The Evolution of Digital Technologies: A Network Perspective on Machine Learning

Fabian Braesemann[1]*

[1]University of Oxford, Saïd Business School, Oxford OX1 1HP, UK,
fabian.braesemann@sbs.ox.ac.uk

This is an extended abstract of a Conference Paper presented at *Complex Network 2019*, 10 – 12 December 2019, Lisbon, Portugal. Available online:

http://complexnetworks.org/

1 Introduction

The rapid technological development due to innovations in Information and Communication Technologies brings vast economic opportunities. At the same time, it might lead to major shifts in the labour market due to technology-enabled offshoring or automatisation of jobs [1]. In particular, digital technologies such as 'Machine Learning' are predicted to have a profound impact on the economy [2]. However, their constituent parts and relations to other technologies are often ill-defined [3]. It is important for technology-specific investments and retraining programs to better understand their evolution and relation to other digital technologies.

Here, we construct a network of technologies related to machine learning based on data from *Stack Overflow*, the world's largest question-and-answer website for programming questions.[1] This network reveals the changing centrality of machine learning topics, libraries, and related programming languages over time as the network links rewire when novel technologies are introduced. It thus allows for understanding the development of the field as combinatorial technological evolution [4], shaped by the replacement of older technologies by novel ones. The data can be used to test network models on innovation and novelty [5, 6], and on creative destruction [7].

2 Data and Methods

Stack Overflow provides more than 18 million questions on thousands of different programming-related topics.[2] Most of these topics refer to technologies such as *Python*, *MATLAB* or to technology domains such as *Machine Learning*.[3] Each question is assigned one or more tags. Here, I focus on all questions tagged with the label 'machine-learning'. A question is represented as a binary vector containing a one, if tag A is present and zero otherwise. The total dataset contains $N = 119,926$ questions (rows) and $T = 2793$ tags (columns) posted between 2008 and 2019.

On this dataset, we applied Association Rule Learning [8] to construct a network at yearly intervals. The associaton rule concept *lift* is used, as it provides a balanced measure of proximity between two

*ORCID-ID: 0000-0002-7671-1920

[1]It is presumed that it is feasible to represent the relations between digital technologies based on co-occurrences on the online platform.

[2]All Stack Overflow data are publicly available at https://archive.org/details/stackexchange.

[3]For simplicity, I assume that each tag refers to a technology; in its wider meaning as a 'a means to fulfill a purpose' [4]. Thus, I will use the terms 'tag' and 'technology' interchangeably.

technologies (tags) in the *Technology Space*, similar to Hidalgo et al.'s [9] proximity measure of products in the product space. Formally, the lift between two technologies A and B is their joint ocurrence probability divided by the technologies' unconditional probabilities:

$$\text{lift}_{A,B} = \frac{P(E_A \cap E_B)}{P(E_A)P(E_B)},$$

where E_A and E_B are the events that questions refer to technology A and B, respectively. A lift > 1 implies that two technologies tend to occur together. Accordingly, this is the threshold for a link to be established between two technologies (nodes) in the network.

In the resulting yearly networks, I calculate the *normalised* betweenness centrality[4] of the individual technologies as a measure of their importance.

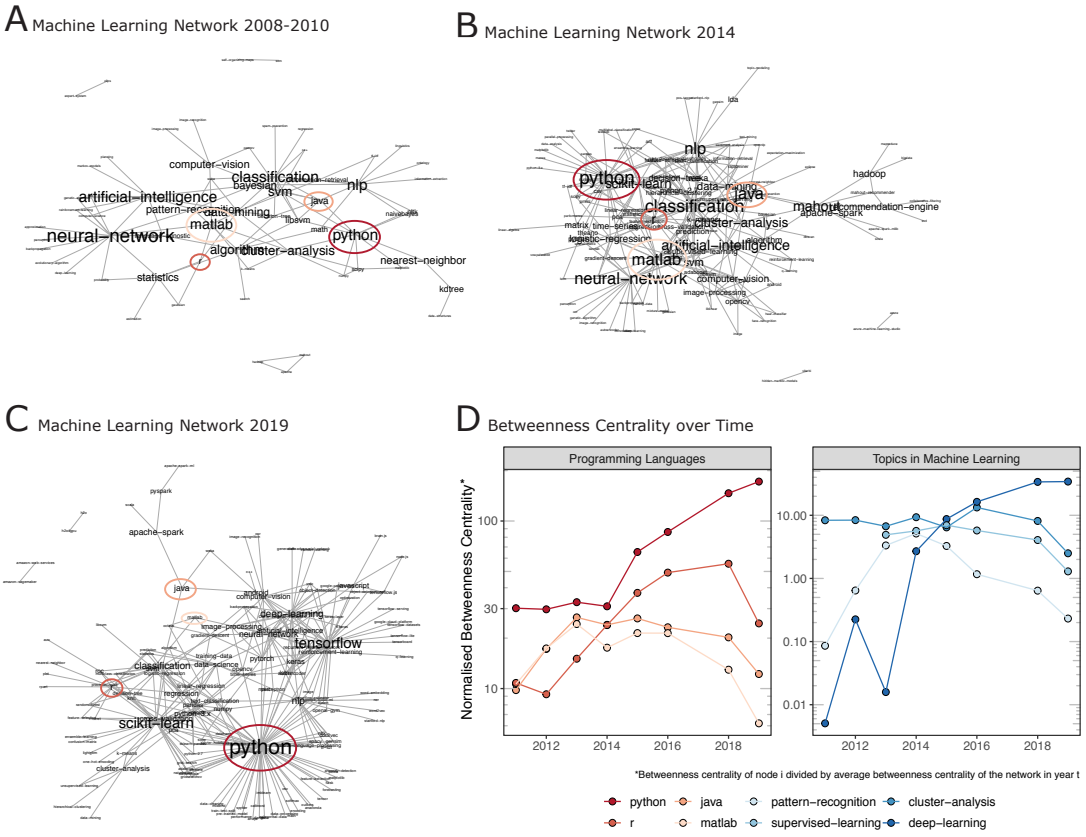

Fig. 1. (**A-C**) Networks of Stack Overflow tags related to 'Machine Learning' (ML) in 2008 – 2010, 2014, and 2019. Node size corresponds to betweenness centrality. The network became larger and denser over time as more ML-technologies are introduced. The centrality of four important programming languages, which can be used for ML, changes over time.
(**D**) Normalised betweenness centrality of four general programming languages (left panel) and four topics related to machine learning (right panel) from 2011 to 2019 (logarithmic scale). With the shifting focus from statistics to deep learning, Python's importance increased together with Python-based deep learning tools such as TensorFlow.

[4]The Betweenness centrality of the nodes is divided by the average betweenness centrality of all nodes in that year to allow comparison between networks.

3 Results

Figure 1 shows the network of technologies (tags) related to 'Machine Learning' in 2008 – 2010, 2014, and 2019 based on the Stack Overflow data. Four important programming languages (Python, R, Java, MATLAB), which can be used for machine learning applications, are highlighted by coloured circles. The early network in 2008 – 2010 is comparatively small and sparse, and the four programming languages have comparable positions in terms of their centrality. Within one decade, the set of technologies related to machine learning has changed considerably: Python has become the dominant programming language, closely related to the shift towards deep learning as a main paradigm in machine learning. Accordingly, Python benefited from the rise of Python-based deep-learