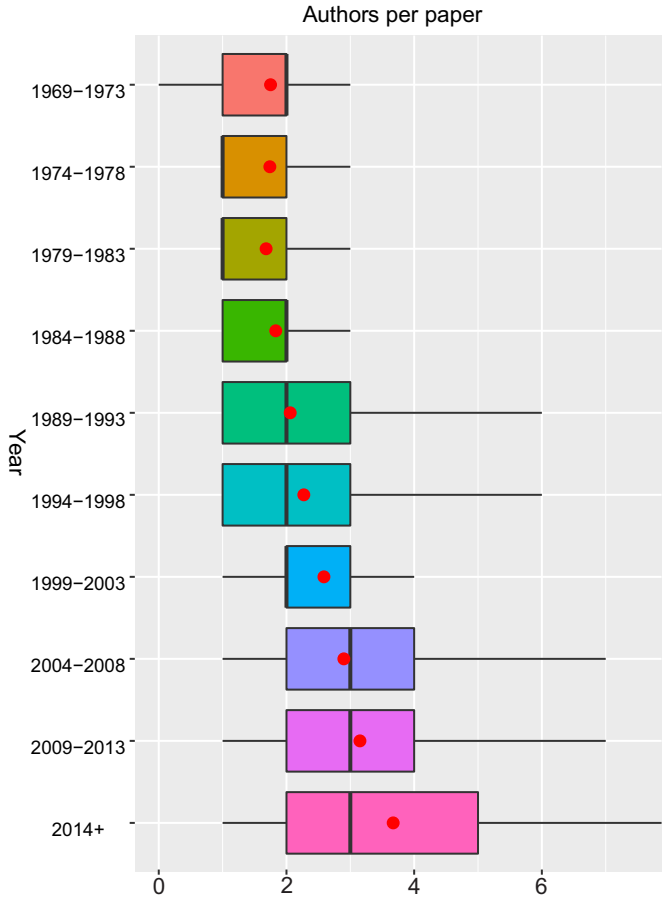
4.1 Raw Data

Although the bulk of this work is intended to revolve around the graph datasets and their centralities built to support our claims, the raw data itself is also able to provide us with great introductory information to the following sections.

Figure 4.1 shows a boxplot with a rising trend in the number of authors per paper over the years. In this boxplot graph the red dot represents the average number of authors per paper, the black line represents the median, the box per se represents the 95% percentile, while the black lines represent the 99% percentile – even showing a failure in the dataset with some papers with 0 authors in the late 1960s. The figure shows how the trend of several authors in a single paper, like (BROWN et al., 2020), (JUMPER et al., 2021), and (SILVER et al., 2016), is recent and rare with not more than 1% of the papers having 7 or more authors since 2004. It is noticeable how the average value jumps to almost 4 in the years past 2014.

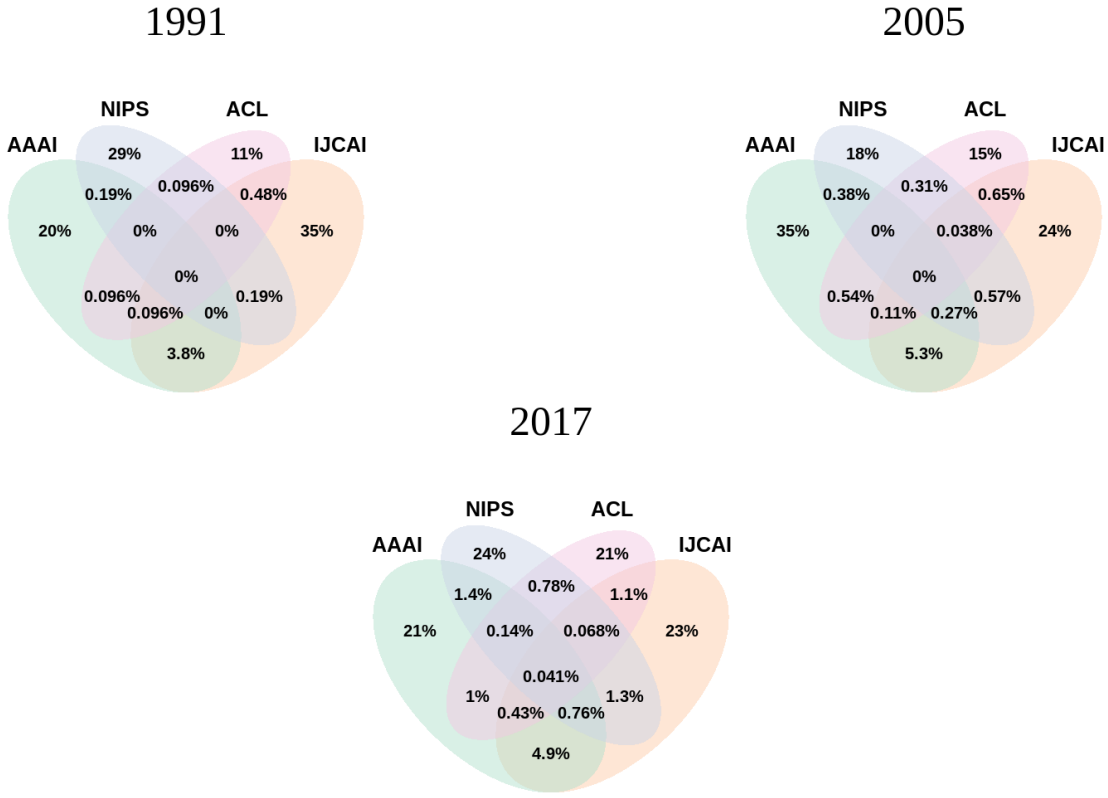
We also intersected the authors who published in the same year in different venues. Some interesting trends arose, such as: AAAI and IJCAI have the biggest overlap in authors than any combination of them with NIPS and ACL (Figure 4.2); CVPR has congregated more authors than NIPS and IJCAI since the beginning of the 2000s and its biggest authors overlap is always with NIPS (Figure 4.3); SIGIR had almost no overlap with these three conferences during the 90s and still has very little overlap nowadays, despite an increase in its intersection with AAAI (Figure 4.4).

Figure 4.1 – Boxplot of the number of authors for each single paper per year



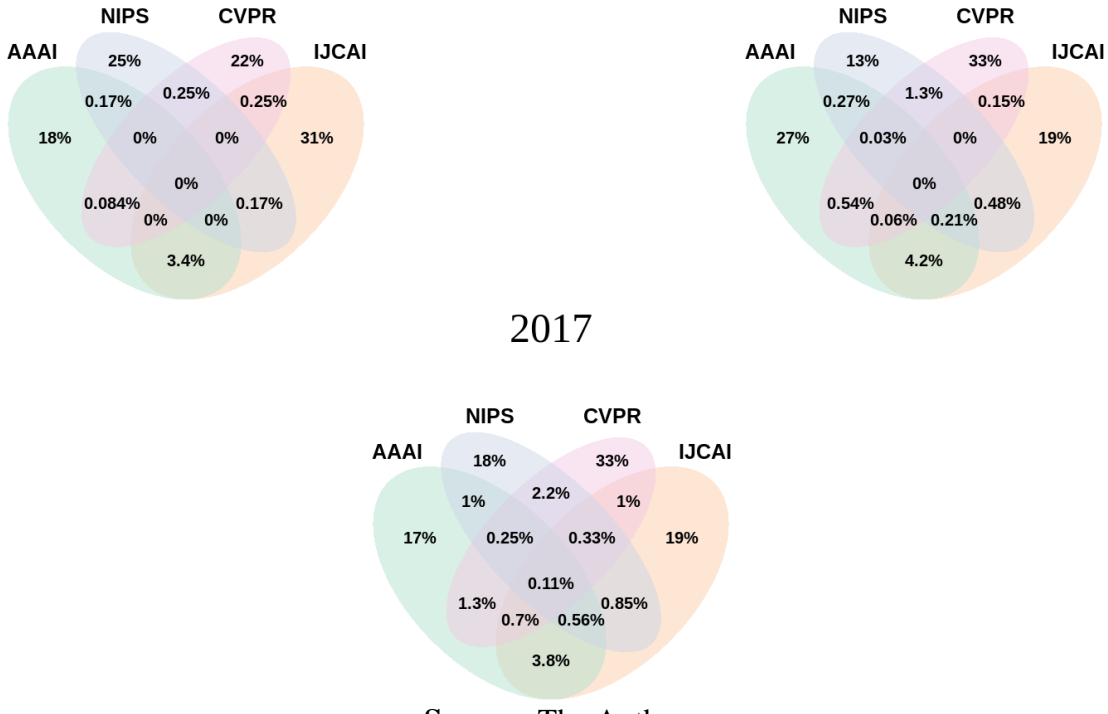
Source: The Author

Figure 4.2 – Percentage of overlapping authors in AAAI, NIPS, ACL, and IJCAI.



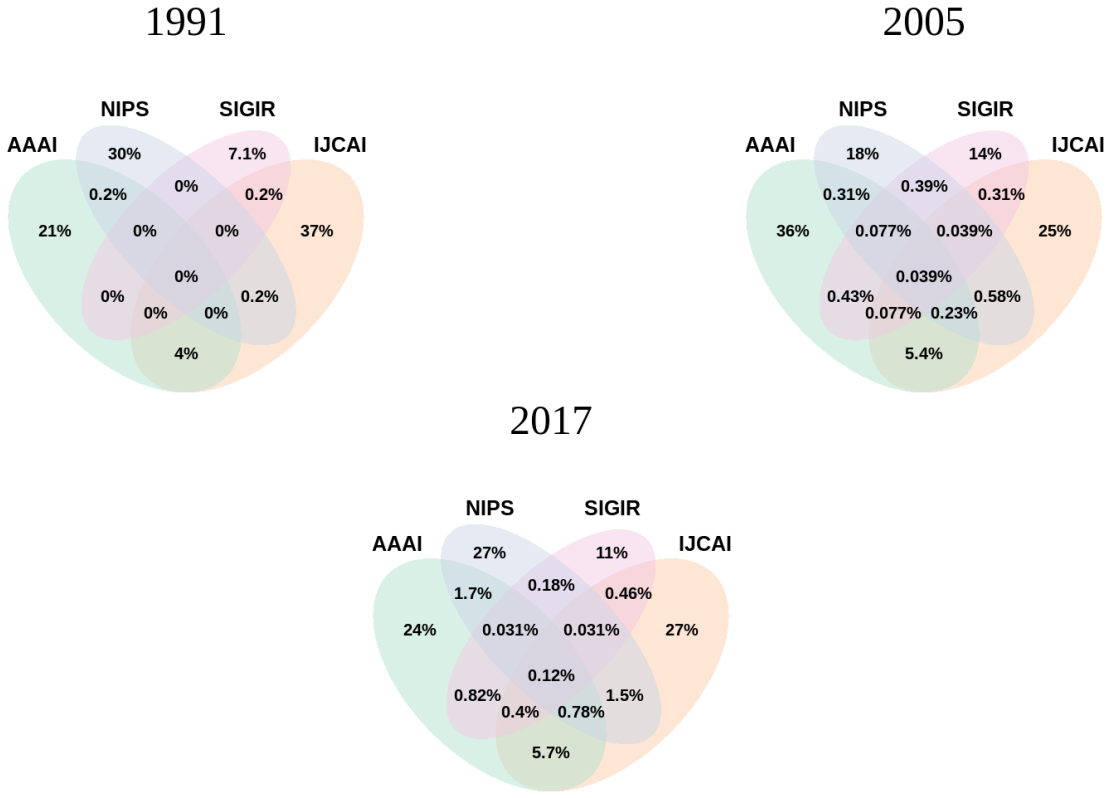
Source: The Author

Figure 4.3 – Percentage of overlapping authors in AAAI, NIPS, CVPR, and IJCAI.



Source: The Author

Figure 4.4 – Percentage of overlapping authors in AAAI, NIPS, SIGIR, and IJCAI.



Source: The Author

4.2 Author Citation Graph

4.2.1 Ranking over time

We have calculated an authors' ranking regarding the aforementioned centralities from 1969 until 2019 using the accumulated citation data – AC graph.

Figures 4.7, 4.8, and 4.9 show the evolution of PageRank, Betweenness and In-Degree centralities, respectively, in our Author Citation Graph. In these figures, a line represents a single author and its ranking evolution over time in some predefined years (chosen to be 1969, 1977, 1985, 1993, 2001, 2009, and 2014). The only authors shown are those who, at any point in one of these years, reached the top 10 in that specific rank. Authors who hadn't published yet in one of these years and, therefore, did not have any rank yet, show as *N/A*.

Although chaotics, these graphs do have some interesting insights. Figure 4.7 is an interesting starting point because it is considerably stable, at least at the top of it. Harry Pople was the top 1 author in this ranking at least from 1977 until 2001, the longest period one will hold this position in any of our analyses. His main work is focused

on Artificial Intelligence to Medicine (DHAR; POPLE, 1987) therefore very central in-between different areas. Also in the PageRank graph, one might see that the rises tend to be meteoric with Andrew Ng going from position 974 in his debut year of 2001 to 16th 8 years later, and then 2nd after 5 more years. The same can be said for most of the dynamics present in this graph.

The aforementioned insights also hold for Figure 4.8 where Betweenness is analyzed. This graph is a lot less stable than PageRank's, as betweenness is easier to evolve when new areas in Machine Learning happen, therefore changing the flow of information in the graph, while PageRank will be more stable because important people at one time will continue to be as important as they were forever, only going down in rank if someone even more influential appears. One can see this dynamic, for example, by looking at the last position in both charts: Larry Tesler – the one but last in the PageRank chart because the last position is an outlier – is 4267th in the PageRank, while the last position in the Betweenness chart is 31159th, showing how low one might drop in the Betweenness ranking even though they once were in the top 10 most influential scientists.

The Indegree chart shows a basic and raw datapoint: which author is the most cited, which should reward older authors with seminal papers. The first place in this ranking belongs to Andrew Zisserman, author of papers such as (SIMONYAN; ZISSERMAN, 2014) and (HARTLEY; ZISSERMAN, 2003), having close to 300,000 citations over his whole life – more than 100,000 of those only for the 2 cited papers. The second position is Andrew Ng with just over a third of the number of citations that Zisserman has.

Considering all these charts together, it is interesting to see how Andrew Ng is the most influential overall author in the AI area when we analyze it from a citation perspective. He is the author of papers such as (BLEI; NG; JORDAN, 2003), and (NG; JORDAN; WEISS, 2001), having an h-index of 134, i.e. 134 papers with at least 134 citation (See H-Index on Section 2.4.1.5 to understand how this metric is computed), the 1403th biggest h-index in Google Scholar[1]. He appears with the biggest betweenness value, and second-biggest indegree and PageRank ranking.

Takeo Kanade, the first position in the page rank ranking, is only 14th and 16th when looking at betweenness and indegree, respectively – although it is worthy of note that in 2001 he was first in in-degree and betweenness while third in PageRank. This is the best result, on average, that can be found in our results. Similarly, Andrew Zisserman, the first position in the in-degree ranking, is sixth when looking at betweenness, and 8th on

[1] <https://www.webometrics.info/en/hlargerthan100>

the PageRank ranking.

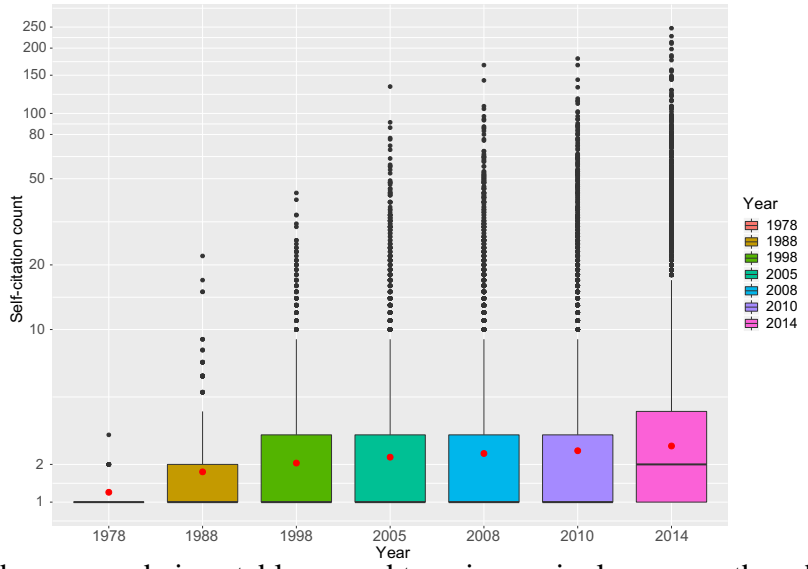
The ranking for the other computed centralities can be seen in Appendix C.

4.2.2 Self-citations

Authors might build up in their previous work, which would introduce self-edges in our graph representing self-citations. Figure 4.5 shows a boxplot of the evolution of self-citations count per year. Despite the average beginning stable at around two, increasingly more authors have been increasing their number of self-citations over the years.

This figure, however, does not represent the full truth because there are more papers recently. Figure 4.6 shows a better view of the same data, clearly showing the average number increasing. The data has its faults because if an author can publish more than one paper per year then it will help to bring the average up by not being divided twice, but this can be said for every single year, so the increasing rate of self-citations would still exist.

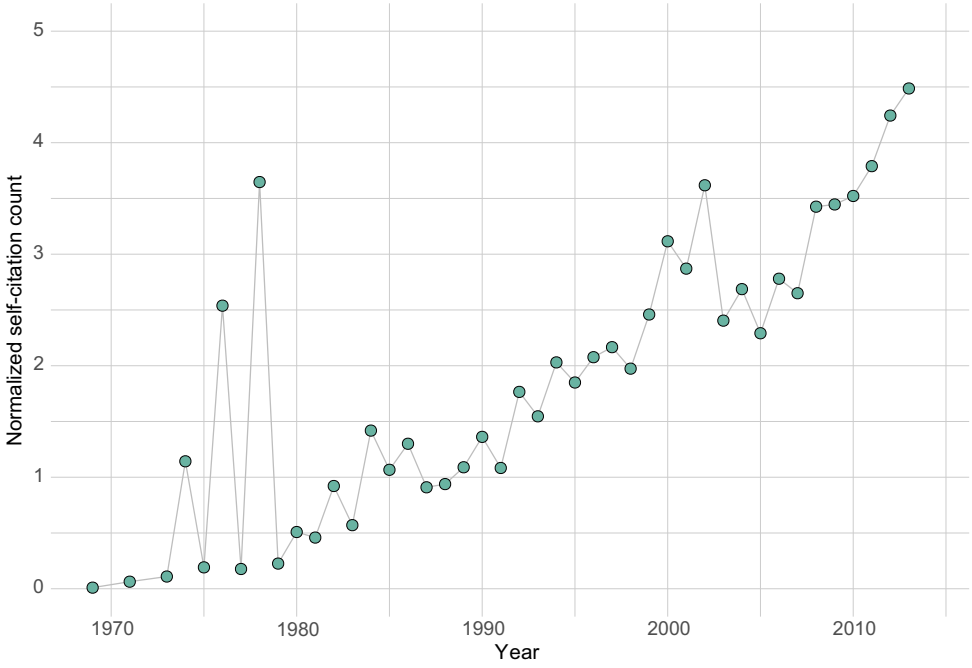
Figure 4.5 – Boxplot of self-citation count per year



Despite the average being stable around two, increasingly more authors have been increasing their number of self-citations over the years

Source: The Author

Figure 4.6 – Normalized self-citation count per year



Number of self-citations divided by number of authors who published during that year
Source: The Author

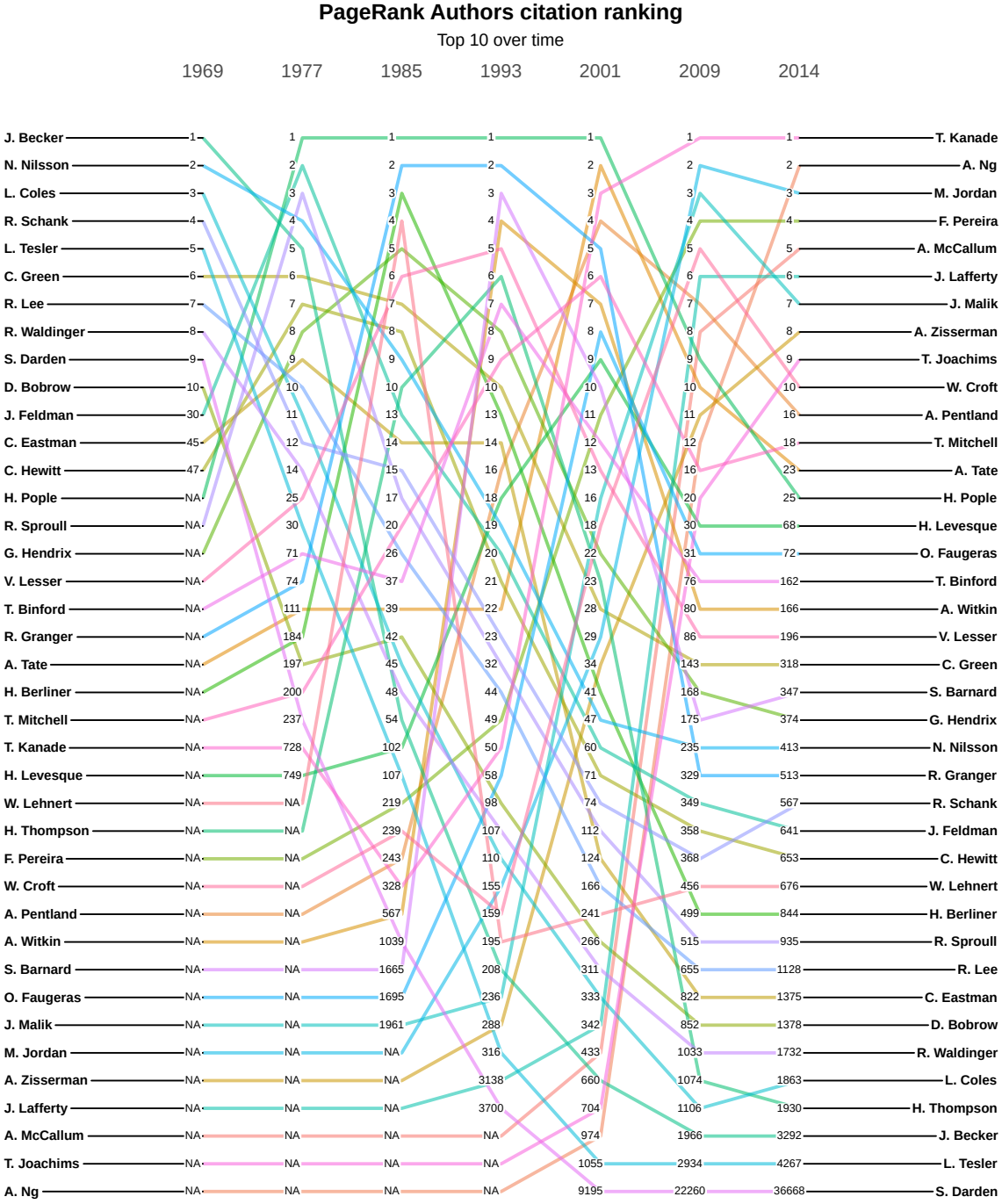
4.3 Author Collaboration Graph

4.3.1 Ranking over time

We have calculated an authors' ranking regarding the six aforementioned centralities from 1969 until 2019 using the accumulated collaboration data – *ACo* graph.

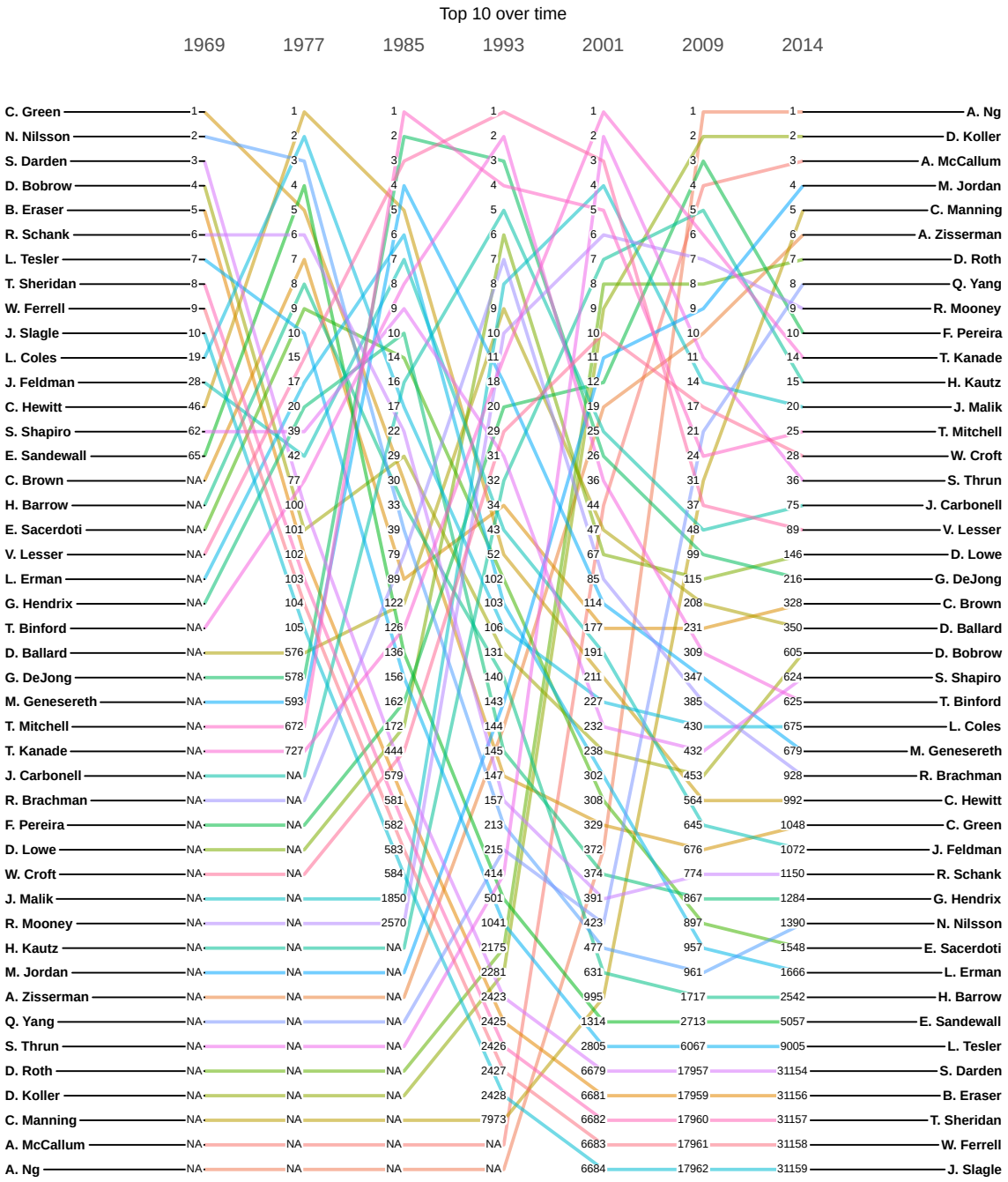
Figures 4.10 and 4.11 demonstrate how the PageRank and Betweenness rankings, respectively, evolved over time. In these figures, we chose to plot the top 10 authors each year, in an 8-year interval. Considering this gap, it is interesting to observe that only in 2009 it is possible to see all authors who appeared in the top 10 ranking during all the selected years. Also, most of the authors entered the ranking during the 80s and the 90s, regardless of the centrality. The remaining rankings (centralities) can be seen in the Appendix D.

Figure 4.7 – Author citation ranking over time according to PageRank centrality.



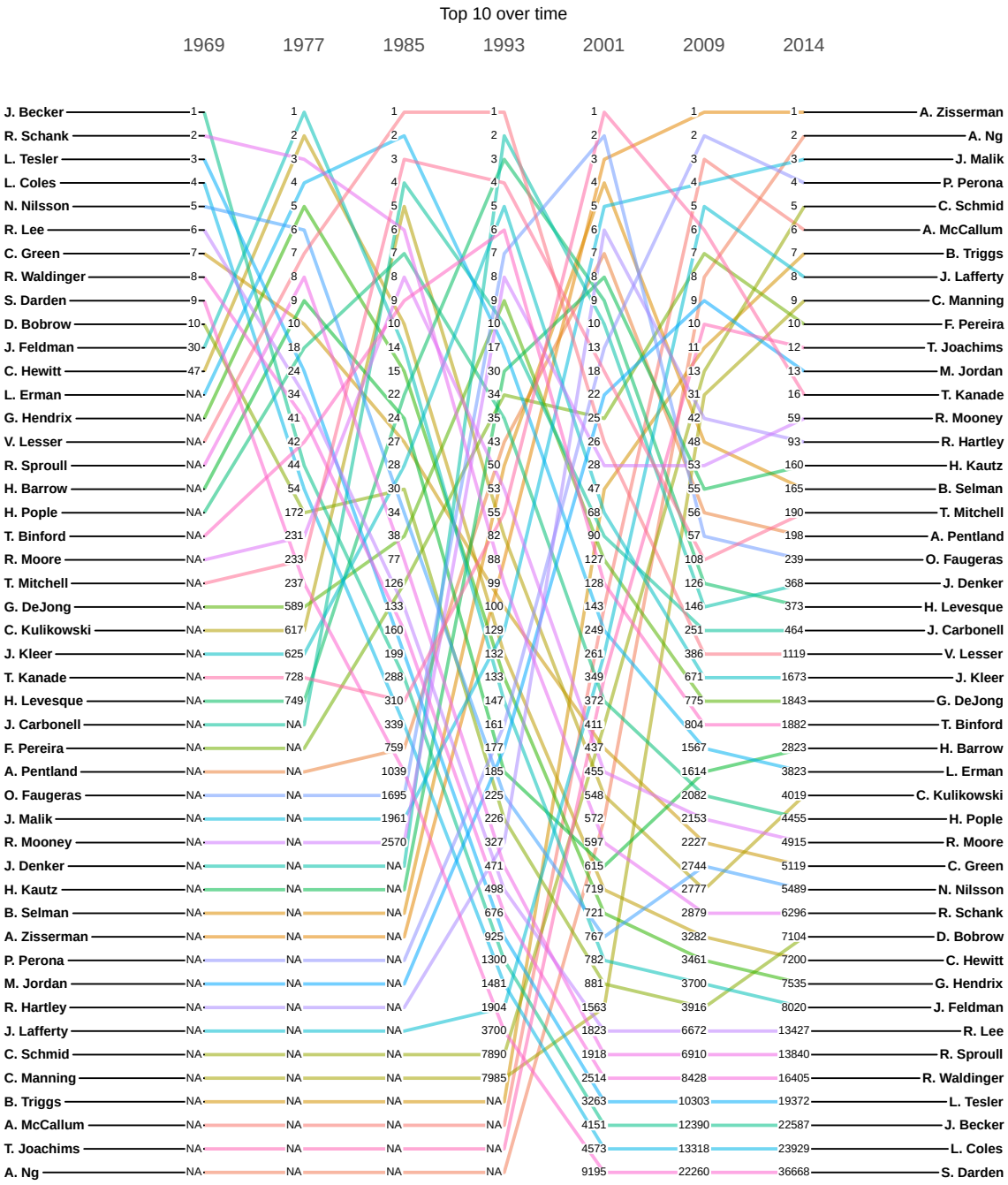
N/A stands for authors who had not published in the selected venues until that year.
Source: The Author

Figure 4.8 – Author citation ranking over time according to Betweenness centrality



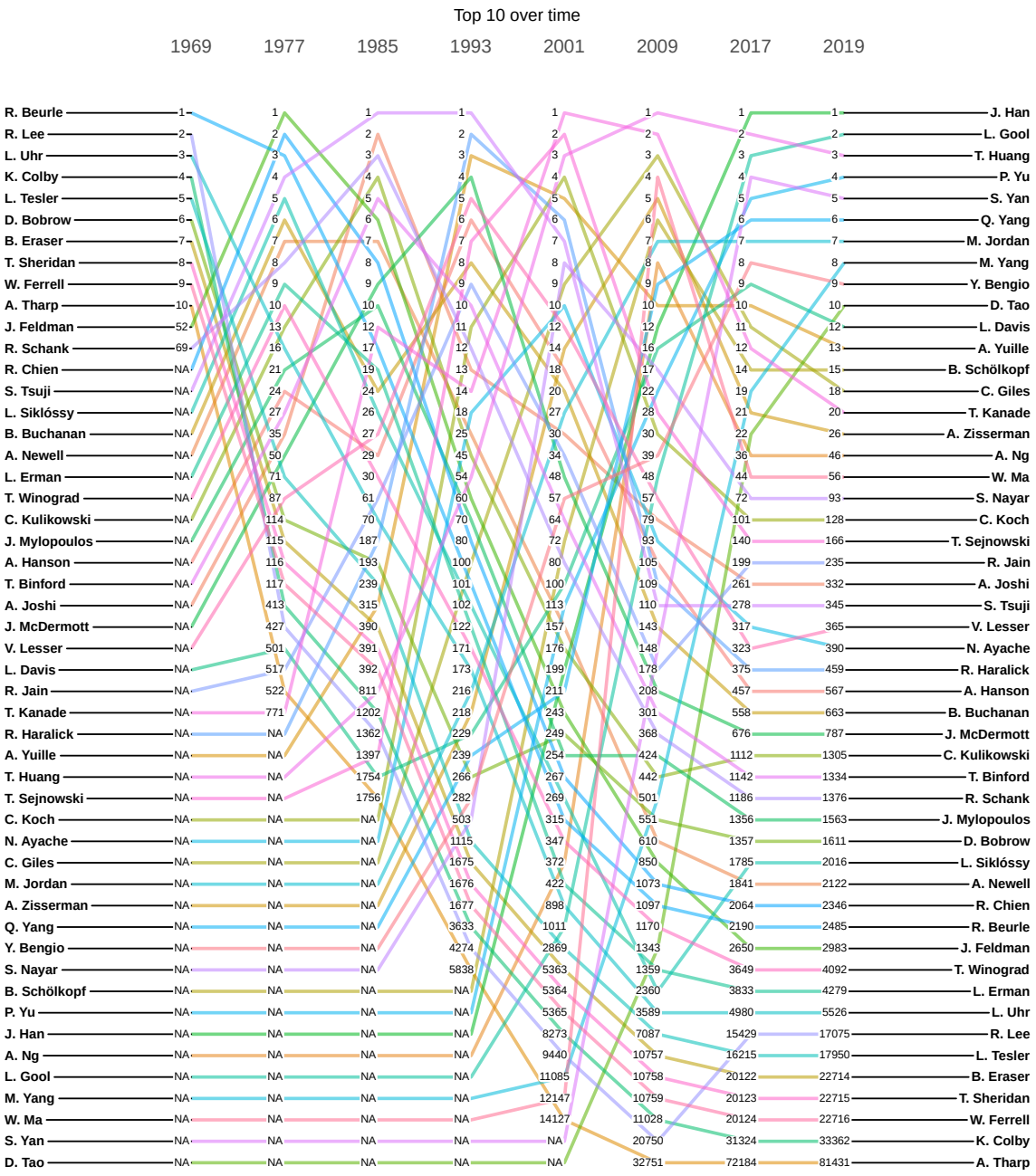
N/A stands for authors who had not published in the selected venues until that year.
Source: The Author

Figure 4.9 – Author citation ranking over time according to In-degree centrality



N/A stands for authors who had not published in the selected venues until that year.
Source: The Author

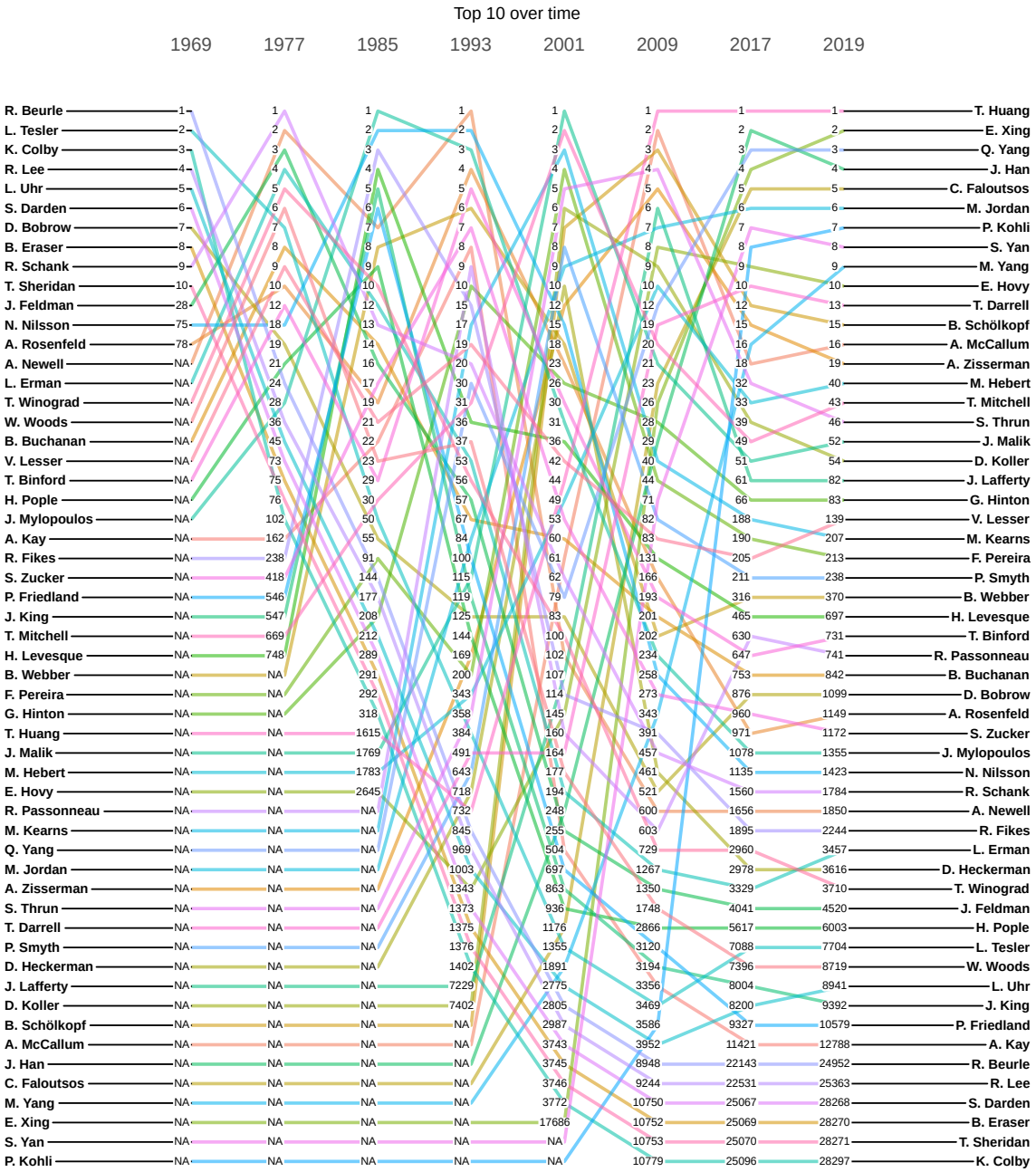
Figure 4.10 – Authors collaboration ranking over time according to PageRank centrality.



N/A stands for authors who had not published in the selected venues until that year.

Source: The Author

Figure 4.11 – Authors collaboration ranking over time according to betweenness centrality



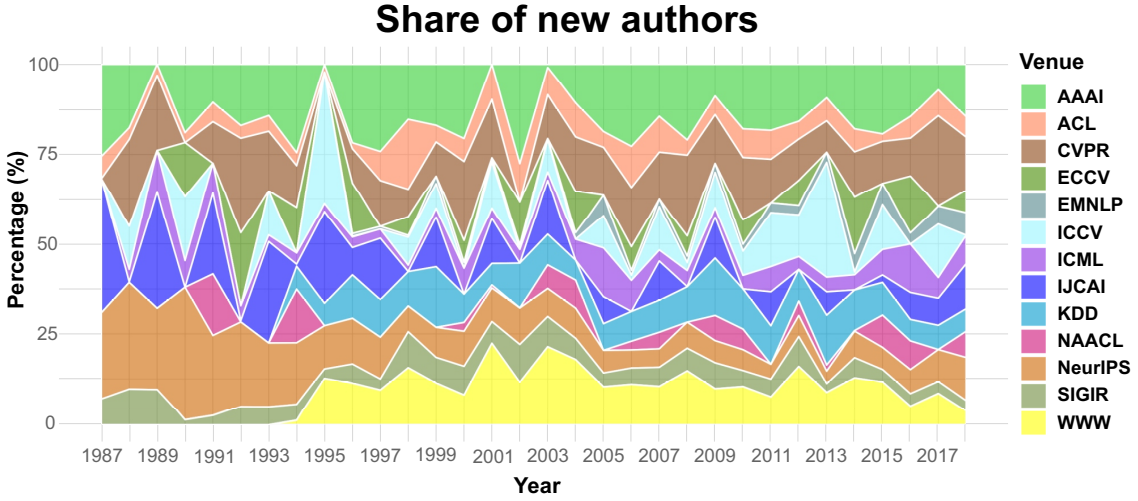
N/A stands for authors who had not published in the selected venues until that year.

Source: The Author

4.3.2 Entering the realm of AI

Every year several researchers publish their first papers in AI-related venues such as the ones we are analyzing throughout this work. Figure 4.12 shows the yearly share of new authors per conference. The stacked area contains spikes due to the fact that several conferences did not occur yearly. NIPS conference (currently NeurIPS) was the main responsible for attracting new authors until the mid-90s together with IJCAI. Since then the share became more and more split into conferences of different areas: highlights to CVPR and ICCV in the last years, and to AAAI and WWW at the beginning of the 2000s.

Figure 4.12 – Share of yearly new authors per conference.



Source: The Author

Table 4.1 lists all the authors who collaborated with more than 200 new authors since 1969. Several of them appeared in the authors' collaboration ranking (especially regarding Betweenness, Closeness, and PageRank centralities). The regular behavior, however, is better described by the average and standard deviation statistics: the average number of new authors that an author collaborated with is around 4 with an std. dev. of 12. It is also a fact that these numbers are highly affected by the career age of a researcher. We estimated this age with the author's first year of publication inside our graph and we used this age to normalize the amount of collaboration with new authors, achieving a normalized average number of new authors per author of 0.3 (avg. career time: 11) with std. dev of 0.94 (std. dev. career time: 9.2), which essentially means that a researcher usually brings a new author to these AI venues after 3 years of his entry into the field.

Table 4.1 – The 23 authors who collaborated with more than 200 new authors since the year 1969

Author	Count
Lei Zhang	488
Luc Van Gool	352
Ming-Hsuan Yang	350
Thomas S. Huang	322
Andrew Y. Ng	298
Jiawei Han	294
Dacheng Tao	282
Yang Liu	280
Philip H. S. Torr	280
Yoshua Bengio	248
Wei Wang	238
Milind Tambe	236
Yang Li	232
Aleš Leonardis	224
Liang Lin	222
Qingming Huang	222
Shuicheng Yan	222
Christos Faloutsos	222
Jiri Matas	214
Michael Felsberg	212
Horst Bischof	212
Philip S. Yu	208
Richard Bowden	206

Source: The Author

4.4 Paper Citation Graph

4.4.1 Ranking over time

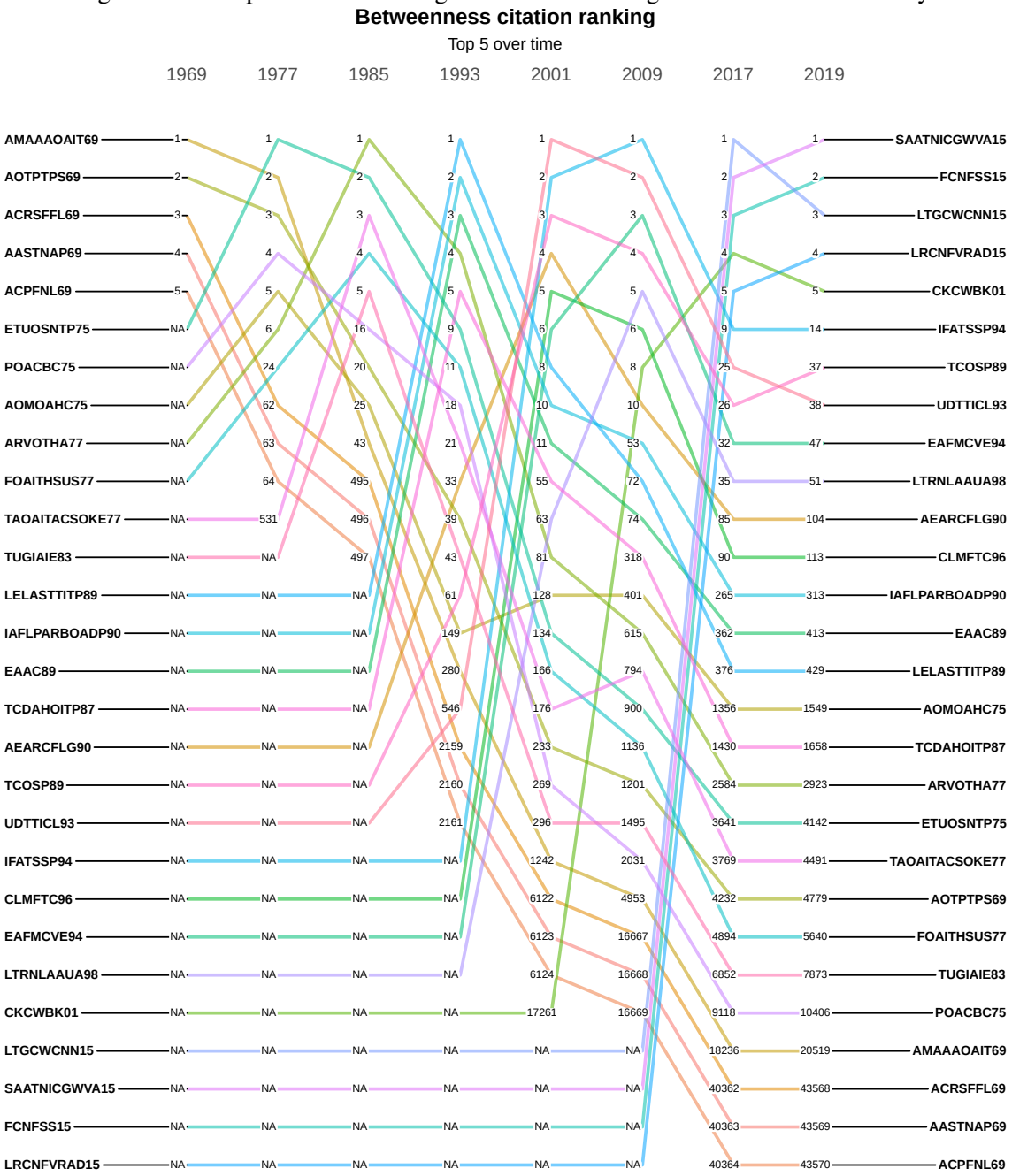
In every centrality measure done for this graph, whenever we plot it we mapped the name of the papers to those in Table E.1. This format is not ideal for readability, but it was the best method found to show this data in its full form.

When it comes to citation networks, the betweenness centrality can be seen as a measure of how a node (paper) is able to connect different research areas, or how it acts to foster interdisciplinarity (LEYDESDORFF, 2007). In this sense, Figure 4.13 shows how the ranking of most important papers (according to betweenness centrality) evolved. It is possible to see that the ranking itself is very volatile as no paper can remain in the top 5 for more than 2 times (inside our gap of 8 years), nevertheless the paper "Constrained K-means Clustering with Background Knowledge" (WAGSTAFF et al., 2001) (CKCWBK01 in the figure) has been in the top 10 at least since 2009. Also, all papers in the top 5 of 2017 and 2019 were published after the year 2000, which could indicate that, despite not being seminal papers, these recent researches are being more helpful in different areas.

4.14 shows the same graph data, but ranked by their in-degree centrality, which simply measures how many citations a paper has received until a given year (inside our graph). The latest top 5 is composed of 3 papers related to computer vision and 2 to natural language processing. A similar pattern can be found until 1993, but back in 1985 and before most of the ranking is composed of papers that tackled reasoning, problem-solving and symbolic learning, such as "Reasoning about knowledge and action" (MOORE, 1977) (RAKAA77 in the figure), "A multi-level organization for problem-solving using many, diverse, cooperating sources of knowledge" (ERMAN; LESSER, 1975) (AMOFPSUMD-CSOK75 in the figure) and "The art of artificial intelligence: themes and case studies of knowledge engineering" (FEIGENBAUM, 1977) (TAOAITACSOKE77 in the figure).

A very stable behavior can be seen in the PageRank ranking (Figure 4.15): most papers remained in the top 5 for 2 gaps (usually 8 years) and many of them for 3 gaps (16 years in the middle, 10 in the end). The paper "Towards automatic visual obstacle avoidance" (MORAVEC, 1977) (TAVOA77 in the figure) is in the top 5 at least since 1993 and it is leading the ranking since 2001. The second one, "Feature extraction from faces using deformable templates" (YUILLE; COHEN; HALLINAN, 1989) (FEFFUDT89 in the figure), is also a somewhat old paper related to computer vision.

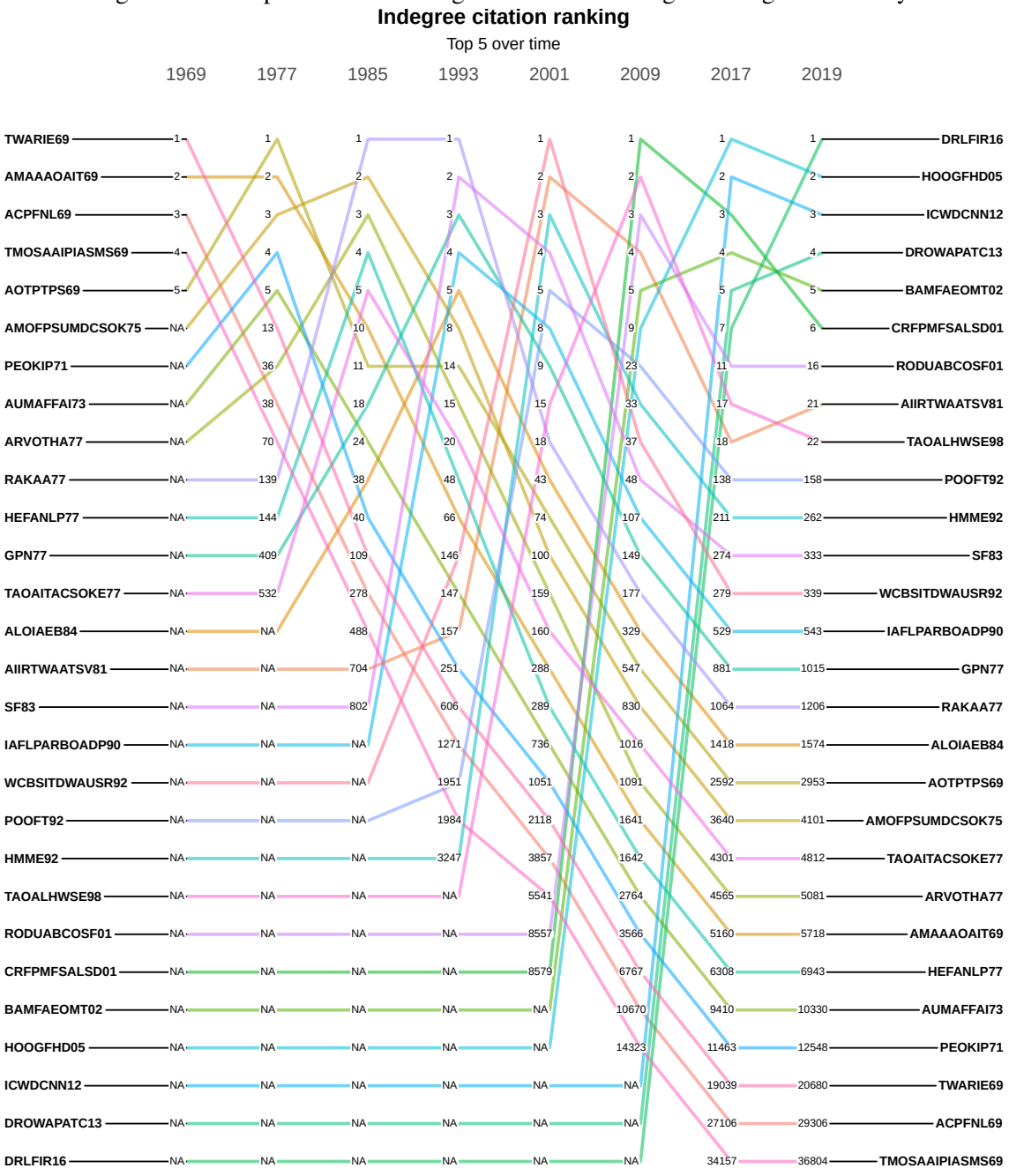
Figure 4.13 – Paper citation ranking over time according to Betweenness centrality.



N/A stands for papers who had not been published in the selected venues until that year. Please refer Table E.1 in the Appendix E to see the details of each ranked paper.

Source: The Author

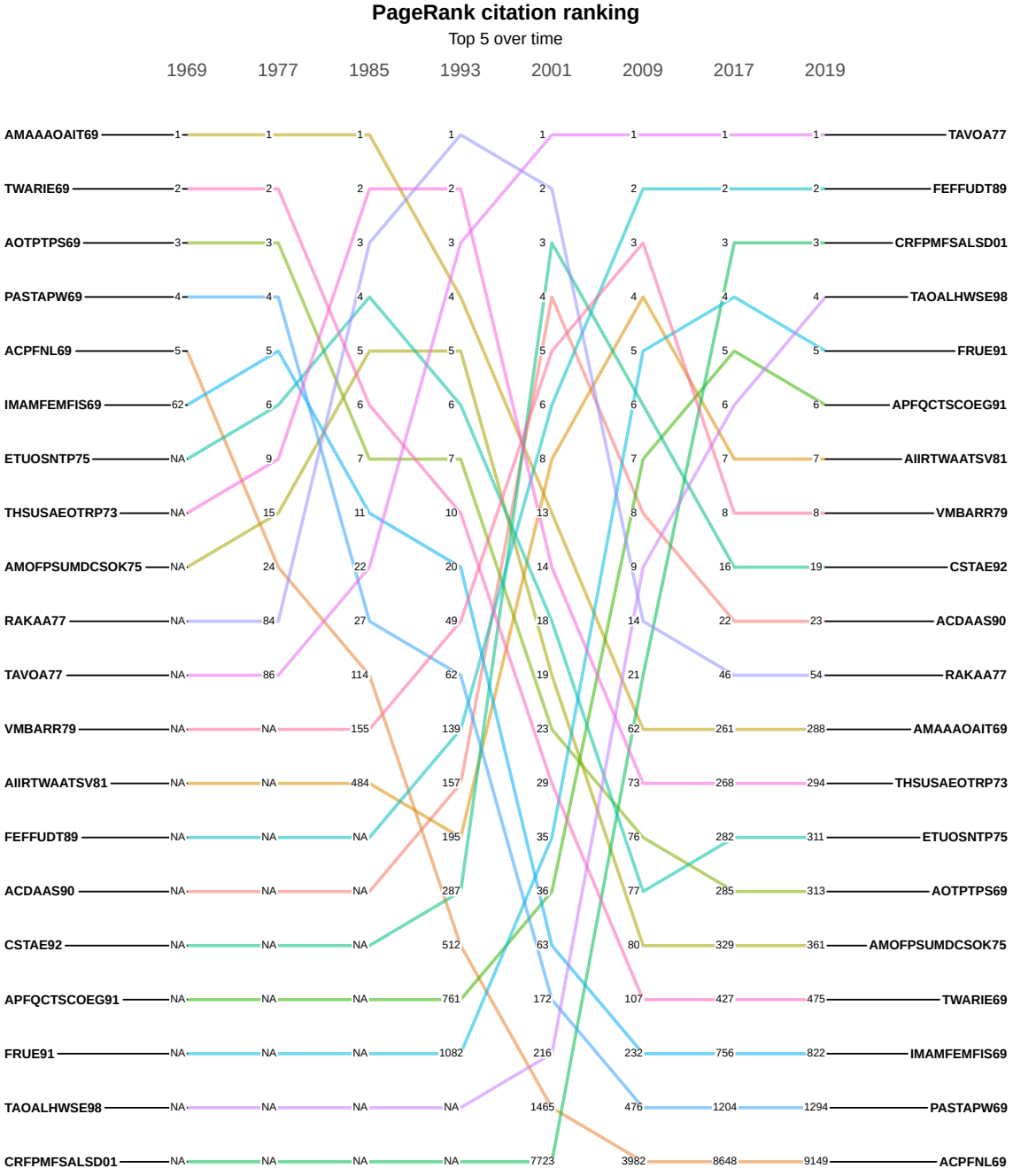
Figure 4.14 – Paper citation ranking over time according to In-degree centrality.



N/A stands for papers who had not been published in the selected venues until that year. Please refer Table E.1 in the Appendix E to see the details of each ranked paper.

Source: The Author

Figure 4.15 – Paper citation ranking over time according to PageRank centrality.



N/A stands for papers who had not been published in the selected venues until that year. Please refer Table E.1 in the Appendix E to see the details of each ranked paper.

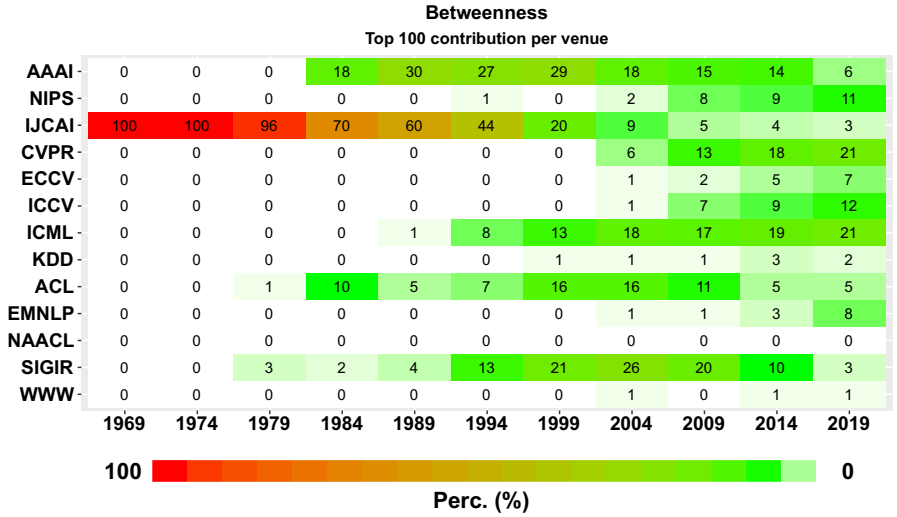
Similarly, one can see the stableness and invariability to change that PageRank offers by noticing that we only had 20 different papers in the top 10 in the selected years, while we have 28 for Betweenness and Indegree, a more befitting number when compared to the figures shown in the previous sections.

The remaining rankings (Closeness, Degree, and Out-degree) can be seen in the Appendix E.

4.4.2 Share of top 100 ranking per venue

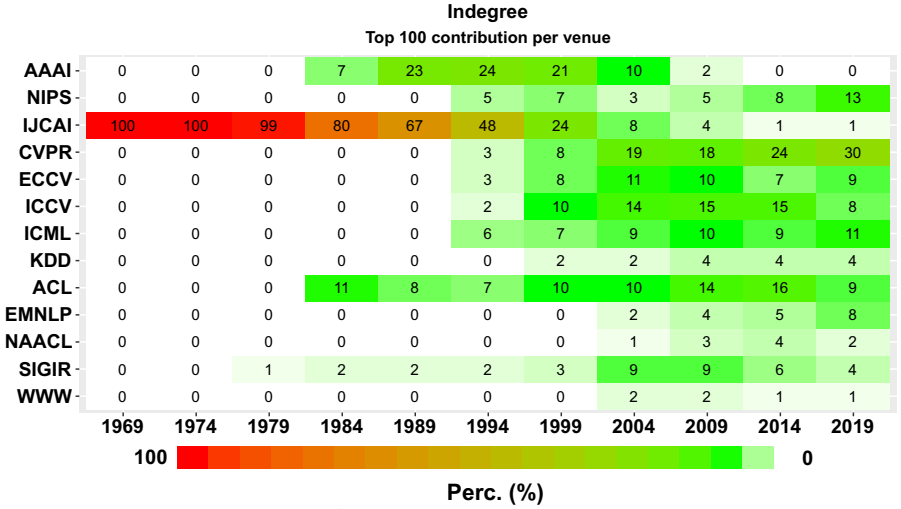
Figures 4.16 to 4.18 reinforce the trend seen in the top 5 ranking in the last section: despite the centrality, computer vision-related venues are progressively gaining importance regarding their published papers, especially the CVPR. However, there is a distinguished contribution by the ACL conference to the most important papers (according to PageRank) since 1984. These three heatmaps also show that the AAAI papers had their peak of importance during the late 1980s and the 1990s, but now they are losing their share of the ranking in the same fashion that IJCAI.

Figure 4.16 – Venue contribution per year (accumulated) in the top 100 most important papers, according to Betweenness.



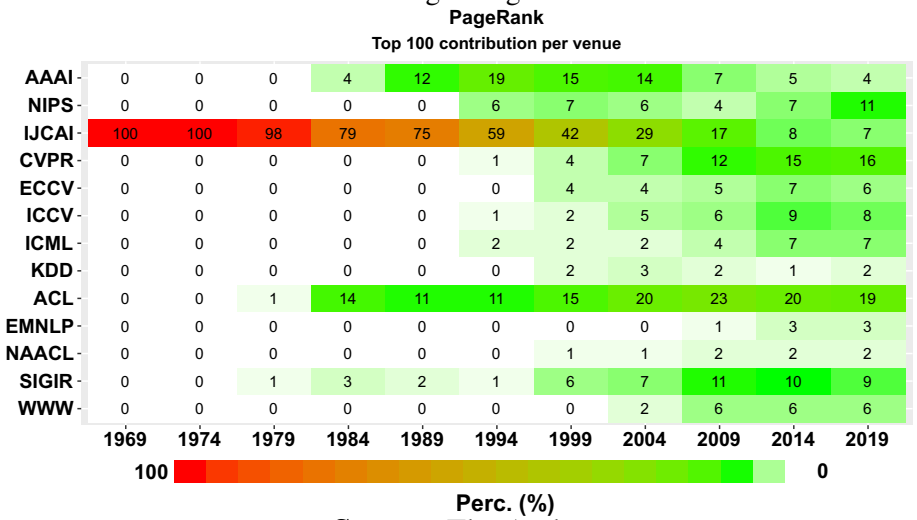
Source: The Author

Figure 4.17 – Venue contribution per year (accumulated) in the top 100 most important papers, according to In-Degree.



Source: The Author

Figure 4.18 – Venue contribution per year (accumulated) in the top 100 most important papers, according to PageRank.



Source: The Author

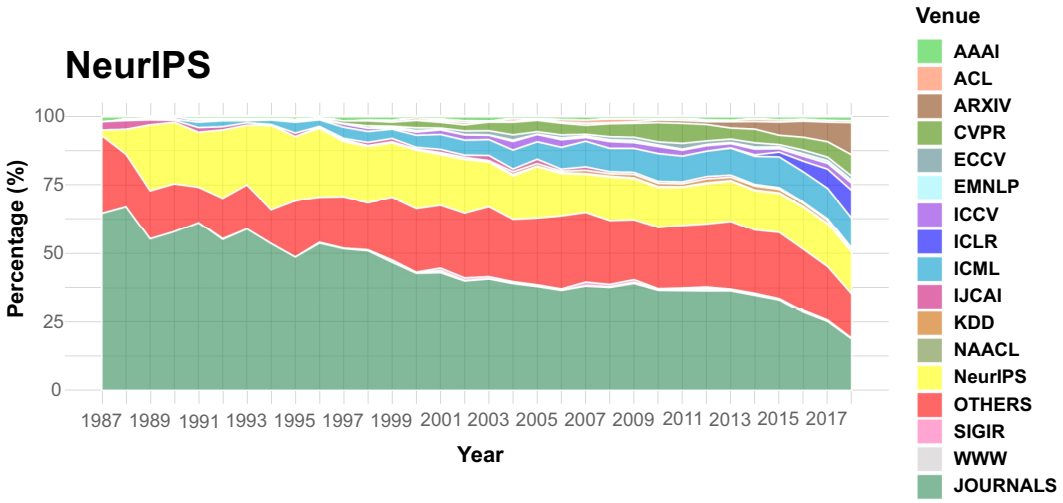
4.4.3 Share of citations per venue

We were also interested in how the citations of each venue have been evolving in the last few years. In this analysis, we were also able to distinguish citations to papers from arXiv, Journals, and the International Conference on Learning Representations (ICLR). For instance, back in the 1980s and early 1990s, around 50% of citations coming from NIPS papers were directed to papers from journals (see Figure 4.19), however, this share nowadays has been reduced to less than 25%. More than that, citations to ICLR papers and especially to arXiv papers have been increasing since the early 2010s.

A similar pattern occurs when we consider papers from AAAI and IJCAI, Fig-

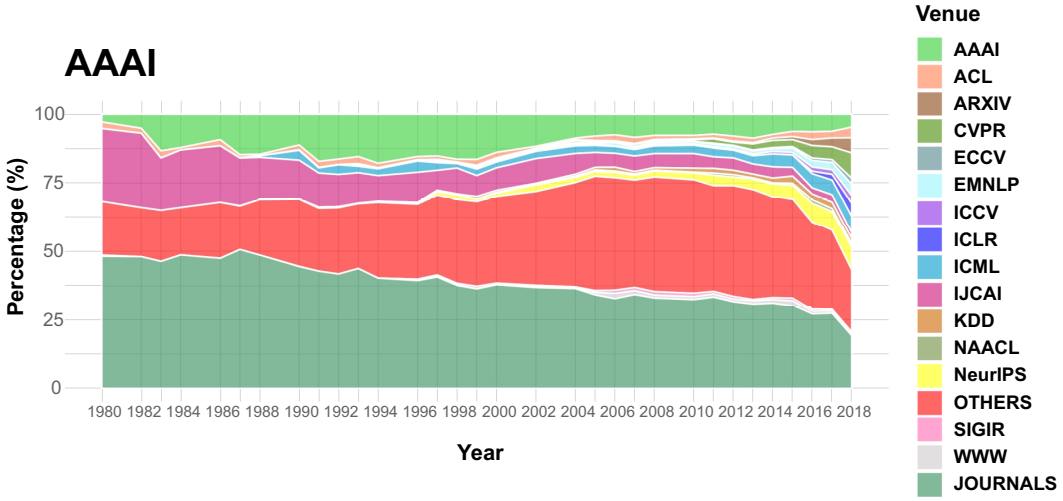
ures 4.20 and 4.21 respectively. However, in their case, there is a much more divided share between all the conferences: it is possible to distinguish some influence from KDD, WWW, ACL, and EMNLP together with the increasing, yet unobtrusive, the influence of arXiv and ICLR.

Figure 4.19 – Where the citations coming from NeurIPS papers are pointing to: share of each venue.



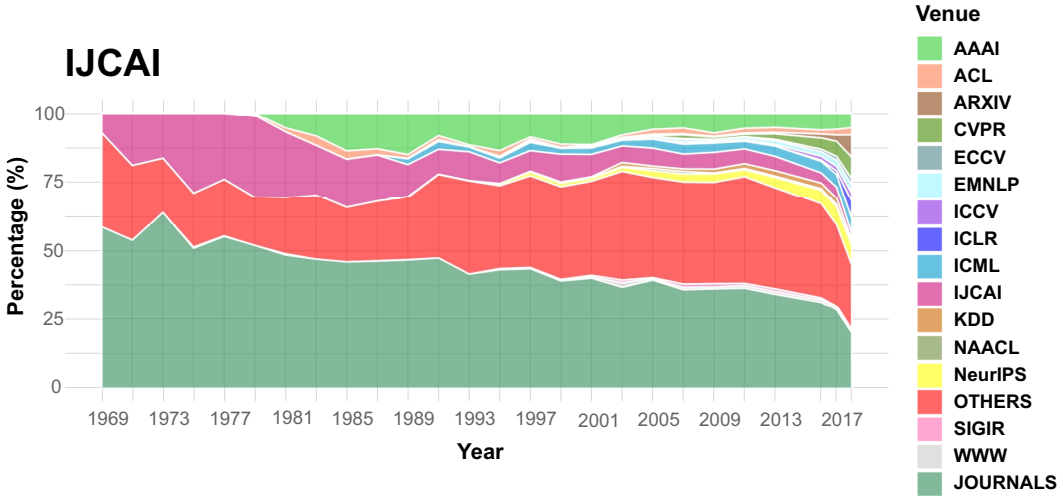
Source: The Author

Figure 4.20 – Where the citations coming from AAAI papers are pointing to: share of each venue.



Source: The Author

Figure 4.21 – Where the citations coming from IJCAI papers are pointing to: share of each venue.



Source: The Author

4.5 Author-Paper Citation Graph

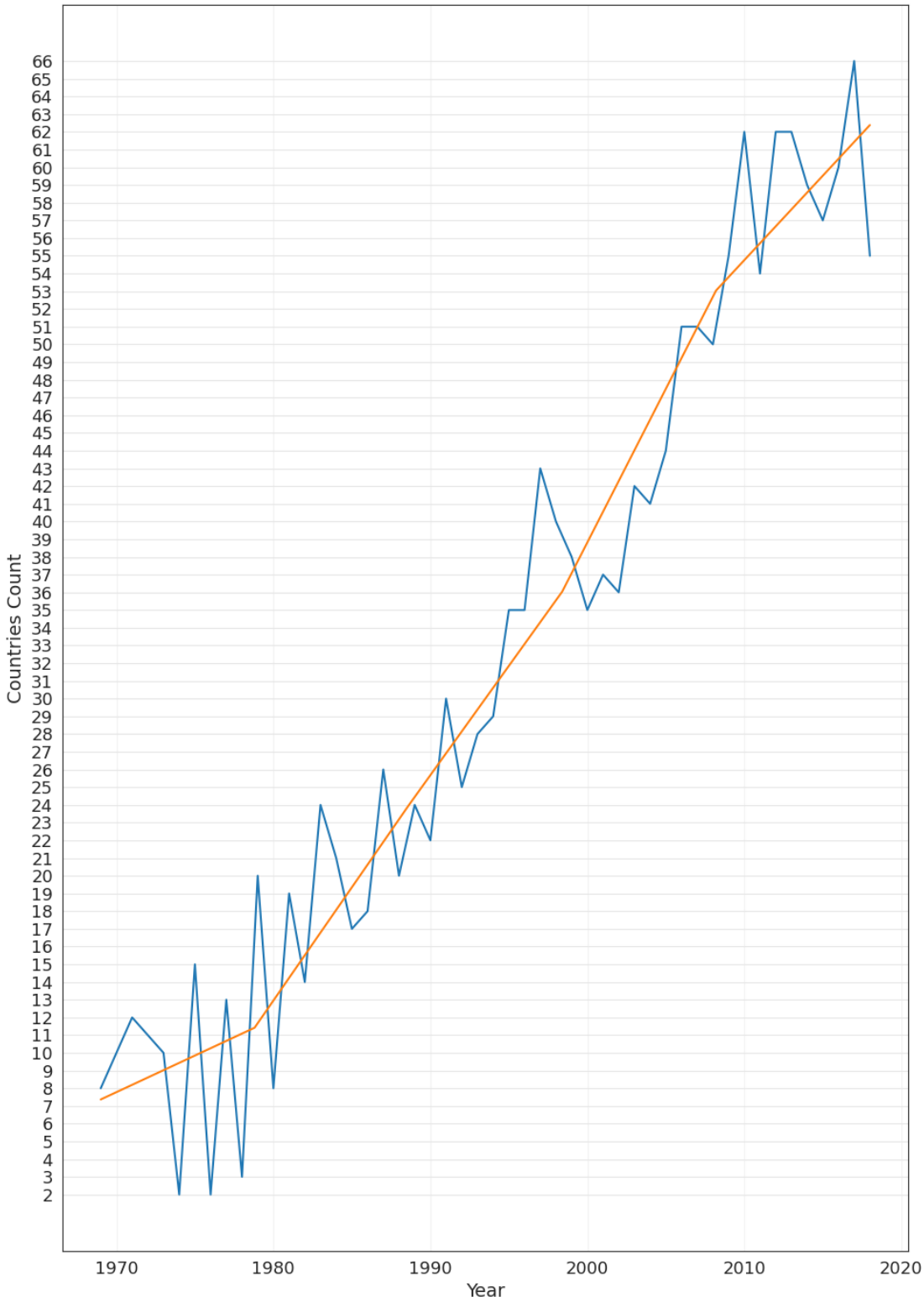
This graph was built to foster the research on recommender systems for papers – "given an author and his publication history, which are the most relevant papers he has not yet cited?"". In this sense, we have not developed any analysis on this graph and we expect to use it as a benchmark for future research on recommender systems: see Section 5.3.

4.6 Country Citation Graph

This graph, computed for every country, is interesting because it has a different cardinality compared to the other graphs: while the others have tens of thousands of nodes, this graph only has 93 nodes with 4776703 edges. Figure 4.22 shows the increase in the number of papers published at these conferences per year. The best-fitting line interpolates the data every 5 years. While the mid-1970s saw just a few countries participating in conferences (1974 and 1976 only had 2 countries: USA and United Kingdom, and USA and Canada, respectively) we have seen a huge increase in countries participating in conferences in the last years, with 66 different countries having published in the conferences of interest in 2017. As mentioned above, in total authors from 93 different countries already published at these conferences.

Figure 4.23 shows a stacked percentual chart of the 15 countries with the most published papers. This data clearly shows the dominance of the United States in Artificial

Figure 4.22 – Quantity of countries that published papers per year.



The interpolating line is the best-fitting linear interpolation with 5 points
Source: The Author

Intelligence research, with a slow increase in the number of papers published by authors in China. Similarly, Figure 4.24 shows the same data but the pink bar at the bottom represents papers that we could not detect their country of affiliation.

Through this data representation, one can clearly see the years when IJCAI happened. Given the fact that IJCAI is commonly held outside the United States, and only in odd-numbered years, we can see a jagged-line pattern in the United States share of papers, with a higher percentage in even-numbered years, and a lower percentage in odd-numbered years (when people from different countries have a higher chance of attending the conference, usually because of less strict visa requirements). For the same reason, after IJCAI started to be held annually (2013) the pattern disappears. Figure F.1 tries fixing this problem by creating a 2-year-wide sliding window and averaging the data before plotting it, creating a clearer view of the data.

An interesting outlier can be seen in 1979 when Japan has the highest share of published papers except for the USA. That happened exactly because IJCAI was held in Tokyo that year.

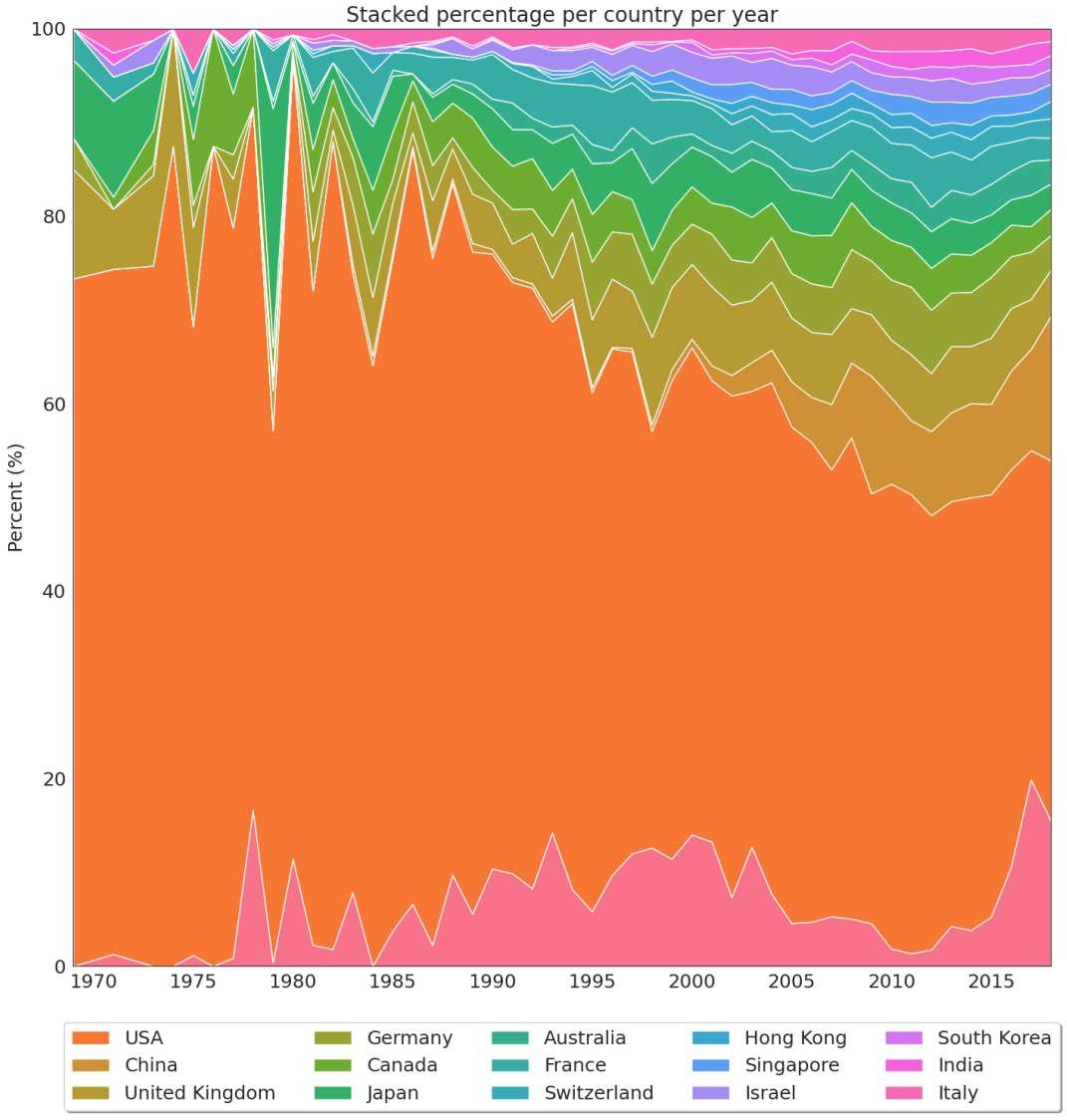
Figure 4.25 shows the same data but with numbers in absolute terms instead of showing it with a stacked percentual. With it, we can see how the rate of acceptance for countries that are not the USA has grown faster than it has for the USA (the curve is steeper at the top). With it, we can also see the striking increase in papers accepted to these conferences in the last years, as already shown in numbers before.

4.6.1 Analysis with more recent years

If we try using Arnet's v13 dataset to generate the above graphs, we can see in Figure 4.26 how much the data deteriorates. For 2019, we can only identify close to 5% of the paper country of affiliation, because for most of them the "organization" field is empty. Note as well that even for the previous year the data is not as clear as it is in Figure 4.24, for instance.

We didn't try applying any mapping similar to the one explained in Section 3.3.5.1 to this data because most of the non-identifiable organization fields are empty, as stated above.

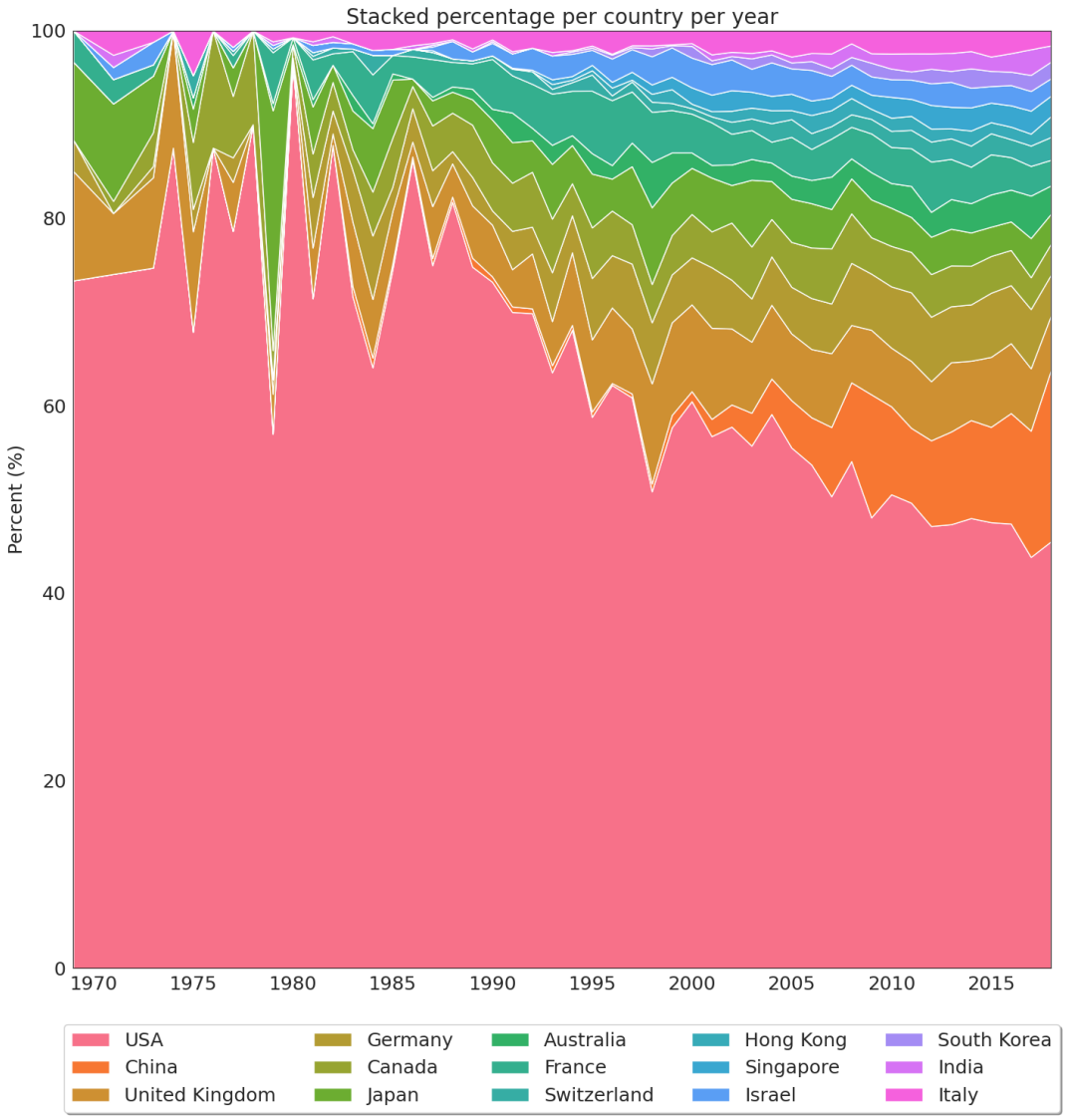
Figure 4.23 – Stacked percentage of papers published per country per year including non-mapped ones.



The pink bar at the bottom indicates papers we could not identify which country they are from.

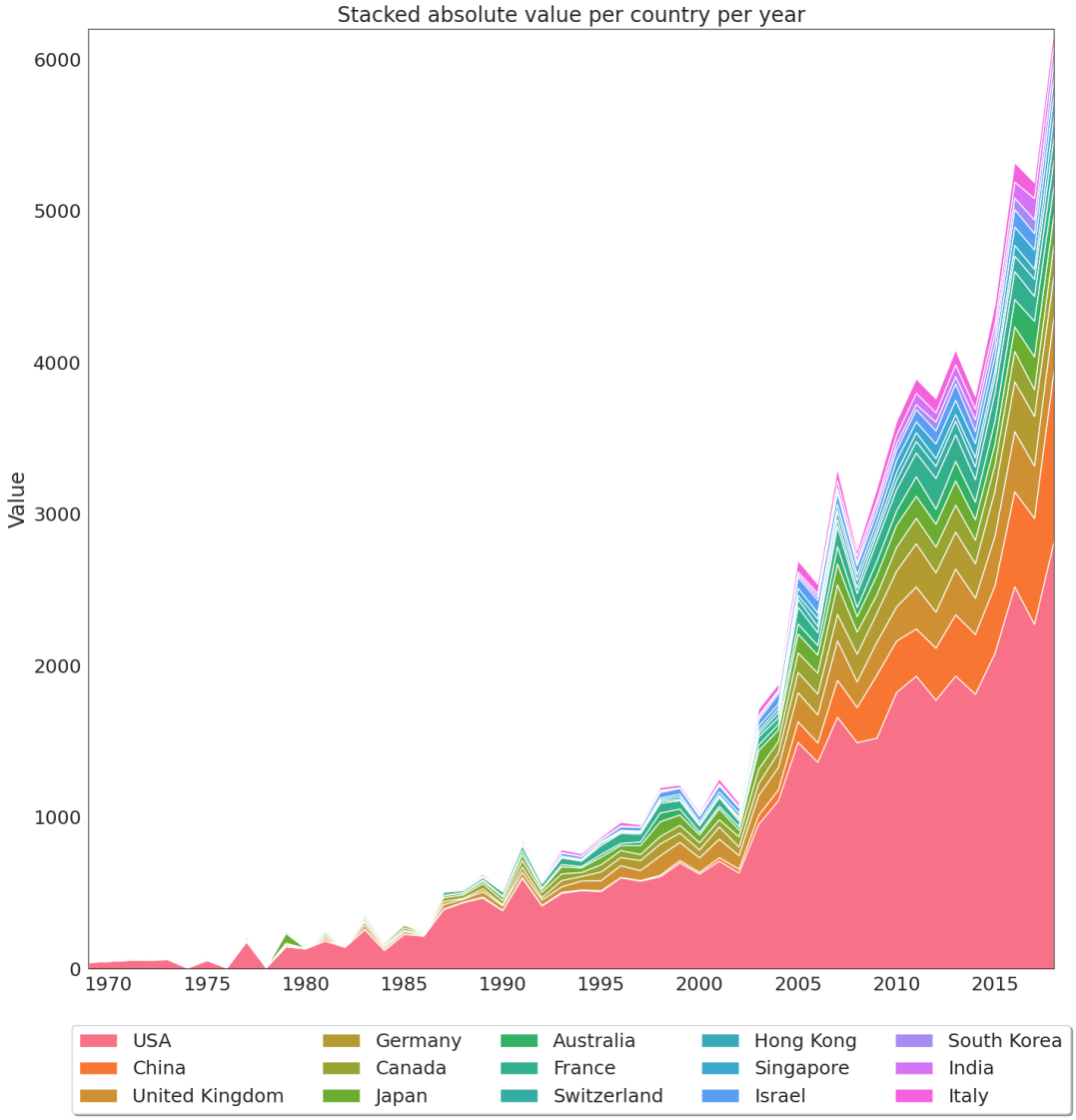
Source: The Author

Figure 4.24 – Stacked percentage of papers publishd per country per year, not considering the ones we can't infer.



Source: The Author

Figure 4.25 – Quantity of papers per country per year



Source: The Author

Figure 4.26 – Deteriorated countries stacked chart with Arnet's V13

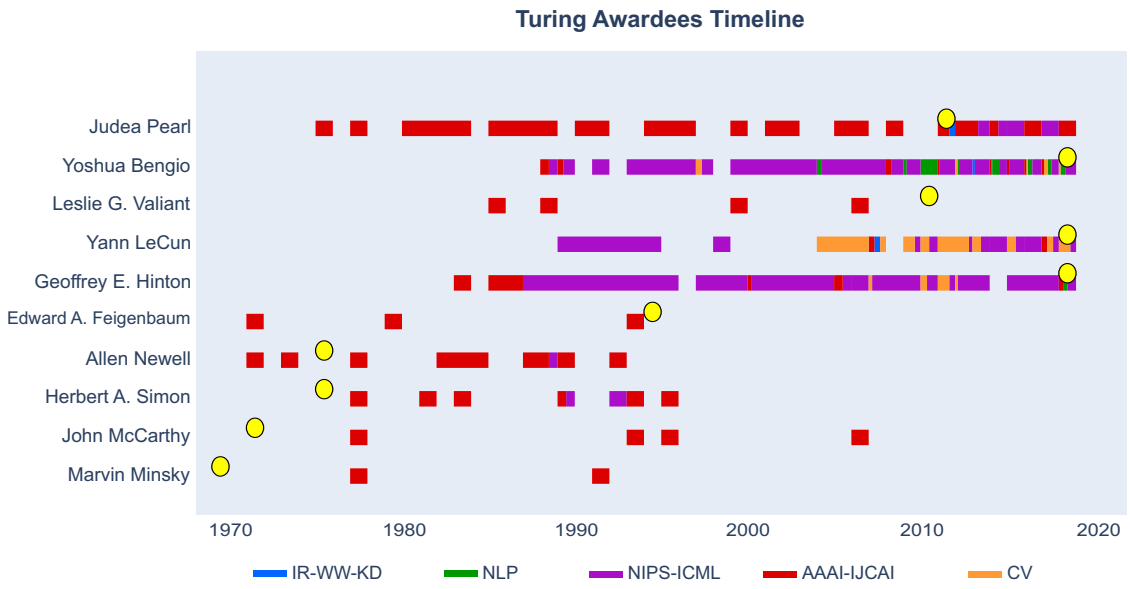
Source: The Author

4.7 Turing Award

As previously stated, the Turing Award has given seven prizes to researchers due to their efforts in AI and related areas, namely: Marvin Minsky (1969), John McCarthy (1971), Allen Newell and Herbert Simon (1975), Edward Feigenbaum and Raj Reddy (1994), Leslie Valiant (2010), Judea Pearl (2011) and Yoshua Bengio, Geoffrey Hinton and Yann LeCun (2018).

The Turing Award winners timeline (see Figure 4.27) depicts a change of focus of these highly prolific researchers over time: most recent awardees have their work divided into several venues (especially machine learning and computer vision related ones, such as NIPS and ICML and CVPR), while the older ones concentrated their efforts in AAAI or IJCAI. It needs to be taken into account, as well, that we are only considering conferences in this work, while most of the works published in the early days of Artificial Intelligence were published in journals.

Figure 4.27 – AI-related Turing Awardees timeline.



Each year is divided proportionally according to the amount of papers published in each group of venues.
Yellow ellipses are placed on the year each award was granted.
Source: The Author

We also verified the Spearman correlation between the titles of papers published by Turing Award winners and the titles of papers published in the selected AI conferences (AAAI, IJCAI, and NIPS) over time. To do so we compare the ranking of the TF-IDFs for

the words in the Turing Award winner's paper titles in that year, related to the ranking of the TF-IDFs conference's (or group of conferences) papers titles in the same year.

As the AI community, in general, has leaned towards a more connectionist approach in the last years, we expected to see a decreasing trend regarding old Turing Award winners who focused on symbolic AI and expert systems – or at least a very little correlation.

Nevertheless, the work of Marvin Minsky (Turing Award of 1969) is still quite in line with what is published in NIPS, for instance, despite being poorly correlated with the three conferences when they are considered altogether (see Figure 4.28). These correlations may be however not very realistic since there are only two papers by Marvin Minksy in the entire dataset. The most positive and significant slope, however, comes from the work of the latest Turing Award winners (Bengio, Hinton, and LeCun, 2018): their papers' titles have a positive and moderate correlation with all three conferences (and naturally with the average) and also show an increasing trend along the years, as depicted on Figure 4.29. The remaining plots can be found in Appendix G

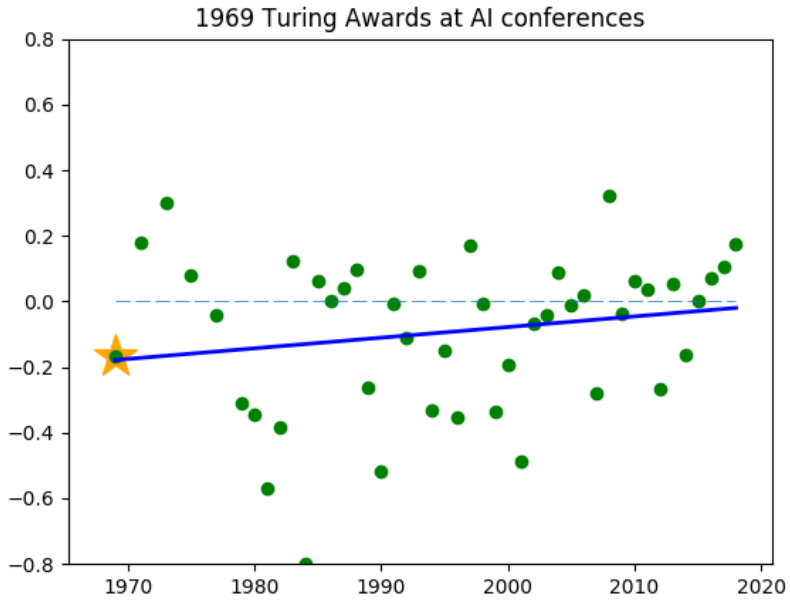
4.7.1 Turing Award Influence takeaway

Figure 4.29 clearly shows how the most recent Turing Awardees (Yoshua Bengio, Geoffrey Hinton, and Yann LeCun) influenced the area, with increasing rates of correlation over the years in all three main conferences from the Artificial Intelligence field. We predict that, in the next few years, if we were to plot the same data again, their correlation would have increased even further, showing that they were able to influence research in general. We base our hypothesis on the fact that the 1969 Turing Award Winner, Marvin Minsky, still has a positive correlation rate in some conferences such as NeurIPS, even though the same cannot be said for the AI field in general. Also, as noted in Chapter 2, Minsky possibly held AI away from the connectionist view with his book (MINSKY; PAPERT, 1969).

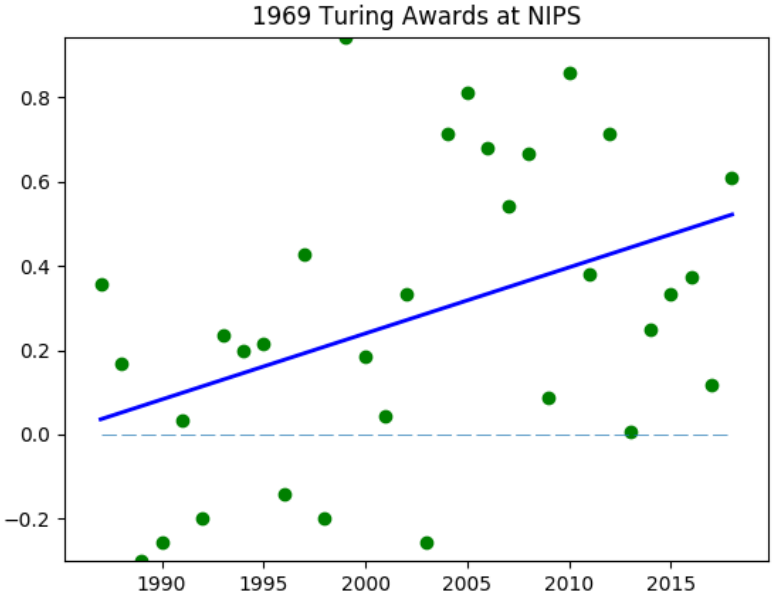
Similarly, this data can also be seen from the opposite side, if we consider the 2018 winners: they were closely following the trend of papers published in these conferences, therefore winning the Turing Award by researching the areas of interest. Even if possible, it is undeniable that their works are of huge importance, and influenced the area in ways not influenced by the others.

Also, it is worth of note that the winners of the Turing Award do not appear in the ranking of authors according to centralities in Section 3.3.1 and 4.3. This is probably

Figure 4.28 – 1969 Turing Award Correlation with AI conferences and NIPS specifically

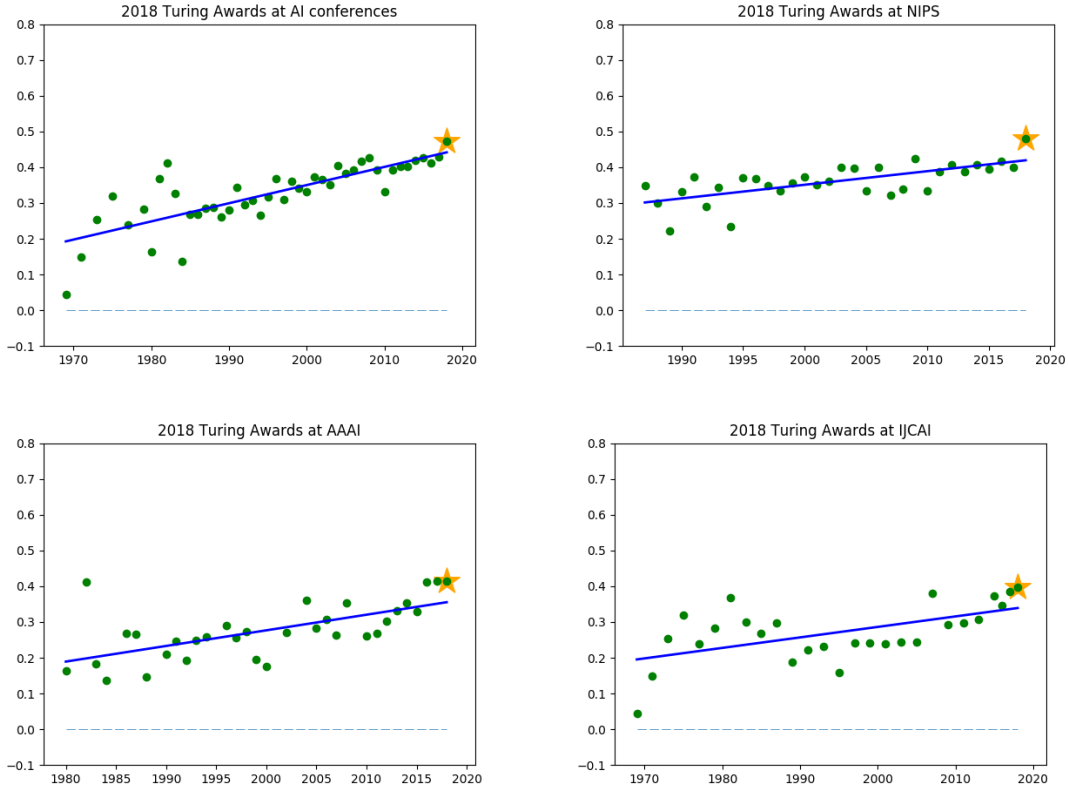


Correlation between titles of papers published by the Turing Award winner of 1969 and titles of all papers published yearly in AAAI, IJCAI and NIPS.



Correlation between titles of papers published by the Turing Award winner of 1969 and titles of all papers published only at NIPS.
Source: The Author

Figure 4.29 – Correlation between titles of papers published by 2018 Turing Award winners and titles of papers published in the three AI flagship conferences.



Source: The Author

because the Turing Awardees do not have enough papers published to be able to reach the top of the rankings, mainly due to their “celebrity” status, and also because they are mostly based around some seminal works.

5 CONCLUSION

5.1 Overview

Artificial Intelligence has gone further than anyone could have thought since (TURING, 1936), (MCCULLOCH; PITTS, 1943), and (TURING, 1950), being used all over academia and the industry in our era, powering multi-billion dollar companies. Some of the processes behind this evolution are not well understood, but this work makes it clear, with thorough research of how AI came to be, having a short but comprehensive survey on AI's history, defining the 5 time periods AI has already gone through. We also show a quick survey in graph centralities, and the Turing Award, necessary to better understand the overall picture of the area.

By analyzing Arnet's v11 dataset, a dataset based on the DBLP corpus, and enhancing it to a graph-based format, we intended to ease paper/author citation/collaboration network research. This dataset generated quite insightful graphs and could generate even more in future works. Also, the code presented in this work makes it such that it is pretty easy to extend it to any other underlying dataset, making it possible to generate and compute the same statistics presented in this work for any other area than AI.

With these graphs, we show insights on self-citations, new authors, and author and paper importance throughout the years. We also proposed a new type of dataset intended to be used as a knowledge graph source for recommender systems, where authors, papers, citations, and collaborations are all defined in the same graph. With the Country Citation Graph, we also introduced an important dataset and pipeline capable of inferring the country of affiliation of an author based on its organization.

Also, by investigating the Turing Award winners and comparing them against the data published we find out there is evidence that they actually "pull" their most common-published venues to their topic of research, at least for the most recent AI researchers winners.

Lastly, the study on countries' affiliation is, to the best of our knowledge, the first of its type, creating a new algorithm able to infer the country of affiliation of an author from its organization, as available at DBLP or Arnet.

5.2 Contributions

This work has some impactful and important contributions, including but not restricted to:

- Five new graph-based datasets, with fully computed centralities to ease paper/author citation/collaboration network research.
- Analyses for these graphs, focusing both on its raw structure and the centrality rankings
- Theoretical algorithms description, allowing anyone to replicate the graph building process in any programming language for any dataset
- Spearman Correlation computation between Turing Awardees papers and conference papers, showing they have a positive correlation

Besides the theoretical background generated in this project, there are also a few important software contributions. They can be read more about in Section H, but are here shortly outlined:

- Python library to convert an XML to JSON in a stream-fashion, i.e. without loading the whole XML and JSON files in memory
- Parallel Python implementation for the Betweenness and Closeness Centralities
- Novel Python implementation for a Graph Parsing pipeline, avoiding duplicate work through data caching
- Python implementation of the proposed algorithm to infer a paper country of affiliation

5.3 Future Work

The dataset created in this work provides uncountable possibilities for future work, especially when we think about the computed centralities. This work presented several analyses of the dataset, and one might think of even more possible ways to visualize it.

Additionally, it would be ideal if Coreness centrality (Section 2.4.1.5) was also computed for this dataset, as it displays the interesting feature of being a discrete value

instead of a continuous one, thus allowing you to more easily identify the most important authors/papers according to it.

The Country Citation Graph has a lot of potential in understanding "brain-draining" by investigating the flow of authors from one country of affiliation to others – easily done with our dataset, without any extra work besides counting the number of "transitions" between countries.

Similarly, we believe that comparing more advanced usages of this dataset with the Turing Awardees might bring even more interesting results. With a better dataset, one where there are abstract data available for every paper, one might be able to achieve better results when running a Spearman Correlation (Section4.7) between the text in Turing Award winners' abstracts and the ones from the remaining venues papers.

REFERENCES

ABBASI, A.; HOSSAIN, L.; LEYDESDORFF, L. Betweenness centrality as a driver of preferential attachment in the evolution of research collaboration networks. **Journal of Informetrics**, v. 6, n. 3, p. 403–412, 2012. ISSN 1751-1577. Available from Internet: <https://www.sciencedirect.com/science/article/pii/S175115771200003X>.

ABBOUD, R. et al. **The Surprising Power of Graph Neural Networks with Random Node Initialization**. arXiv, 2020. Available from Internet: <https://arxiv.org/abs/2010.01179>.

AUDIBERT, R. B. **streamxml2json Library**. 2022. Available from Internet: <https://github.com/rafaeelaudibert/streamxml2json>.

AUDIBERT, R. B. **streamxml2json Library PyPI**. 2022. Available from Internet: <https://pypi.org/project/streamxml2json/>.

AVIN, C.; SHPITSER, I.; PEARL, J. Identifiability of path-specific effects. In: **Proceedings of the 19th International Joint Conference on Artificial Intelligence**. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2005. (IJCAI'05), p. 357–363.

BAHDANAU, D.; CHO, K.; BENGIO, Y. **Neural Machine Translation by Jointly Learning to Align and Translate**. arXiv, 2014. Available from Internet: <https://arxiv.org/abs/1409.0473>.

BELTAGY, I.; LO, K.; COHAN, A. SciBERT: A pretrained language model for scientific text. In: **Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)**. Hong Kong, China: Association for Computational Linguistics, 2019. p. 3615–3620. Available from Internet: <https://aclanthology.org/D19-1371>.

BLEI, D. M.; NG, A. Y.; JORDAN, M. I. Latent dirichlet allocation. **J. Mach. Learn. Res.**, JMLR.org, v. 3, n. null, p. 993–1022, mar 2003. ISSN 1532-4435.

BORDES, A. et al. Learning structured embeddings of knowledge bases. In: **Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence**. [S.l.]: AAAI Press, 2011. (AAAI'11), p. 301–306.

BOUREAU, Y.-L. et al. Learning mid-level features for recognition. In: **2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2010. p. 2559–2566.

BRIN, S.; PAGE, L. The anatomy of a large-scale hypertextual web search engine. **Computer Networks**, v. 30, n. 1-7, p. 107–117, 1998.

BROWN, T. B. et al. Language models are few-shot learners. **CoRR**, abs/2005.14165, 2020. Available from Internet: <https://arxiv.org/abs/2005.14165>.

CAMPBELL, M.; HOANE, A.; HSU, F. hsiung. Deep blue. **Artificial Intelligence**, v. 134, n. 1, p. 57–83, 2002. ISSN 0004-3702. Available from Internet: <https://www.sciencedirect.com/science/article/pii/S0004370201001291>.

CARDENTE, J. Using centrality measures to identify key members of an innovation collaboration network. In: . [S.l.: s.n.], 2012.

CHEN, D. et al. Identifying influential nodes in complex networks. **Physica A: Statistical Mechanics and its Applications**, v. 391, n. 4, p. 1777–1787, 2012. ISSN 0378-4371. Available from Internet: <https://www.sciencedirect.com/science/article/pii/S0378437111007333>.

CHEN, T. et al. **A Simple Framework for Contrastive Learning of Visual Representations**. arXiv, 2020. Available from Internet: <https://arxiv.org/abs/2002.05709>.

CHEN, X.; HE, K. **Exploring Simple Siamese Representation Learning**. arXiv, 2020. Available from Internet: <https://arxiv.org/abs/2011.10566>.

CLARK, P.; TAFJORD, O.; RICHARDSON, K. **Transformers as Soft Reasoners over Language**. arXiv, 2020. Available from Internet: <https://arxiv.org/abs/2002.05867>.

COBBE, K. et al. Training verifiers to solve math word problems. **CoRR**, abs/2110.14168, 2021. Available from Internet: <https://arxiv.org/abs/2110.14168>.

COOK, S. A. The complexity of theorem-proving procedures. In: **Proceedings of the Third Annual ACM Symposium on Theory of Computing**. New York, NY, USA: Association for Computing Machinery, 1971. (STOC '71), p. 151–158. ISBN 9781450374644. Available from Internet: <https://doi.org/10.1145/800157.805047>.

CSRANKINGS. **CSRankings**. 2019. Available from Internet: <http://csrankings.org/>.

DBLP. **DBLP network dataset**. 2019. Available from Internet: <https://dblp.uni-trier.de/>.

DEVELOPERS, T. P. **PIP - Python Package Manager**. 2008. Available from Internet: <https://pypi.org/project/pip/>.

DEVLIN, J. et al. **BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding**. arXiv, 2018. Available from Internet: <https://arxiv.org/abs/1810.04805>.

DHAR, V.; POPLE, H. E. Rule-based versus structure-based models for explaining and generating expert behavior. **Commun. ACM**, Association for Computing Machinery, New York, NY, USA, v. 30, n. 6, p. 542–555, jun 1987. ISSN 0001-0782. Available from Internet: <https://doi.org/10.1145/214762.214773>.

DICKMANNS, E. D. An integrated approach to feature based dynamic vision. In: **IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 1988, 5-9 June, 1988, Ann Arbor, Michigan, USA**. IEEE, 1988. p. 820–825. Available from Internet: <https://doi.org/10.1109/CVPR.1988.196328>.

DING, Y. et al. Pagerank for ranking authors in co-citation networks. **CoRR**, abs/1012.4872, 2010. Available from Internet: <http://arxiv.org/abs/1012.4872>.

ERMAN, L. D.; LESSER, V. R. A multi-level organization for problem solving using many, diverse, cooperating sources of knowledge. In: **Advance Papers of the Fourth International Joint Conference on Artificial Intelligence, Tbilisi, Georgia, USSR, September 3-8, 1975**. [S.l.: s.n.], 1975. p. 483–490.

EVANS, T. G. A heuristic program to solve geometric-analogy problems. In: **Proceedings of the April 21-23, 1964, Spring Joint Computer Conference**. New York, NY, USA: Association for Computing Machinery, 1964. (AFIPS '64 (Spring)), p. 327–338. ISBN 9781450378901. Available from Internet: <https://doi.org/10.1145/1464122.1464156>.

FEIGENBAUM, E. A. The art of artificial intelligence: Themes and case studies of knowledge engineering. In: **Proceedings of the 5th International Joint Conference on Artificial Intelligence. Cambridge, MA, USA, August 22-25, 1977**. [S.l.: s.n.], 1977. p. 1014–1029.

FEIGENBAUM, E. A.; FELDMAN, J. **Computers and Thought**. USA: McGraw-Hill, Inc., 1963. ISBN 0070203709.

FELDMAN, J. A. et al. The stanford hand-eye project. In: **Proceedings of the 1st International Joint Conference on Artificial Intelligence**. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1969. (IJCAI'69), p. 521–526.

FENG, Z. et al. **CodeBERT: A Pre-Trained Model for Programming and Natural Languages**. arXiv, 2020. Available from Internet: <https://arxiv.org/abs/2002.08155>.

FREEMAN, L. C. A set of measures of centrality based on betweenness. **Sociometry**, [American Sociological Association, Sage Publications, Inc.], v. 40, n. 1, p. 35–41, 1977. ISSN 00380431. Available from Internet: <http://www.jstor.org/stable/3033543>.

FREEMAN, L. C. Centrality in social networks conceptual clarification. **Social Networks**, v. 1, n. 3, p. 215–239, 1978. ISSN 0378-8733. Available from Internet: <https://www.sciencedirect.com/science/article/pii/0378873378900217>.

FREITAG, D. Machine learning for information extraction in informal domains. **Machine Learning**, v. 39, n. 2, p. 169–202, May 2000. ISSN 1573-0565. Available from Internet: <https://doi.org/10.1023/A:1007601113994>.

FU, J. et al. **Dual Attention Network for Scene Segmentation**. arXiv, 2018. Available from Internet: <https://arxiv.org/abs/1809.02983>.

GLOROT, X.; BENGIO, Y. Understanding the difficulty of training deep feedforward neural networks. In: TEH, Y. W.; TITTERINGTON, M. (Ed.). **Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics**. Chia Laguna Resort, Sardinia, Italy: PMLR, 2010. (Proceedings of Machine Learning Research, v. 9), p. 249–256. Available from Internet: <https://proceedings.mlr.press/v9/glorot10a.html>.

GOMEZ-URIBE, C. A.; HUNT, N. The netflix recommender system: Algorithms, business value, and innovation. **ACM Trans. Manage. Inf. Syst.**, Association for Computing Machinery, New York, NY, USA, v. 6, n. 4, dec 2016. ISSN 2158-656X. Available from Internet: <https://doi.org/10.1145/2843948>.

GOODFELLOW, I. J. et al. **Generative Adversarial Networks**. arXiv, 2014. Available from Internet: <https://arxiv.org/abs/1406.2661>.

GREEN, C. Application of theorem proving to problem solving. In: **Proceedings of the 1st International Joint Conference on Artificial Intelligence**. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1969. (IJCAI'69), p. 219–239.

GRUBBS, J. C.; GLASS, R. I.; KILMARX, P. H. Coauthor Country Affiliations in International Collaborative Research Funded by the US National Institutes of Health, 2009 to 2017. **JAMA Network Open**, v. 2, n. 11, p. e1915989–e1915989, 11 2019. ISSN 2574-3805. Available from Internet: <https://doi.org/10.1001/jamanetworkopen.2019.15989>.

GUNS, R.; LIU, Y.; MAHBUBA, D. Q-measures and betweenness centrality in a collaboration network: A case study of the field of informetrics. **Scientometrics**, v. 87, p. 133–147, 04 2011.

HAGBERG, A.; SWART, P.; CHULT, D. Exploring network structure, dynamics, and function using networkx. In: . [S.l.: s.n.], 2008.

HARTLEY, R.; ZISSERMAN, A. **Multiple View Geometry in Computer Vision**. 2. ed. USA: Cambridge University Press, 2003. ISBN 0521540518.

HASSELT, H. van; GUEZ, A.; SILVER, D. **Deep Reinforcement Learning with Double Q-learning**. arXiv, 2015. Available from Internet: <https://arxiv.org/abs/1509.06461>.

HE, X. et al. **LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation**. arXiv, 2020. Available from Internet: <https://arxiv.org/abs/2002.02126>.

HIRSCH, J. E. An index to quantify an individual's scientific research output. **Proceedings of the National Academy of Sciences**, v. 102, n. 46, p. 16569–16572, 2005. Available from Internet: <https://www.pnas.org/doi/abs/10.1073/pnas.0507655102>.

HOPFIELD, J. J. Neural networks and physical systems with emergent collective computational abilities. **Proceedings of the National Academy of Sciences**, v. 79, n. 8, p. 2554–2558, 1982. Available from Internet: <https://www.pnas.org/doi/abs/10.1073/pnas.79.8.2554>.

HOTTENROTT, H.; ROSE, M.; LAWSON, C. **The Rise of Multiple Institutional Affiliations in Academia**. arXiv, 2019. Available from Internet: <https://arxiv.org/abs/1912.05576>.

HWANG, T. **Deepfakes: A Grounded Threat Assessment**. Georgetown University, 2020. Available from Internet: <cset.georgetown.edu/research/deepfakes-a-grounded-threat-assessment/.https://doi.org/10.51593/20190030>.

IOFFE, S.; SZEGEDY, C. **Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift**. arXiv, 2015. Available from Internet: <https://arxiv.org/abs/1502.03167>.

JIAO, X. et al. **TinyBERT: Distilling BERT for Natural Language Understanding**. arXiv, 2019. Available from Internet: <https://arxiv.org/abs/1909.10351>.

JIN, W. et al. **Graph Structure Learning for Robust Graph Neural Networks**. arXiv, 2020. Available from Internet: <https://arxiv.org/abs/2005.10203>.

JUMPER, J. et al. Highly accurate protein structure prediction with alphafold. **Nature**, v. 596, n. 7873, p. 583–589, Aug 2021. ISSN 1476-4687. Available from Internet: <https://doi.org/10.1038/s41586-021-03819-2>.

KARP, R. M. Reducibility among combinatorial problems. In: MILLER, R. E.; THATCHER, J. W. (Ed.). **Proceedings of a symposium on the Complexity of Computer Computations, held March 20-22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York, USA**. Plenum Press, New York, 1972. (The IBM Research Symposia Series), p. 85–103. Available from Internet: <https://doi.org/10.1007/978-1-4684-2001-2_9>.

KARRAS, T.; LAINE, S.; AILA, T. **A Style-Based Generator Architecture for Generative Adversarial Networks**. arXiv, 2018. Available from Internet: <https://arxiv.org/abs/1812.04948>.

KITSAK, M. et al. Identification of influential spreaders in complex networks. **Nature Physics**, v. 6, n. 11, p. 888–893, Nov 2010. ISSN 1745-2481. Available from Internet: <https://doi.org/10.1038/nphys1746>.

KREBS, V. **Uncloaking Terrorist Networks**. 2002. Available from Internet: <https://firstmonday.org/ojs/index.php/fm/article/view/941/863>.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: PEREIRA, F. et al. (Ed.). **Advances in Neural Information Processing Systems**. Curran Associates, Inc., 2012. v. 25. Available from Internet: <https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf>.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **nature**, Nature Publishing Group, v. 521, n. 7553, p. 436, 2015.

LECUN, Y. et al. Backpropagation applied to handwritten zip code recognition. **Neural Computation**, v. 1, n. 4, p. 541–551, 1989.

LECUN, Y. et al. Gradient-based learning applied to document recognition. **Proceedings of the IEEE**, v. 86, n. 11, p. 2278–2324, 1998.

LECUN, Y.; HUANG, F. J.; BOTTOU, L. Learning methods for generic object recognition with invariance to pose and lighting. In: **Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.** [S.l.: s.n.], 2004. v. 2, p. II–104 Vol.2.

LEYDESDORFF, L. Betweenness centrality as an indicator of the interdisciplinarity of scientific journals. **J. Am. Soc. Inf. Sci. Technol.**, John Wiley & Sons, Inc., New York, NY, USA, v. 58, n. 9, p. 1303–1319, jul. 2007. ISSN 1532-2882. Available from Internet: <http://dx.doi.org/10.1002/asi.20614>.

LI, G. et al. **Unicoder-VL: A Universal Encoder for Vision and Language by Cross-modal Pre-training**. arXiv, 2019. Available from Internet: <https://arxiv.org/abs/1908.06066>.

LIU, M.; GAO, H.; JI, S. Towards deeper graph neural networks. In: **Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining**. ACM, 2020. Available from Internet: <https://doi.org/10.1145\%2F3394486.3403076>.

LLOYD, J. W. **Foundations of Logic Programming**. Berlin, Heidelberg: Springer-Verlag, 1984. ISBN 0387132996.

MAHABADI, R. K. et al. Perfect: Prompt-free and efficient few-shot learning with language models. **Association for Computational Linguistics (ACL)**, 2022. Available from Internet: <https://research.facebook.com/publications/perfect-prompt-free-and-efficient-few-shot-learning-with-language-models/>.

MARCUS, G. **Deep Learning: A Critical Appraisal**. arXiv, 2018. Available from Internet: <https://arxiv.org/abs/1801.00631>.

MARCUS, G. The next decade in AI: four steps towards robust artificial intelligence. **CoRR**, abs/2002.06177, 2020. Available from Internet: <https://arxiv.org/abs/2002.06177>.

MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. **The bulletin of mathematical biophysics**, v. 5, n. 4, p. 115–133, Dec 1943. ISSN 1522-9602. Available from Internet: <https://doi.org/10.1007/BF02478259>.

MICHALSKA-SMITH, M. et al. The scientific impact of nations: Journal placement and citation performance. **PloS one**, v. 9, p. e109195, 10 2014.

MIJWIL, M.; ABTTAN, R. Artificial intelligence: A survey on evolution and future trends. **Asian Journal of Applied Sciences**, v. 9, p. 87–93, 04 2021.

MINSKY, M. Steps toward artificial intelligence. **Proceedings of the IRE**, v. 49, n. 1, p. 8–30, 1961.

MINSKY, M.; PAPERT, S. **Perceptrons: An Introduction to Computational Geometry**. Cambridge, MA, USA: MIT Press, 1969.

MOORE, R. C. Reasoning about knowledge and action. In: **Proceedings of the 5th International Joint Conference on Artificial Intelligence. Cambridge, MA, USA, August 22-25, 1977**. [S.l.: s.n.], 1977. p. 223–227.

MORAVEC, H. **Mind Children: The Future of Robot and Human Intelligence**. USA: Harvard University Press, 1988. ISBN 0674576160.

MORAVEC, H. P. Towards automatic visual obstacle avoidance. In: **Proceedings of the 5th International Joint Conference on Artificial Intelligence. Cambridge, MA, USA, August 22-25, 1977**. [S.l.: s.n.], 1977. p. 584.

NAIR, V.; HINTON, G. E. Rectified linear units improve restricted boltzmann machines. In: **Proceedings of the 27th International Conference on International Conference on Machine Learning**. Madison, WI, USA: Omnipress, 2010. (ICML'10), p. 807–814. ISBN 9781605589077.

NG, A. Y.; JORDAN, M. I.; WEISS, Y. On spectral clustering: Analysis and an algorithm. In: **Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic**. Cambridge, MA, USA: MIT Press, 2001. (NIPS'01), p. 849–856.

NGUYEN, T. T. et al. Deep learning for deepfakes creation and detection. **CoRR**, abs/1909.11573, 2019. Available from Internet: <http://arxiv.org/abs/1909.11573>.

NILSSON, N. J. A mobius automation: An application of artificial intelligence techniques. In: **Proceedings of the 1st International Joint Conference on Artificial Intelligence**. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1969. (IJCAI'69), p. 509–520.

OKE, S. A literature review on artificial intelligence. **International Journal of Information and Management Sciences**, v. 19, p. 535–570, 12 2008.

PARK, T. et al. Gaugan: Semantic image synthesis with spatially adaptive normalization. In: **ACM SIGGRAPH 2019 Real-Time Live!** New York, NY, USA: Association for Computing Machinery, 2019. (SIGGRAPH '19). ISBN 9781450363150. Available from Internet: <https://doi.org/10.1145/3306305.3332370>.

PASCANU, R.; MIKOLOV, T.; BENGIO, Y. **On the difficulty of training Recurrent Neural Networks**. arXiv, 2012. Available from Internet: <https://arxiv.org/abs/1211.5063>.

PASZKE, A. et al. **PyTorch: An Imperative Style, High-Performance Deep Learning Library**. arXiv, 2019. Available from Internet: <https://arxiv.org/abs/1912.01703>.

PEARL, J. **Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference**. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1988. ISBN 0934613737.

PEARL, J. **Causality: Models, Reasoning and Inference**. 2nd. ed. USA: Cambridge University Press, 2009. ISBN 052189560X.

QIU, J. et al. GCC. In: **Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining**. ACM, 2020. Available from Internet: <https://doi.org/10.1145\%2F3394486.3403168>.

RAMESH, A. et al. Zero-shot text-to-image generation. **CoRR**, abs/2102.12092, 2021. Available from Internet: <https://arxiv.org/abs/2102.12092>.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning representations by back-propagating errors. **Nature**, v. 323, n. 6088, p. 533–536, Oct 1986. ISSN 1476-4687. Available from Internet: <https://doi.org/10.1038/323533a0>.

SABIDUSSI, G. The centrality index of a graph. **Psychometrika**, v. 31, n. 4, p. 581–603, Dec 1966. ISSN 1860-0980. Available from Internet: <https://doi.org/10.1007/BF02289527>.

SAXENA, A.; IYENGAR, S. **Centrality Measures in Complex Networks: A Survey**. arXiv, 2020. Available from Internet: <https://arxiv.org/abs/2011.07190>.

SHAHAM, T. R.; DEKEL, T.; MICHAELI, T. **SinGAN: Learning a Generative Model from a Single Natural Image**. arXiv, 2019. Available from Internet: <https://arxiv.org/abs/1905.01164>.

SILVER, D. et al. Mastering the game of go with deep neural networks and tree search. **Nature**, v. 529, n. 7587, p. 484–489, Jan 2016. ISSN 1476-4687. Available from Internet: <https://doi.org/10.1038/nature16961>.

SIMONYAN, K.; ZISSERMAN, A. **Very Deep Convolutional Networks for Large-Scale Image Recognition**. arXiv, 2014. Available from Internet: <https://arxiv.org/abs/1409.1556>.

STAFF, C. 2007. Available from Internet: <https://web.archive.org/web/20090323234900/http://awards.acm.org/prizes.cfm>.

STAFF, C. Acm's turing award prize raised to $1 million. **Commun. ACM**, Association for Computing Machinery, New York, NY, USA, v. 57, n. 12, p. 20, nov 2014. ISSN 0001-0782. Available from Internet: <https://doi.org/10.1145/2685372>.

STRUBELL, E.; GANESH, A.; MCCALLUM, A. **Energy and Policy Considerations for Deep Learning in NLP**. arXiv, 2019. Available from Internet: <https://arxiv.org/abs/1906.02243>.

TANASE, D. et al. Emotions and activity profiles of influential users in product reviews communities. **Frontiers in Physics**, v. 3, 11 2015.

TANG, J. et al. Arnetminer: extraction and mining of academic social networks. In: **KDD**. [S.l.]: ACM, 2008. p. 990–998.

TEED, Z.; DENG, J. **RAFT: Recurrent All-Pairs Field Transforms for Optical Flow**. arXiv, 2020. Available from Internet: <https://arxiv.org/abs/2003.12039>.

TESAURO, G. Td-gammon, a self-teaching backgammon program, achieves master-level play. **Neural Comput.**, MIT Press, Cambridge, MA, USA, v. 6, n. 2, p. 215–219, mar 1994. ISSN 0899-7667. Available from Internet: <https://doi.org/10.1162/neco.1994.6.2.215>.

THOMANEK, F.; DICKMANNS, E. D. Autonomous road vehicle guidance in normal traffic. In: LI, S. Z. et al. (Ed.). **Recent Developments in Computer Vision, Second Asian Conference on Computer Vision, ACCV '95, Singapore, December 5-8, 1995, Invited Session Papers**. Springer, 1995. (Lecture Notes in Computer Science, v. 1035), p. 499–507. Available from Internet: <https://doi.org/10.1007/3-540-60793-5_103>.

TURING, A. On computable numbers, with an application to the n tscheidungsproblem. **Proceedings of the London Mathematical Society**, Association for Symbolic Logic, v. 42, n. 1, p. 230–265, 1936.

TURING, A. M. Computing machinery and intelligence. **Mind**, Oxford University Press on behalf of the Mind Association, v. 59, n. 236, p. 433–460, 1950. ISSN 00264423. Available from Internet: <http://www.jstor.org/stable/2251299>.

VALIANT, L. G. A theory of the learnable. **Commun. ACM**, Association for Computing Machinery, New York, NY, USA, v. 27, n. 11, p. 1134–1142, nov 1984. ISSN 0001-0782. Available from Internet: <https://doi.org/10.1145/1968.1972>.

VERMA, V. et al. **Interpolation Consistency Training for Semi-Supervised Learning**. arXiv, 2019. Available from Internet: <https://arxiv.org/abs/1903.03825>.

WAGSTAFF, K. et al. Constrained k-means clustering with background knowledge. In: **Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, June 28 - July 1, 2001**. [S.l.: s.n.], 2001. p. 577–584.

WAN, Z. et al. A survey on centrality metrics and their network resilience analysis. **IEEE Access**, v. 9, p. 104773–104819, 2021.

WANG, Z. et al. Global context enhanced graph neural networks for session-based recommendation. In: **Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval**. ACM, 2020. Available from Internet: <https://doi.org/10.1145%2F3397271.3401142>.

WARTBURG, I. von; ROST, K.; TEICHERT, T. Inventive progress measured by patent citation network analysis: Technological change in variable valve actuation for internal combustion engines. 04 2022.

WEHMUTH, K.; ZIVIANI, A. Daccer: Distributed assessment of the closeness centrality ranking in complex networks. **Computer Networks**, v. 57, n. 13, p. 2536–2548, 2013. ISSN 1389-1286. Available from Internet: <https://www.sciencedirect.com/science/article/pii/S1389128613001412>.

WEIZENBAUM, J. Eliza—a computer program for the study of natural language communication between man and machine. **Commun. ACM**, Association for Computing Machinery, New York, NY, USA, v. 9, n. 1, p. 36–45, jan 1966. ISSN 0001-0782. Available from Internet: <https://doi.org/10.1145/365153.365168>.

WU, J. et al. Self-supervised graph learning for recommendation. In: **Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval**. ACM, 2021. Available from Internet: <https://doi.org/10.1145%2F3404835.3462862>.

WU, W. et al. Analysis of scientific collaboration networks among authors, institutions, and countries studying adolescent myopia prevention and control: A review article. **Iranian journal of public health**, Tehran University of Medical Sciences, v. 48, n. 4, p. 621–631, Apr 2019. ISSN 2251-6085. 31110972[pmid]. Available from Internet: <https://pubmed.ncbi.nlm.nih.gov/31110972>.

WU, Z. et al. **Connecting the Dots: Multivariate Time Series Forecasting with Graph Neural Networks**. arXiv, 2020. Available from Internet: <https://arxiv.org/abs/2005.11650>.

XIA, R. et al. Supervised hashing for image retrieval via image representation learning. In: **Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence**. [S.l.]: AAAI Press, 2014. (AAAI'14), p. 2156–2162.

XU, F. et al. Explainable ai: A brief survey on history, research areas, approaches and challenges. In: TANG, J. et al. (Ed.). **Natural Language Processing and Chinese Computing**. Cham: Springer International Publishing, 2019. p. 563–574. ISBN 978-3-030-32236-6.

YU, J. et al. **A paper's corresponding affiliation and first affiliation are consistent at the country level in Web of Science**. arXiv, 2021. Available from Internet: <https://arxiv.org/abs/2101.09426>.

YUILLE, A. L.; COHEN, D. S.; HALLINAN, P. W. Feature extraction from faces using deformable templates. In: **IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 1989, 4-8 June, 1989, San Diego, CA, USA**. [S.l.: s.n.], 1989. p. 104–109.

ZHANG, H. et al. **Self-Attention Generative Adversarial Networks**. arXiv, 2018. Available from Internet: <https://arxiv.org/abs/1805.08318>.

APPENDIX A — ARNET DATASET

This chapter presents some extra data in the Arnet Dataset, which did not fit in the main work.

Tables A.1 , A.2, and A.3 display the inner objects we referenced in Table 3.2.

Table A.4 shows a manual count of papers published at AAAI, NeurIPS and IJCAI. This was built to support Figure 3.3 and bring the point that the v13 dataset did not had not even half the data present at these conferences. In this table a question mark ("?") indicates that we could not infer the number of papers for that conference in that year.

To aggregate the data present in this table, we used AAAI's website statistics, the NeurIPS API and the IJCAI Proceedings page. The harder to get this data was IJCAI as they do not have anything the count of papers anywhere, only containing links to every paper published in every year, therefore we built a Javascript snippet that counted the number of links in those pages. Ultimately, we could not use this script in the years 1979 and 2001 because their data is badly formatted.

Figure A.1 shows an example of data present in Arnet dataset, and used throughout our work when we needed an example.

Table A.1 – Data structure for an *Author* entry in the Arnet JSON dataset

Field Name	Type	Description
id*	*string*	Unique ID for the author - unique across papers
name*	*string*	Full author name
org?*	*string*	Name of the organization this author was in when of this paper

"" indicates the field was used in this work*
"?" indicates the field is optional
Source: The Author

Table A.2 – Data structure for a *Venue* entry in the Arnet JSON dataset

Field Name	Type	Description
id*	*string*	Unique ID for the venue - unique across papers
raw*	*string*	The raw name of the venue
name?	*string*	Humand readable name of the venue

"" indicates the field was used in this work*
"?" indicates the field is optional
Source: The Author

Table A.3 – Data structure for a *IndexedAbstract* entry in the Arnet JSON dataset

Field Name	Type	Description
IndexLength	*integer*	How many words we have in the abstract
InvertedIndex*	*object<string, integer[]>*	Inverted index structure with the position of every word in the paper abstract

"" indicates the field was used in this work*
Source: The Author

Table A.4 – Manual count of papers per main AI conference per year

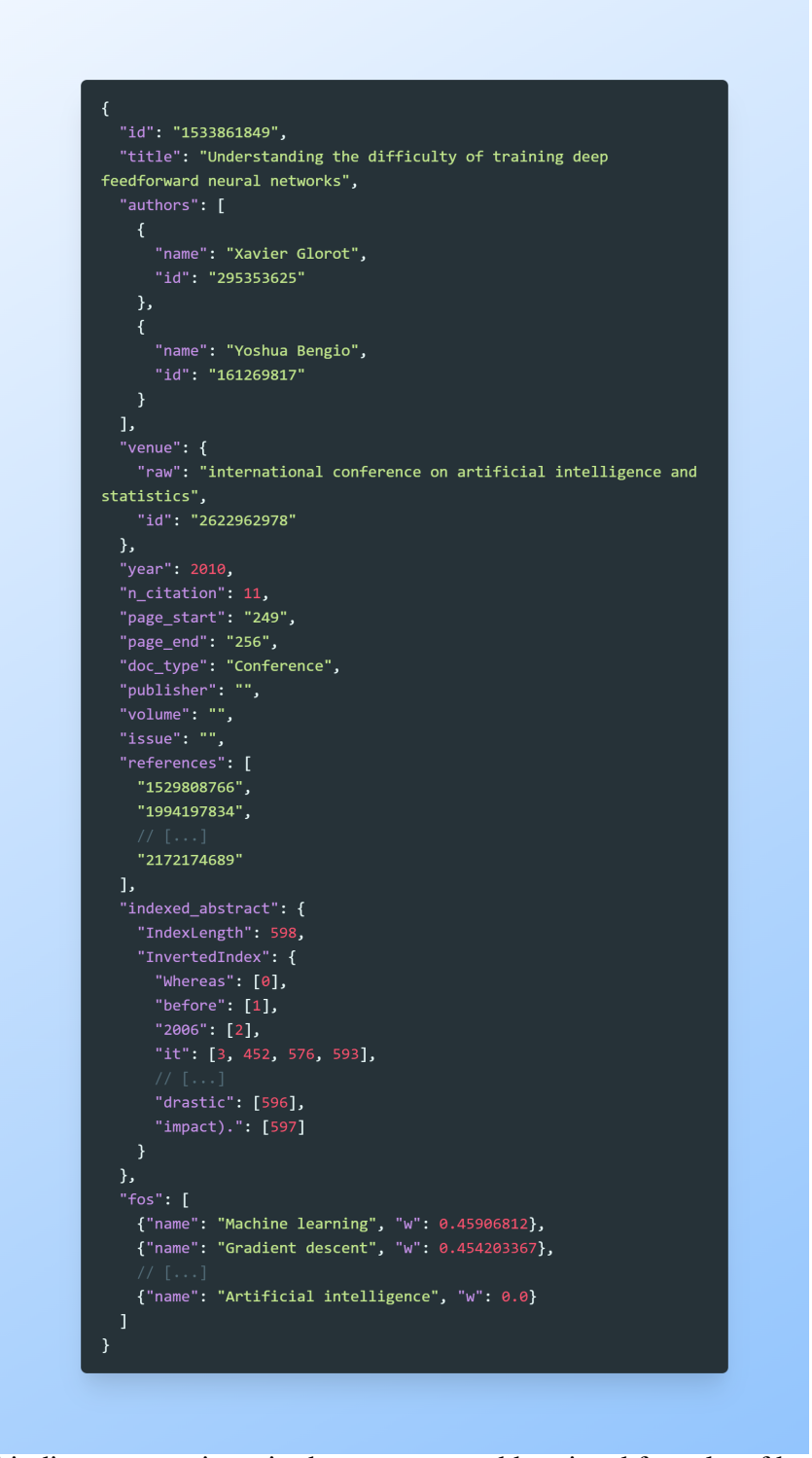
	AAAI	**NeurIPS**	**IJCAI**
1969	0	0	63
1970	0	0	0
1971	0	0	58
1972	0	0	0
1973	0	0	77
1974	0	0	0
1975	0	0	141
1976	0	0	0
1977	0	0	200
1978	0	0	0
1979	0	0	*?*
1980	*?*	0	0
1981	0	0	106
1982	*?*	0	0
1983	*?*	0	233
1984	*?*	0	0
1985	0	0	257
1986	*?*	0	0
1987	*?*	90	301
1988	*?*	94	0
1989	0	101	270
1990	0	143	0
1991	*?*	144	190
1992	*?*	127	0
1993	*?*	158	137
1994	341	140	0
1995	0	152	275
1996	336	152	0
1997	268	150	183
1998	269	151	0
1999	235	150	203

Table A.4 continued from previous page

	AAAI	NeurIPS	IJCAI
2000	265	152	0
2001	0	197	*?*
2002	256	207	0
2003	0	198	297
2004	250	207	0
2005	530	207	340
2006	718	204	0
2007	702	207	480
2008	648	250	0
2009	0	262	331
2010	780	292	0
2011	743	306	494
2012	707	370	0
2013	720	360	484
2014	912	411	0
2015	1101	403	656
2016	1163	569	658
2017	1049	679	782
2018	1201	1009	871
2019	1150	1428	965
2020	1591	1898	779
2021	1692	2334	722
Total	17627	13902	10553

Data for AAAI was extracted from their API; for NeurIPS it was extracted from their official statistics website; and for IJCAI it was manually (using JavaScript) counted on their website.
Cells with a "?" text indicate the years we were not able to find an accurate count of papers for that conference, reinforcing the fact that this is a lower-bound estimate.
Source: The Author

Figure A.1 – Example of a JSON entry for (GLOROT; BENGIO, 2010) in the Arnet dataset




```
{
  "id": "1533861849",
  "title": "Understanding the difficulty of training deep
feedforward neural networks",
  "authors": [
    {
      "name": "Xavier Glorot",
      "id": "295353625"
    },
    {
      "name": "Yoshua Bengio",
      "id": "161269817"
    }
  ],
  "venue": {
    "raw": "international conference on artificial intelligence and
statistics",
    "id": "2622962978"
  },
  "year": 2010,
  "n_citation": 11,
  "page_start": "249",
  "page_end": "256",
  "doc_type": "Conference",
  "publisher": "",
  "volume": "",
  "issue": "",
  "references": [
    "1529808766",
    "1994197834",
    // [...]
    "2172174689"
  ],
  "indexed_abstract": {
    "IndexLength": 598,
    "InvertedIndex": {
      "Whereas": [0],
      "before": [1],
      "2006": [2],
      "it": [3, 452, 576, 593],
      // [...]
      "drastic": [596],
      "impact).": [597]
    }
  },
  "fos": [
    {"name": "Machine learning", "w": 0.45906812},
    {"name": "Gradient descent", "w": 0.454203367},
    // [...]
    {"name": "Artificial intelligence", "w": 0.0}
  ]
}
```


[...] indicates some itens in the array were abbreviated for sake of brevity
Source: The author and (TANG et al., 2008)

APPENDIX B — CODEBASE

In this chapter, we present Algorithm 4 used in Section 3.3.5.1 along Algorithm 3 to be able to infer a country of origin from an organization. This basically removes the clutter present in Arnet's data.

Table B.1 shows every open-source library used to develop this work. We are very thankful for every library contributor's work to the open source community.

Table B.1 – Python libraries used in this work

Library Name	**Usage**
click	Create CLI to run experiments with different parameters
fire	Create CLI to run experiments with different parameters
matplotlib	Plot the charts
networkx	Build the graph datasets
nltk	Tokenize words and detect stop words
numpy	Manipulate data arrays in a vector-fashion
scipy	Compute Spearman Correlations
seaborn	Improve matplotlib's plots look
sklearn	Generate linear models and compute TF-IDF
tqdm	Generate progress bars for long data processing pipelines

Source: The Author

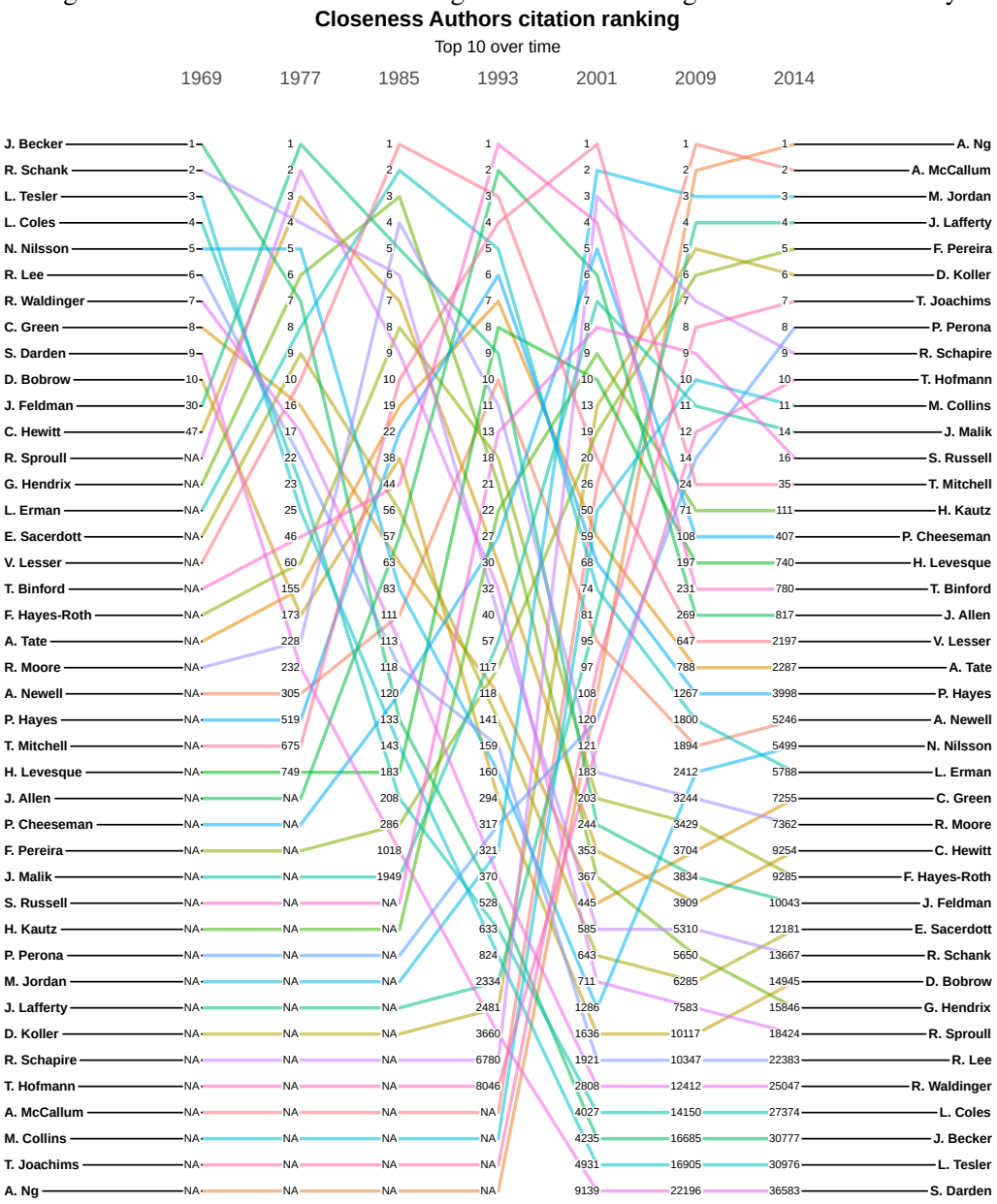
Algorithm 4 Organization Name Cleaning Preprocessing

Require: org ▷ Organization name
org ← split(org, ",") ▷ Split the text in every comma, turning it into a list
org ← org[-1] ▷ Last item in the array
org ← replace(org, "#TAB#", "") ▷ Remove unknown tag
org ← replace(org, "#tab#", "") ▷ Remove unknown tag
org ← replace(org, /[\(\)\[\]\-_]/, "") ▷ Regex-based replacement
return org

APPENDIX C — AUTHOR CITATION

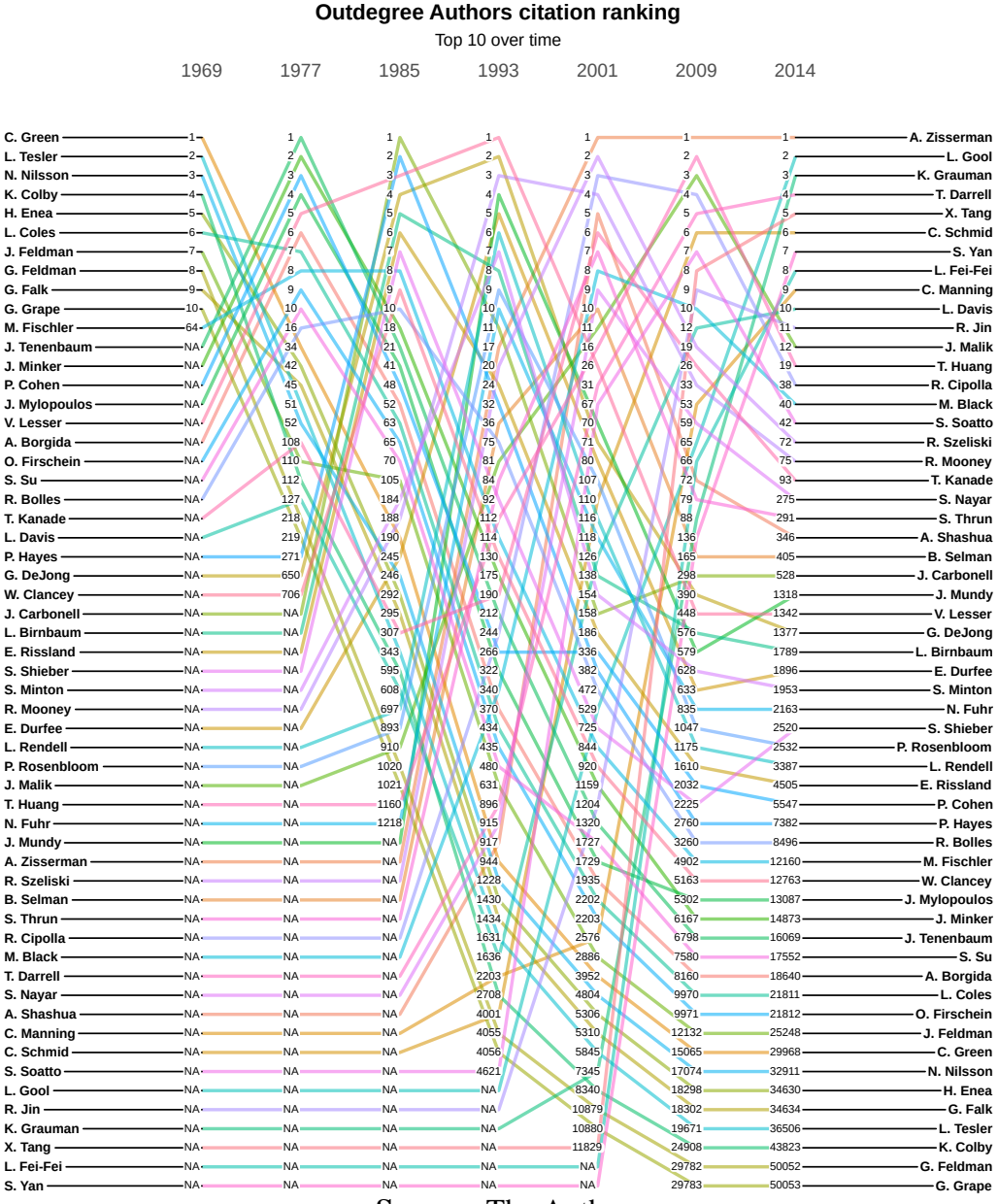
The charts presented in this chapter are related to centralities from Section 3.3.1.

Figure C.1 – Authors citation ranking over time according to Closeness centrality.



Source: The Author

Figure C.2 – Authors citation ranking over time according to Out-degree centrality

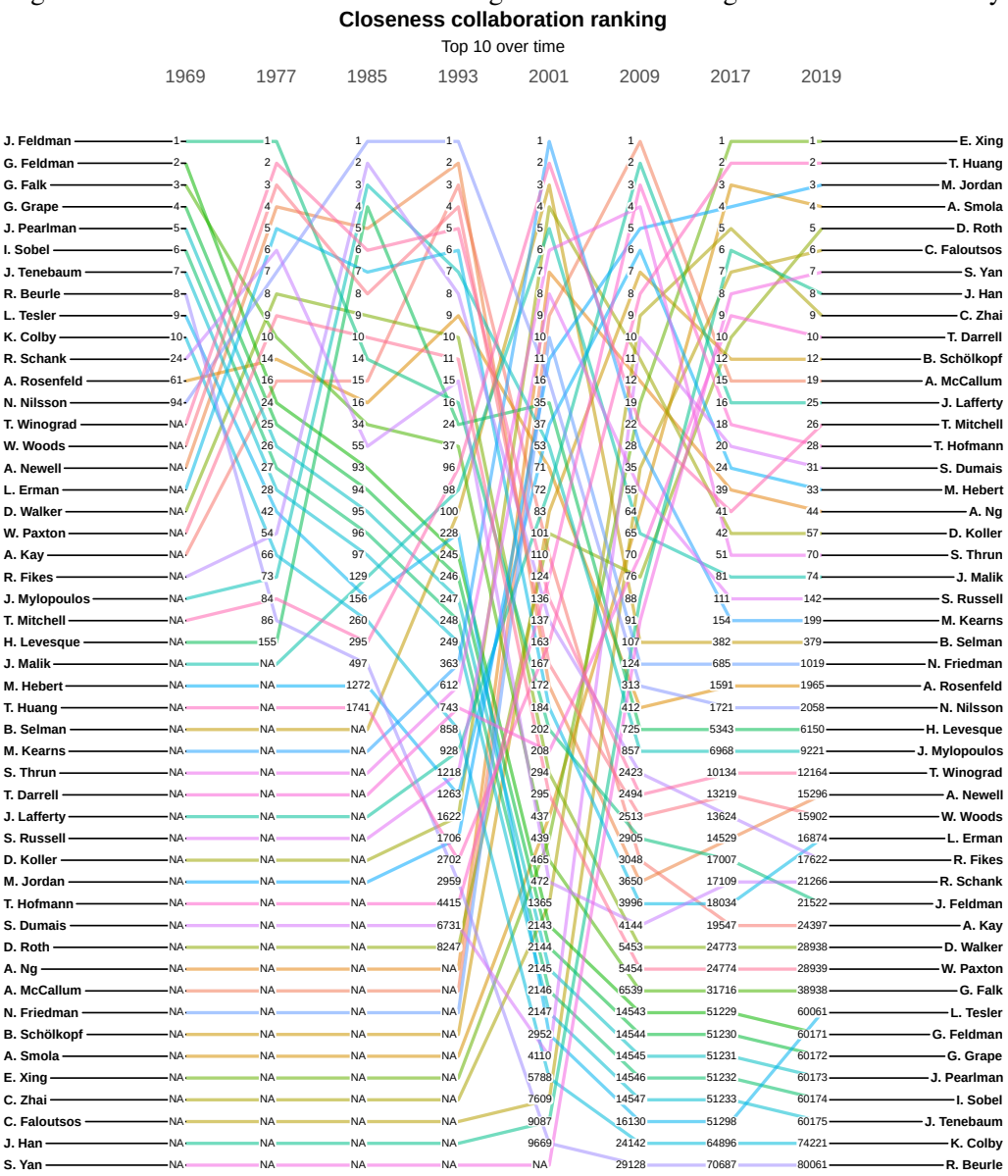


Source: The Author

APPENDIX D — AUTHOR COLLABORATION

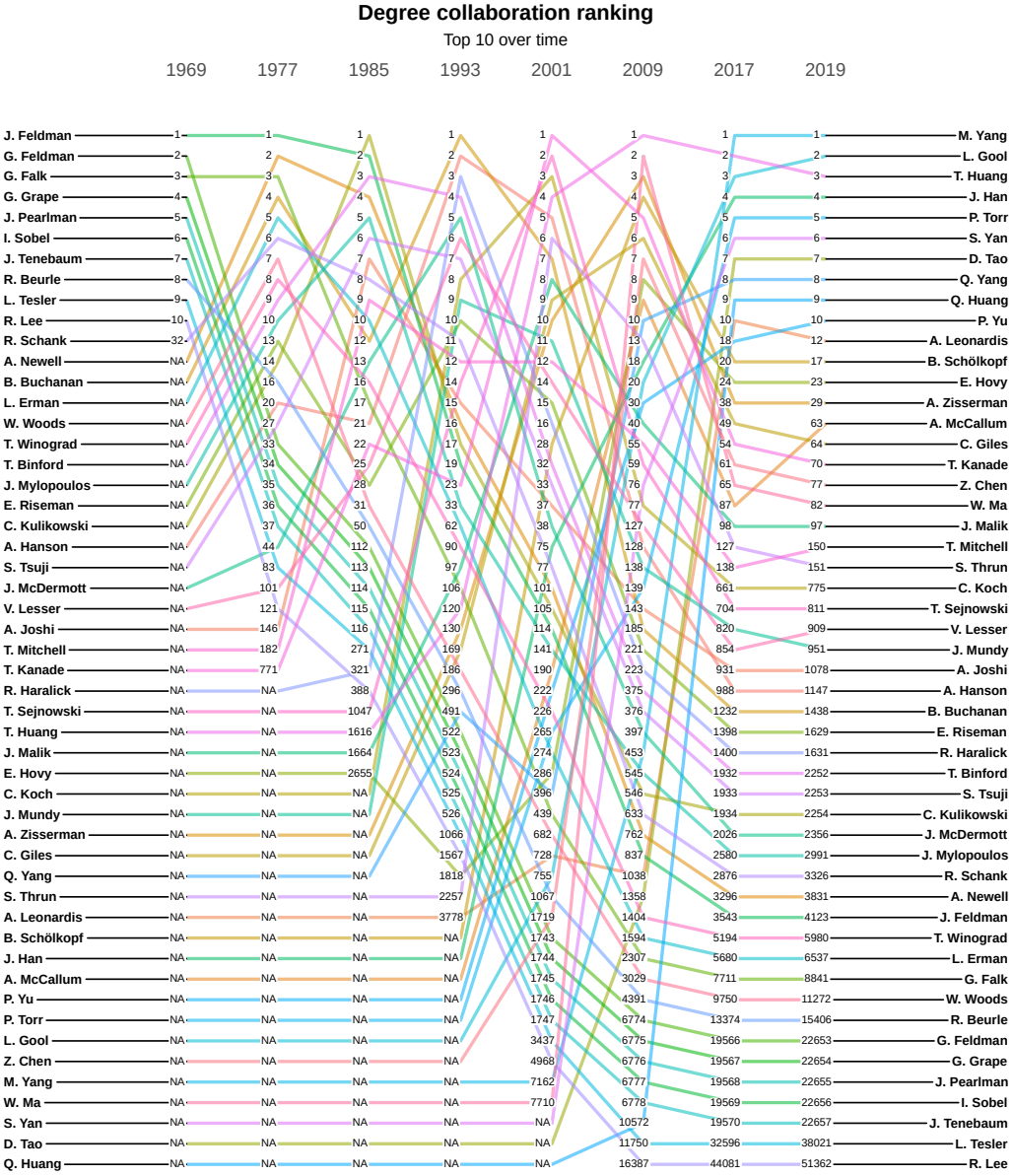
The charts presented in this chapter are related to centralities from Section 4.3.

Figure D.1 – Authors collaboration ranking over time according to Closeness centrality.



Source: The Author

Figure D.2 – Authors collaboration ranking over time according to In-Degree centrality.



Source: The Author

APPENDIX E — PAPER CITATION

The charts presented in this chapter are related to centralities from Section 4.4.

Similarly, Table E.1 is used to map every paper entry in these charts to a paper title and year.

Table E.1 – Dictionary for the papers which appeared in the Top 5 rankings

Initials	**Title**	**Year**	**Venue**
AAAOSL92	An asymptotic analysis of speedup learning	1992	ICML
AALR85	AI and legal reasoning	1985	IJCAI
AAOLTPAASP92	An analysis of learning to plan as a search problem	1992	ICML
AASTNAP69	An augmented state transition network analysis procedure	1969	IJCAI
ACDAAS90	Accurate corner detection: an analytical study	1990	ICCV
ACPFNL69	A conceptual parser for natural language	1969	IJCAI
ACRSFFL69	A contextual recognition system for formal languages	1969	IJCAI
ACSOICI86	A case study of incremental concept induction	1986	AAAI
AEARCFLG90	AUTOMATICALLY EXTRACTING AND REPRESENTING COLLOCATIONS FOR LANGUAGE GENERATION	1990	ACL
AIIFIR83	Artificial intelligence implications for information retrieval	1983	SIGIR
AIIRTWAATSV81	An iterative image registration technique with an application to stereo vision	1981	IJCAI
ALOIAEB84	A logic of implicit and explicit belief	1984	AAAI
AMAAAOAIT69	A mobius automation: an application of artificial intelligence techniques	1969	IJCAI
AMEMFPT96	A Maximum Entropy Model for Part-Of-Speech Tagging	1996	EMNLP
AMOFPSUMDCSOK75	A multi-level organization for problem solving using many, diverse, cooperating sources of knowledge	1975	IJCAI

Continued on next page

Table E.1 – *Continued from previous page*

Initials	Title	Year	Venue
ANSFSISDAR71	A net structure for semantic information storage, deducation and retrieval	1971	IJCAI
AOMOAHC75	Acquisition of moving objects and hand-eye coordination	1975	IJCAI
AOTPTPS69	Application of theorem proving to problem solving	1969	IJCAI
APFFSC01	A probabilistic framework for space carving	2001	ICCV
APFQCTSCOEG91	A Procedure for Quantitatively Comparing the Syntactic Coverage of English Grammars	1991	NAACL
ARVOTHA77	A retrospective view of the Hearsay-II architecture	1977	IJCAI
ASNAAMOHM73	Active semantic networks as a model of human memory	1973	IJCAI
AUMAFFAI73	A universal modular ACTOR formalism for artificial intelligence	1973	IJCAI
BAMFAEOMT02	Bleu: a Method for Automatic Evaluation of Machine Translation	2002	ACL
BBOWEMDAFIIR08	Beyond bags of words: effectively modeling dependence and features in information retrieval	2008	SIGIR
BTMANEFTEOVM97	Boosting the margin: A new explanation for the effectiveness of voting methods	1997	ICML
CCALMFIA11	Combining concepts and language models for information access	2011	SIGIR
CKCWBK01	Constrained K-means Clustering with Background Knowledge	2001	ICML
CLMFTC96	Context-sensitive learning methods for text categorization	1996	SIGIR
CPS84	Classification problem solving	1984	AAAI
CRFPMFSALSD01	Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data	2001	ICML
CSS73	Case structure systems	1973	IJCAI
CSTAE92	Camera Self-Calibration: Theory and Experiments	1992	ECCV

Continued on next page

Table E.1 – *Continued from previous page*

Initials	**Title**	**Year**	**Venue**
DCOEW93	DISTRIBUTIONAL CLUSTERING OF ENGLISH WORDS	1993	ACL
DRLFIR16	Deep Residual Learning for Image Recognition	2016	CVPR
DROWAPATC13	Distributed Representations of Words and Phrases and their Compositionality	2013	NIPS
EAAC89	Execution architectures and compilation	1989	IJCAI
EAFMCVE94	Efficient algorithms for minimizing cross validation error	1994	ICML
ETUOSNTP75	Expanding the utility of semantic networks through partitioning	1975	IJCAI
EWANBA96	Experiments with a new boosting algorithm	1996	ICML
EWASAFTDBOAHBS69	Experiments with a search algorithm for the data base of a human belief structure	1969	IJCAI
EWSRD13	Explicit web search result diversification	2013	SIGIR
FAATIOAIOS73	Forecasting and assessing the impact of artificial intelligence on society	1973	IJCAI
FCNFSS15	Fully convolutional networks for semantic segmentation	2015	CVPR
FEFFUDT89	Feature extraction from faces using deformable templates	1989	CVPR
FOAITHSUS77	Focus of attention in the Hearsay-II speech understanding system	1977	IJCAI
FRUE91	Face recognition using eigenfaces	1991	CVPR
GPN77	Generating project networks	1977	IJCAI
HEFANLP77	Human Engineering for Applied Natural Language Processing.	1977	IJCAI
HMME92	Hierarchical Model-Based Motion Estimation	1992	ECCV
HOOGFHD05	Histograms of oriented gradients for human detection	2005	CVPR
HTUWYK75	How to use what you know	1975	IJCAI
IAA88	Interpretation as Abduction	1988	ACL

Continued on next page

Table E.1 – *Continued from previous page*

Initials	**Title**	**Year**	**Venue**
IAFLPARBOADP90	Integrated architecture for learning, planning, and reacting based on approximating dynamic programming	1990	ICML
ICWDCNN12	ImageNet Classification with Deep Convolutional Neural Networks	2012	NIPS
IEAUWA08	Intelligent email: aiding users with AI	2008	AAAI
IFATSSP94	Irrelevant features and the subset selection problem	1994	ICML
IMAMFEMFIS69	Implicational molecules: a method for extracting meaning from input sentences	1969	IJCAI
INLG01	Instance-based natural language generation	2001	NAACL
ISARASAFD77	Information storage and retrieval: a survey and functional description	1977	SIGIR
LAAATFTOLA85	Lexical ambiguity as a touchstone for theories of language analysis	1985	IJCAI
LELASTTITP89	Lazy explanation-based learning: a solution to the intractable theory problem	1989	IJCAI
LPPKICE96	Learning procedural planning knowledge in complex environments	1996	AAAI
LRCNFVRAD15	Long-term recurrent convolutional networks for visual recognition and description	2015	CVPR
LTGCWCNN15	Learning to generate chairs with convolutional neural networks	2015	CVPR
LTRFIR10	Learning to rank for information retrieval	2010	SIGIR
LTRNLAAUA98	Learning to resolve natural language ambiguities: a unified approach	1998	AAAI
LTRWPD08	Learning to rank with partially-labeled data	2008	SIGIR
MIFRVS91	Multidimensional indexing for recognizing visual shapes	1991	CVPR
MRLUAV98	Mobile Robot Localisation Using Active Vision	1998	ECCV

Continued on next page

Table E.1 – *Continued from previous page*

Initials	Title	Year	Venue
MSCBSSFVSTN3MSAR01	Multi-view scene capture by surfel sampling: from video streams to non-rigid 3D motion, shape and reflectance	2001	ICCV
MSDIAAN71	Managing semantic data in an associative net	1971	SIGIR
NATCA18	Neural Approaches to Conversational AI	2018	SIGIR
NCFPS90	NOUN CLASSIFICATION FROM PREDICATE-ARGUMENT STRUCTURES	1990	ACL
PASTAPW69	PROW: a step toward automatic program writing	1969	IJCAI
PAUAODNPID83	PROVIDING A UNIFIED ACCOUNT OF DEFINITE NOUN PHRASES IN DISCOURSE	1983	ACL
PEOKIP71	Procedural embedding of knowledge in planner	1971	IJCAI
POACBC75	Progress on a computer based consultant	1975	IJCAI
POOFT92	Performance of optical flow techniques	1992	CVPR
PSGANL83	Phrase structure grammars and natural languages	1983	IJCAI
RAKAA77	Reasoning about knowledge and action	1977	IJCAI
RODUABCOSF01	Rapid object detection using a boosted cascade of simple features	2001	CVPR
SAATNICGWVA15	Show, Attend and Tell: Neural Image Caption Generation with Visual Attention	2015	ICML
SCASFPM08	Server characterisation and selection for personal metasearch	2008	SIGIR
SF83	Scale-space filtering	1983	IJCAI
SMIS14	Semantic Matching in Search	2014	SIGIR
STFSWA06	Semi-Supervised Training for Statistical Word Alignment	2006	ACL
TAOAITACSOKE77	The art of artificial intelligence: themes and case studies of knowledge engineering	1977	IJCAI
TAOALHWSE98	The anatomy of a large-scale hypertextual Web search engine	1998	WWW
TAVOA77	Towards automatic visual obstacle avoidance	1977	IJCAI

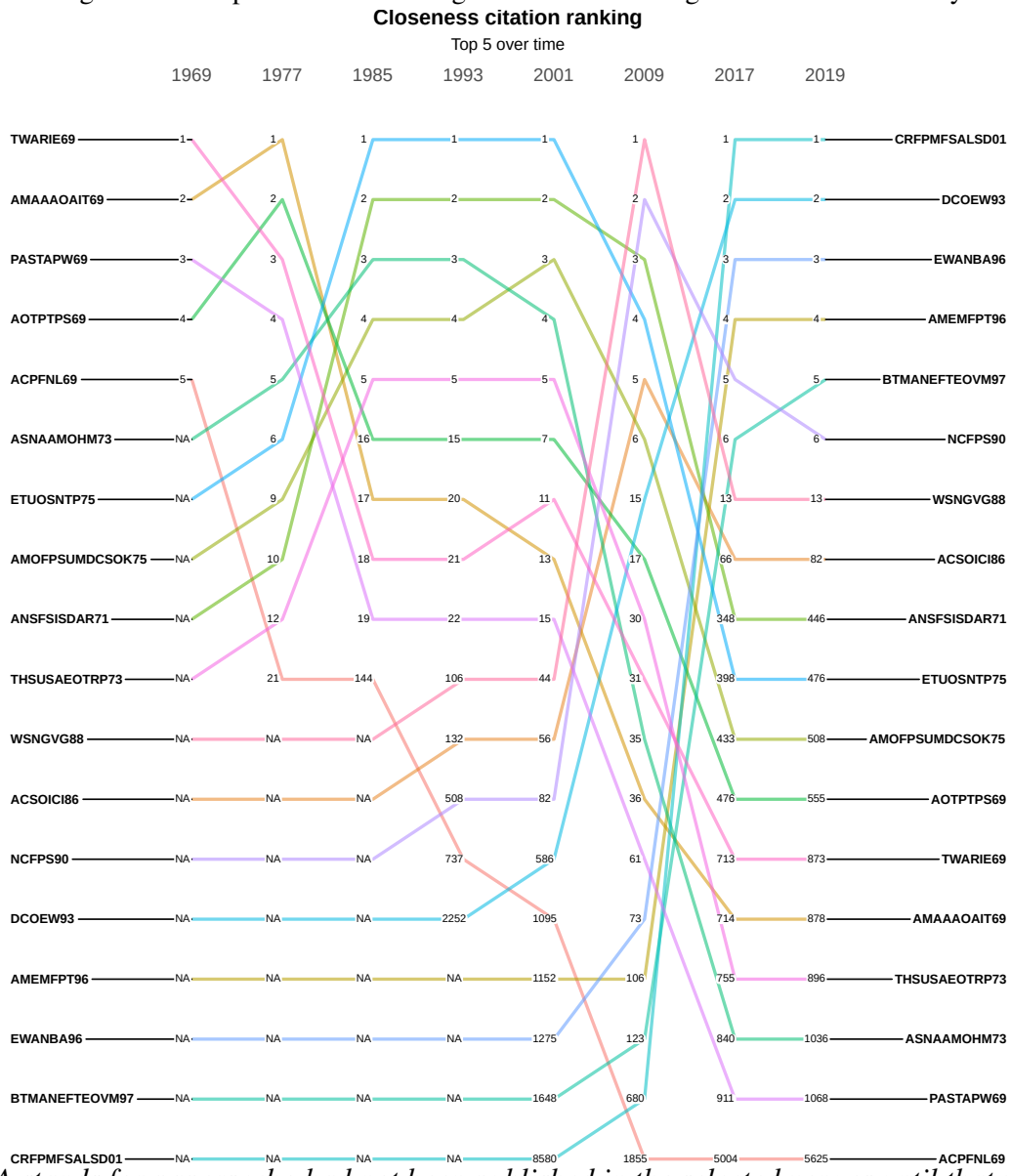
Continued on next page

Table E.1 – *Continued from previous page*

Initials	**Title**	**Year**	**Venue**
TCDAHOITP87	The classification, detection and handling of imperfect theory problems	1987	IJCAI
TCOSP89	Term clustering of syntactic phrases	1989	SIGIR
THSUSAEOTRP73	The hearsay speech understanding system: an example of the recognition process	1973	IJCAI
TMOSAAIPIASMS69	The modeling of simple analogic and inductive processes in a semantic memory system	1969	IJCAI
TSHP69	The Stanford hand-eye project	1969	IJCAI
TUGIAIE83	Tracking user goals in an information-seeking environment	1983	AAAI
TWARIE69	Talking with a robot in English	1969	IJCAI
UDTTICL93	Using decision trees to improve case-based learning	1993	ICML
VMBARR79	Visual mapping by a robot rover	1979	IJCAI
WCBSITDWAUSR92	What can be seen in three dimensions with an uncalibrated stereo rig	1992	ECCV
WSNGVG88	What Size Net Gives Valid Generalization	1988	NIPS
WYCRNTKYLIK92	What your computer really needs to know, you learned in kindergarten	1992	AAAI

Source: The Author

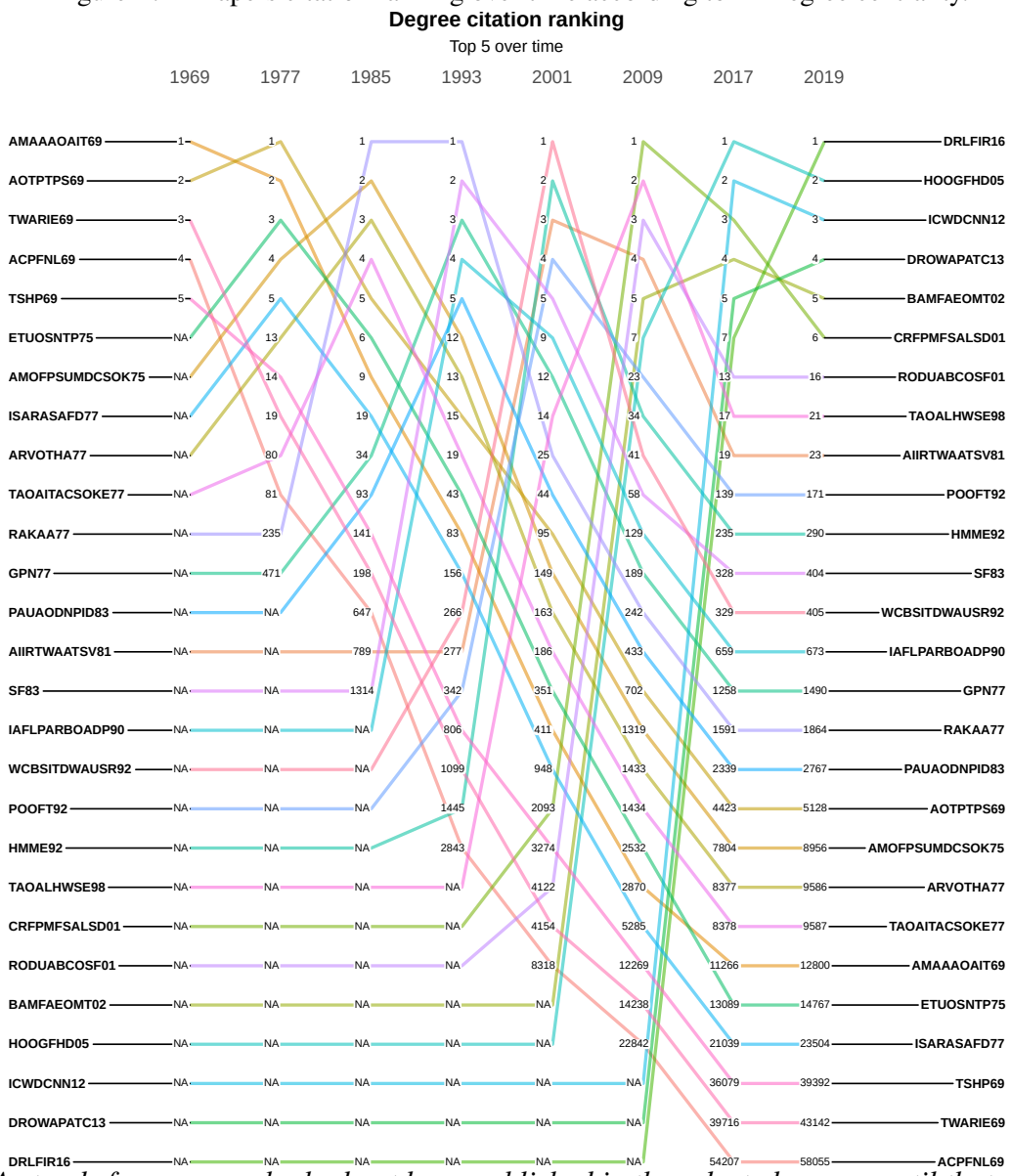
Figure E.1 – Papers citation ranking over time according to Closeness centrality.



N/A stands for papers who had not been published in the selected venues until that year. Please refer to Table E.1 in the Appendix E to see the details of each ranked paper.

Source: The Author

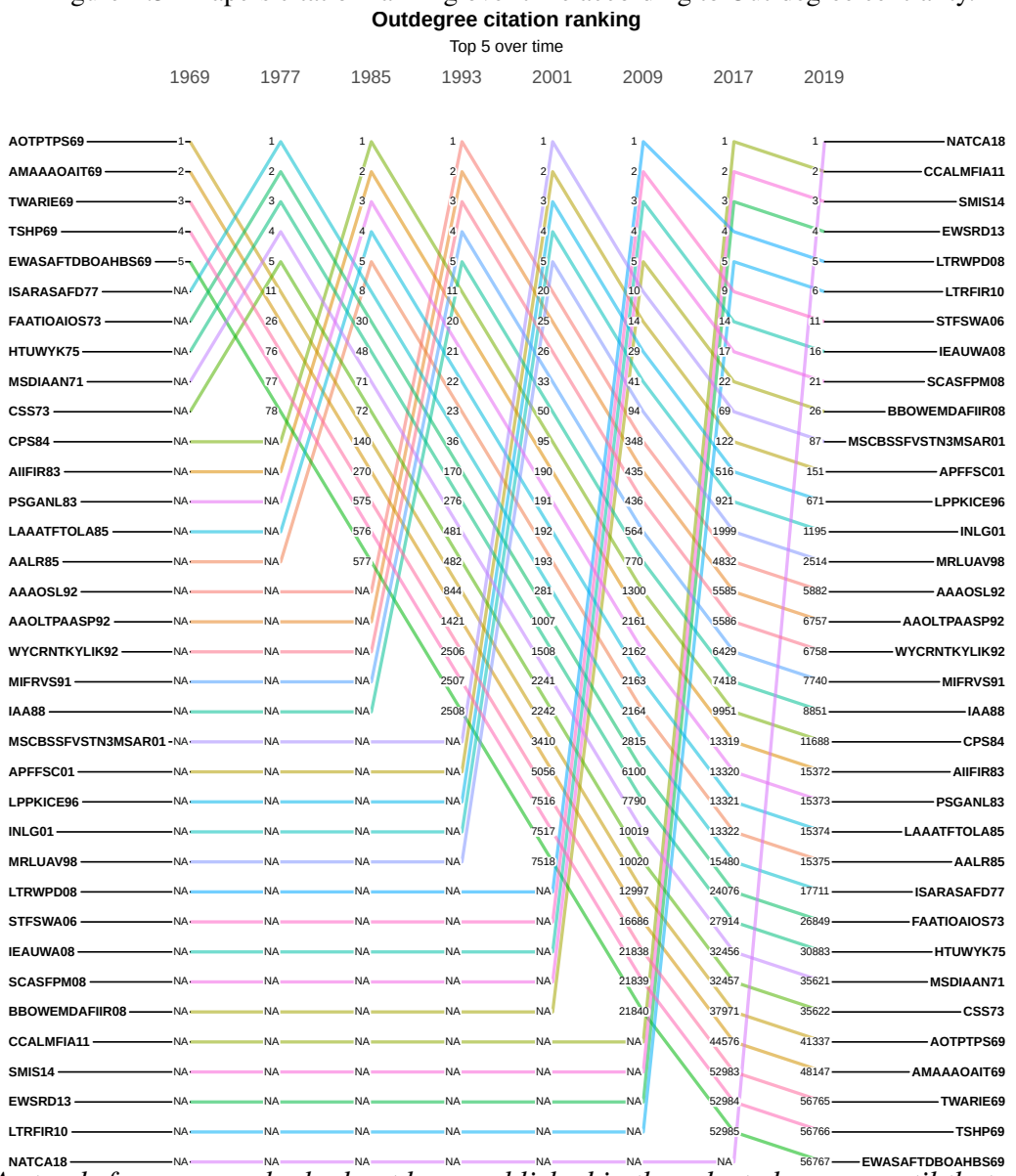
Figure E.2 – Papers citation ranking over time according to In-Degree centrality.



N/A stands for papers who had not been published in the selected venues until that year. Please refer to Table E.1 in the Appendix E to see the details of each ranked paper.

Source: The Author

Figure E.3 – Papers citation ranking over time according to Out-degree centrality.



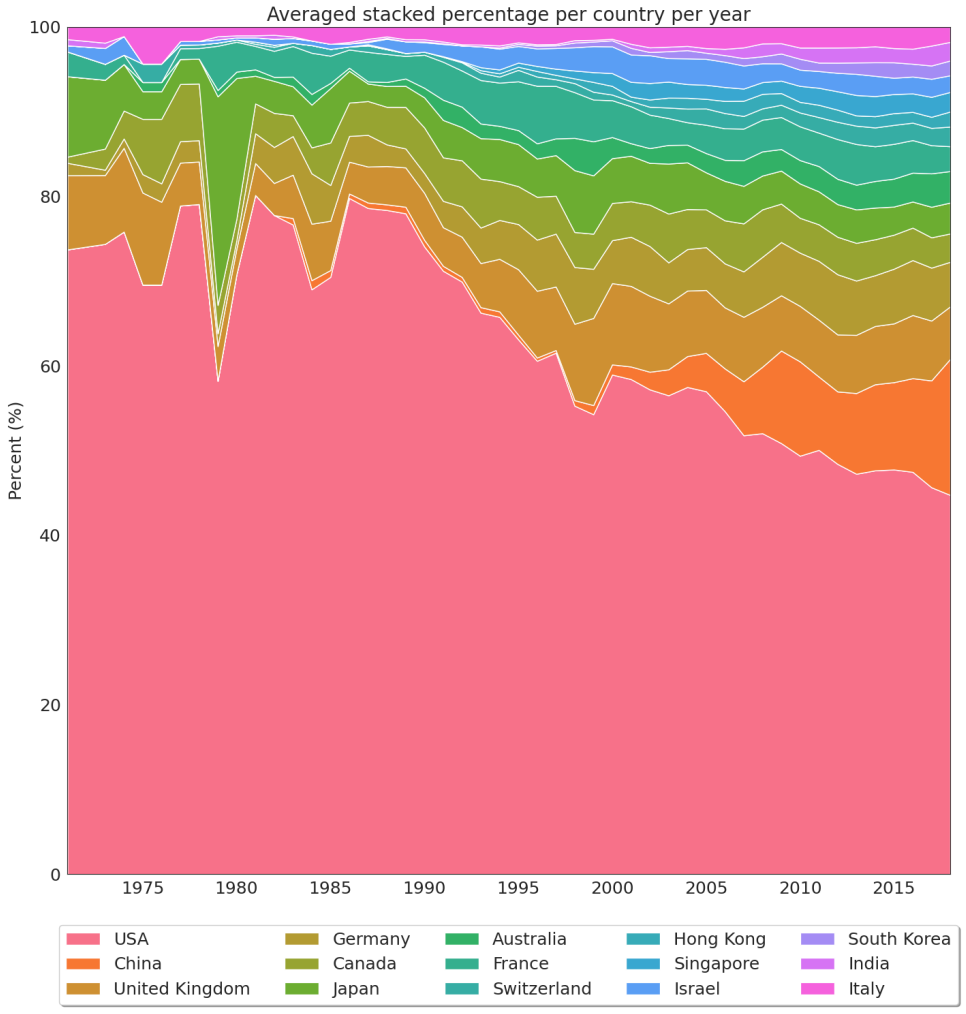
N/A stands for papers who had not been published in the selected venues until that year. Please refer to Table E.1 in the Appendix E to see the details of each ranked paper.

Source: The Author

APPENDIX F — COUNTRY CITATION GRAPH

Figure F.1 presents a different view than the one available at Section 4.6 by generating the stacked version of the countries but with a 2-year-wide sliding average window, i.e. every datapoint is actually the average between the year and its prior window, trying to avoid the variation seen in Figure 4.24 because of IJCAI being held only in odd-numbered years. The steady decline of the USA share in the graph is clearly seen.

Figure F.1 – Stacked percentage of papers viewed with a 2-years-wide sliding average window



Source: The Author

APPENDIX G — TURING AWARDS

These chapter presents some charts related to the correlation between Turing Awardees papers abstracts and the other authors abstracts words. We used a Spearman Correlation to compute these.

Table G.1 contains every single Turnig Award winner, with the year they won the prize and also their country of birth.

Table G.1 – Turing Award Winners per year

Year	Winner	Country of Birth
1966	Perlis, Alan J. *	United States
1967	Wilkes, Maurice V. *	United Kingdom
1968	Hamming, Richard W. *	United States
1969	Minsky, Marvin *	United States
1970	Wilkinson, James Hardy ("Jim") *	United Kingdom
1971	McMarthy, John *	United States
1972	Dijkstra, Edsger Wybe *	Netherlands
1973	Bachman, Charles William *	United States
1974	Knuth, Donald ("Don") Ervin	United States
1975	Newel, Allen *	United States
	Simon, Herbert ("Herb") Alexander *	United States
1976	Rabin, Michael O.	Poland
	Scott, Dana Stewart	United States
1977	Backus, John *	United States
1978	Floyd, Robert (Bob) W. *	United States
1979	Iverson, Kenneth E. ("Ken") *	Canada
1980	Hoare, C. Antony ("Tony") R.	Sri Lanka
1981	Codd, Edgar F. ("Ted") *	United Kingdom

Table G.1 – *Continued from previous page*

Year	Winner	Country of Birth
1982	Cook, Stephen Arthur	United States
1983	Ritchie, Dennis M. *	United States
	Thompson, Kenneth Lane	United States
1984	Wirth, Niklaus E.	Switzerland
1985	Karp, Richard ("Dick") Manning	United States
1986	Hopcroft, John E	United States
	Tarjan, Robert (Bob) Endre	United States
1987	Cocke, John *	United States
1988	Sutherland, Ivan	United States
1989	Kahan, William ("Velvel") Morton	Canada
1990	Corbato, Fernando J. ("Corby") *	United States
1991	Milner, Arthur John Robin Gorell ("Robin") *	United Kingdom
1992	Lampson, Butler W.	United States
1993	Hartmanis, Juris	Latvia
	Stearns, Richard ("Dick") Edwin	United States
1994	Feigenbaum, Edward A. ("Ed")	United States
	Reddy, Dabbala Rajagopal ("Raj")	India
1995	Blum, Manuel	Venezuela
1996	Pnueli, Amir *	Israel
1997	Engelbart, Douglas *	United States
1998	Gray, James ("Jim") Nicholas *	United States
1999	Brooks, Frederick ("Fred")	United States
2000	Yao, Andrew Chi-Chih	China
2001	Dahl, Ole-Johan *	Norway

Table G.1 – *Continued from previous page*

Year	Winner	Country of Birth
	Nygaard, Kristen	Norway
2002	Adleman, Leonard (Len) Max	United States
	Rivest, Ronald (Ron) Linn	United States
	Shamir, Adi	Israel
2003	Kay, Alan	United States
2004	Cerf, Vinton ("Vint") Gray	United States
	Kahn, Robert ("Bob") Elliot	United States
2005	Naur, Peter *	Denmark
2006	Allen, Frances ("Fran") Elizabeth *	United States
2007	Clarke, Edmund Melson *	United States
	Emerson, E. Allen	United States
	Sifakis, Joseph	France
2008	Liskov, Barbara	United States
2009	Thacker, Charles P. (Chuck) *	United States
2010	Valiant, Leslie Gabriel	Hungary
2011	Pearl, Judea	Israel
2012	Goldwasser, Shafi	United States
	Micali, Silvio	Italy
2013	Lamport, Leslie	United States
2014	Stonebraker, Michael	United States
2015	Diffie, Whitfield	United States
	Hellman, Martin	United States
2016	Bernes-Lee, Tim	United Kingdom
2017	Hennesy, John L.	United States

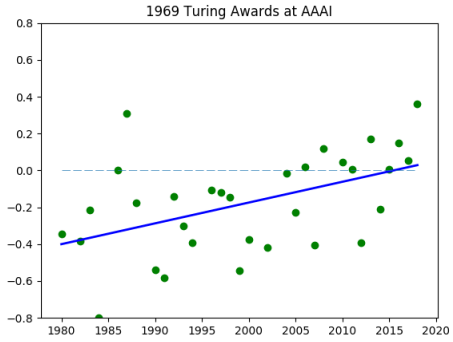
Table G.1 – *Continued from previous page*

Year	Winner	Country of Birth
	Patterson, David	United States
2018	Bengio, Yoshua	France
	Hinton, Geoffrey E.	United Kingdom
	LeCun, Yann	France
2019	Catmull, Edwin E.	United States
	Hanrahan, Patrick M.	United States
2020	Aho, Alfred Vaino	Canada
	Ullman, Jeffrey David	United States
2021	Dongarra, Jack	United States

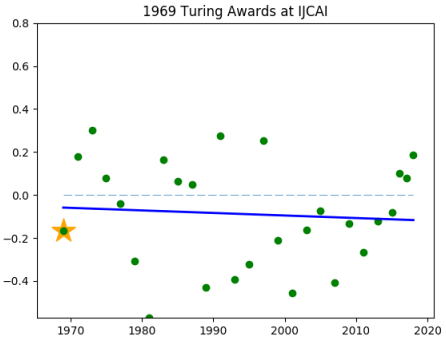
** indicates the winner is deceased*

Source: ACM Turing Award, available at <https://amturing.acm.org/byyear.cfm>

Figure G.1 – Correlation between 1969 Turing Award Winner papers and AAAI and IJCAI-published ones.



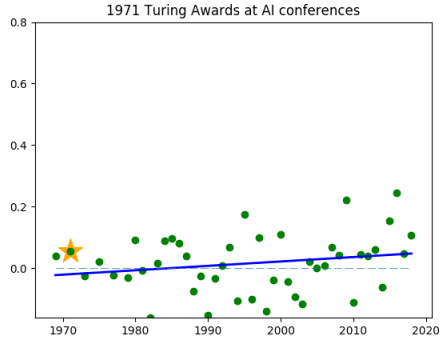
Correlation between titles of all papers published by the Turing Award winner of 1969 and all titles of papers published yearly in AAAI.



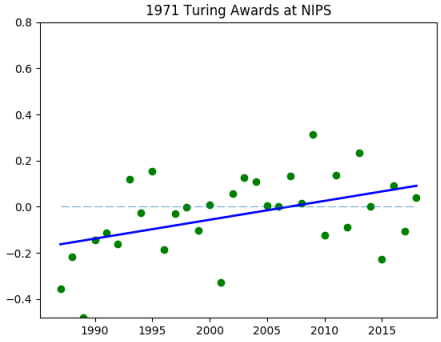
Correlation between titles of all papers published by the Turing Award winner of 1969 and all titles of papers published yearly in IJCAI.

Source: The Author

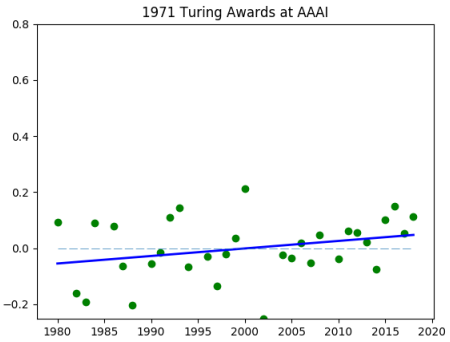
Figure G.2 – Correlation between titles of papers published by the 1971 Turing Award winner and titles of papers published in the three AI flagship conferences.



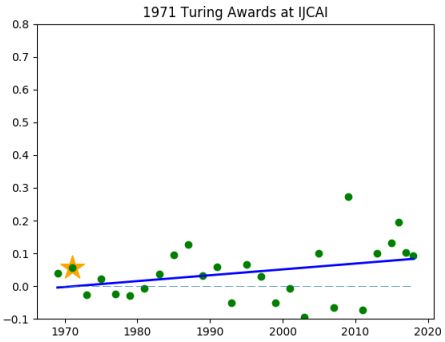
(a) Correlation between titles of all papers published by the Turing Award winner of 1971 and all titles of papers published yearly in AI conferences.



(b) Correlation between titles of all papers published by the Turing Award winner of 1971 and all titles of papers published yearly in NIPS.



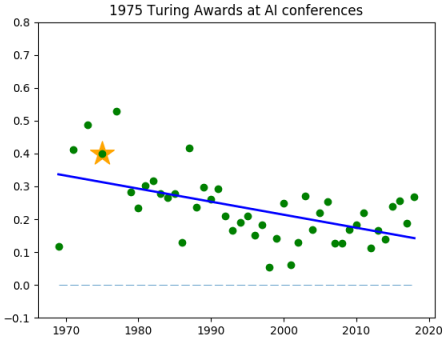
(c) Correlation between titles of all papers published by the Turing Award winner of 1971 and all titles of papers published yearly in AAAI.



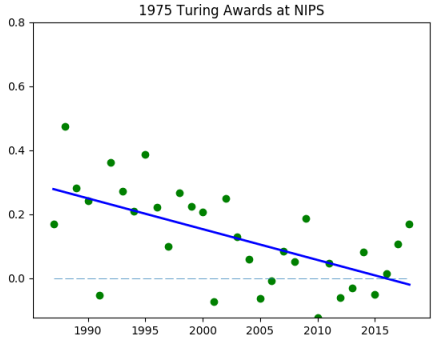
(d) Correlation between titles of all papers published by the Turing Award winner of 1971 and all titles of papers published yearly in IJCAI.

Source: The Author

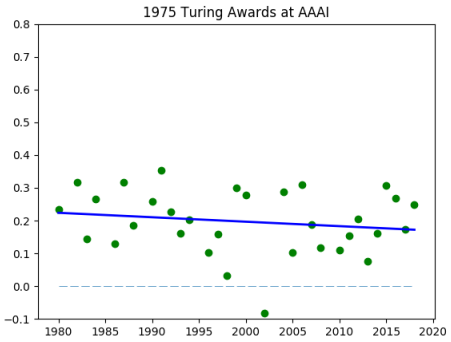
Figure G.3 – Correlation between titles of papers published by the 1975 Turing Award winners and titles of papers published in the three AI flagship conferences.



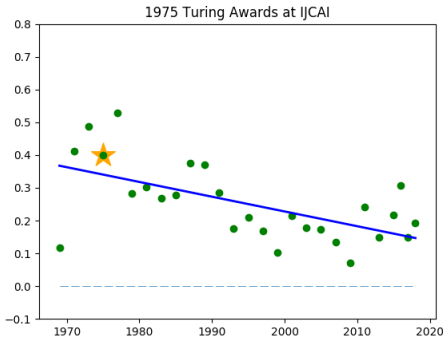
(a) Correlation between titles of all papers published by the Turing Award winner of 1975 and all titles of papers published yearly in AI conferences.



(b) Correlation between titles of all papers published by the Turing Award winner of 1975 and all titles of papers published yearly in NIPS.



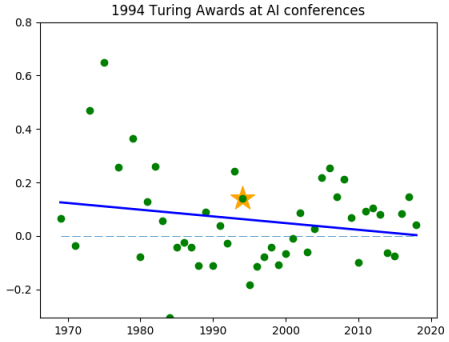
(c) Correlation between titles of all papers published by the Turing Award winner of 1975 and all titles of papers published yearly in AAAI.



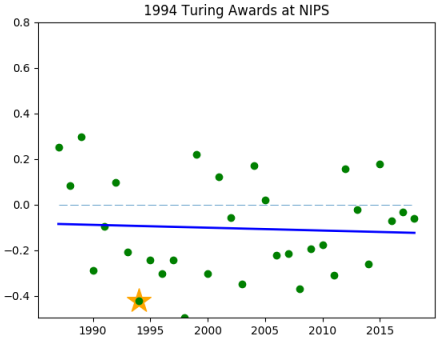
(d) Correlation between titles of all papers published by the Turing Award winner of 1975 and all titles of papers published yearly in IJCAI.

Source: The Author

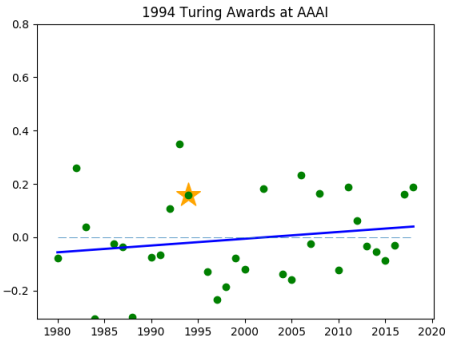
Figure G.4 – Correlation between titles of papers published by the 1994 Turing Award winners and titles of papers published in the three AI flagship conferences.



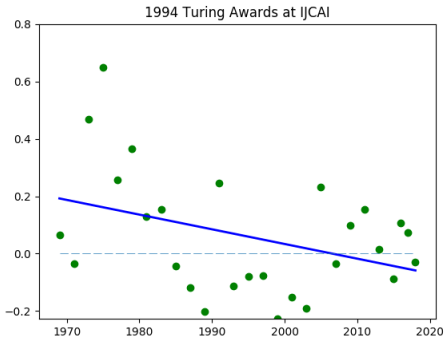
(a) Correlation between titles of all papers published by the Turing Award winner of 1994 and all titles of papers published yearly in AI conferences.



(b) Correlation between titles of all papers published by the Turing Award winner of 1994 and all titles of papers published yearly in NIPS.



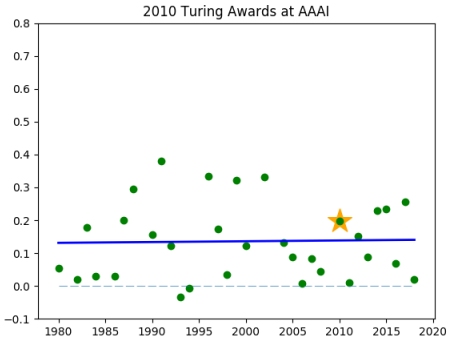
(c) Correlation between titles of all papers published by the Turing Award winner of 1994 and all titles of papers published yearly in AAAI.



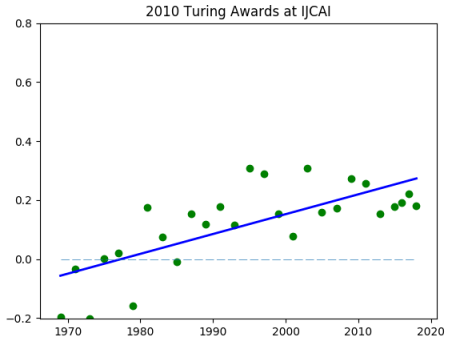
(d) Correlation between titles of all papers published by the Turing Award winner of 1994 and all titles of papers published yearly in IJCAI.

Source: The Author

Figure G.5 – Correlation between titles of papers published by the 2010 Turing Award winners and titles of papers published in AAAI and IJCAI.



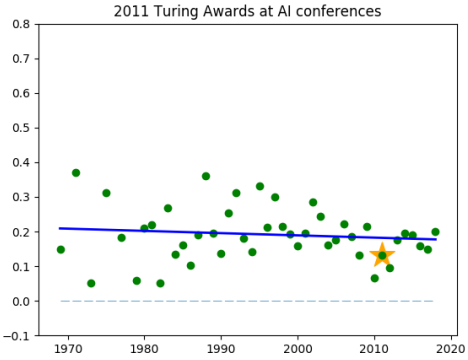
(a) Correlation between titles of all papers published by the Turing Award winner of 2010 and all titles of papers published yearly in AAAI.



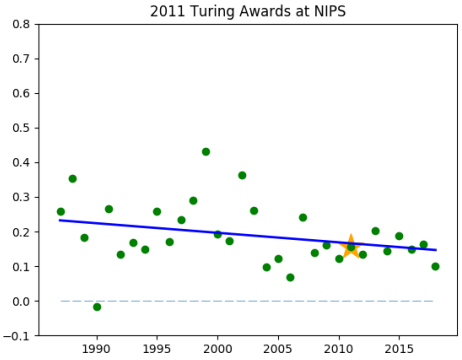
(b) Correlation between titles of all papers published by the Turing Award winner of 2010 and all titles of papers published yearly in IJCAI.

Source: The Author

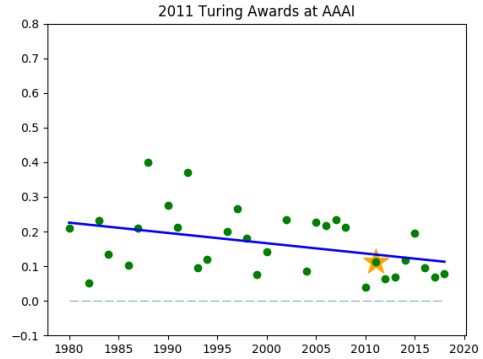
Figure G.6 – Correlation between titles of papers published by the 2011 Turing Award winner and titles of papers published in the three AI flagship conferences.



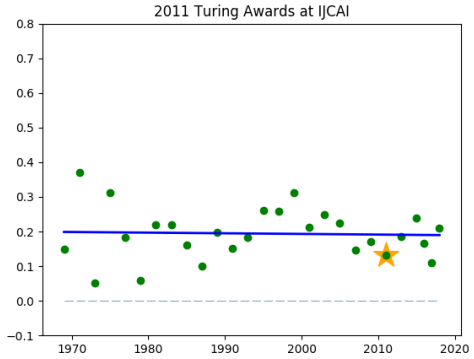
(a) Correlation between titles of all papers published by the Turing Award winner of 2011 and all titles of papers published yearly in AI conferences.



(b) Correlation between titles of all papers published by the Turing Award winner of 2011 and all titles of papers published yearly in NIPS.



(c) Correlation between titles of all papers published by the Turing Award winner of 2011 and all titles of papers published yearly in AAAI.



(a) Correlation between titles of all papers published by the Turing Award winner of 2011 and all titles of papers published yearly in IJCAI.

Source: The Author

APPENDIX H — SOFTWARE CONTRIBUTIONS

Throughout this work, we have built some interesting pieces of software that might be used by others in similarly sized tasks. They are briefly described and discussed below.

H.1 *streamxml2json* Library

When we were trying to use the original DBLP dataset (See Section 3.1 for more context) we had some trouble when trying to convert the downloaded XML file to a JSON file we could more easily manipulate. The benefits of a JSON file over the XML file go from being more human-readable to the fact of it being a bit smaller (in our case, a 3.3GB XML yields a 3GB JSON file, a 10% size reduction) – therefore, easier to load in memory.

It is a fact, however, that because we had the intention to parse this file in a CI environment, to be able to weekly generate new charts (See Section 5.3) we would need to be able to do this conversion from XML to JSON without loading the whole file into memory. After searching on Github and PyPi we realized that a tool to convert from XML to JSON without loading the whole file in memory did not exist.

That clarified, we decided we could build such a tool by using a few already existent libraries as building blocks: simplejson[1], jsonstreams[2] and xmltodict[3]. Streaming over any XML file and parsing only the necessary data, we can then output it to a JSON file, also through a file stream, without any substantial memory usage. Because our data was gzipped, the library supports reading directly from a *.xml.gz* file, not requiring the user to unzip it.

The library *streamxml2json* (AUDIBERT, 2022a) (AUDIBERT, 2022b) is available at Github in <https://github.com/rafaeelaudibert/streamxml2json> and publicly downloadable from PyPi on <https://pypi.org/project/streamxml2json/>. For sake of completeness, the library can be downloaded if you have *pip* (DEVELOPERS, 2008) installed in your machine by running *"pip install streamxml2json""*.

In the end, because we did not use this dataset, we do not use this library in our work, but the contribution was deemed important enough to the whole Python ecosystem in general so we are adding it to this section. We did keep in our main repository the file

[1] <https://github.com/simplejson/simplejson>
[2] <https://github.com/dcbaker/jsonstreams>
[3] <https://github.com/martinblech/xmltodict>

used to convert from XML to JSON, as a library usage example: <https://github.com/rafaeelaudibert/conferences_insights/blob/v11/scripts/xml2json.py>.

H.2 Python Parallel Centralities Implementation

Throughout our work, we used UFRGS HPC Group's (PCAD[4] supercomputers to be able to properly generate the graph we were building. We had to use their supercomputers because when we are computing graph centralities we need a lot of memory - for betweenness we need to store the shortest path between every single node of our graph that contains more than 100,000 nodes. Computing these centralities, however, was still pretty slow because we have to do it for every single node in every single year. An easy way to increase speed in computation, especially when you are using supercomputers, is to parallelize your job across the available physical processors. In our case, we had access to a machine with 16 cores (32 threads) allowing us to compute our results a lot faster.

Therefore, using networkx's implementations as a base, we developed a parallel Betweenness and a parallel Closeness algorithm capable of running close to 5x faster in a machine with 16 cores. The results are not 16x faster as expected because of Python's GIL which severely degrades Python's parallel performance.

The codes for these implementations can be found in <https://github.com/rafaeelaudibert/conferences_insights/blob/v11/graph_generation/parallel_betweenness.py> and <https://github.com/rafaeelaudibert/conferences_insights/blob/v11/graph_generation/parallel_closeness.py> for Betweenness and Closeness, respectively.

H.3 Graph Parsing pipeline

In our work, we had to generate several different types of graphs, with several different parameters in each of them. We also wanted to be able to easily cache data we had already computed, avoiding unnecessary computation.

To solve these problems, we devised a simple structure where we could extend a base *GenerateGraph* class (available in <https://github.com/rafaeelaudibert/conferences_insights/blob/v11/graph_generation/generate_graph.py>) that exposed several methods that made our job easier. Some of the exposed methods help us in the process of caching

[4] <http://gppd-hpc.inf.ufrgs.br/>

our data. Whenever we want to build a new graph, if we have no caching, we need to do these steps:

1. Filter papers from the required venues from dblp's JSON file
2. Generate the full graph for every year
3. After the full graph is complete, compute the centralities

If we always followed these steps, whenever we made a code change to the centralities computation, we would need to run everything before. We can easily solve this by calling some of the base class helper methods that know how to save a pre-parsed list of papers from selected venues, or even an already partial graph if we had only built it until a given year (imagine you noticed something wrong or an exception was raised after you had parsed half the dataset).

Also, to be able to control which type of graph we wanted to run from the command line, we built a CLI on top of this class using Google's fire[5] library. It is used to automatically generate a CLI from the parameters of a function, effectively allowing us to simply add a new parameter to a function and then pass the parameter value from the command line to properly pass the parameters to our code.

When we want a new type of graph, therefore, we simply extend this *GenerateGraph* class and add a new parameter to the main function, allowing us to easily call this new type of graph generation.

It is worth noting, however, that, ideally, *fire* should be replaced by *click*. Click[6] is a more maintained library, with better features: automatically generated fully-customizable help command, subcommands to avoid the extra work of manually creating flags when creating new types of graphs, proper filename handling, etc.

[5] <https://github.com/google/python-fire>
[6] <https://click.palletsprojects.com/en/8.1.x/>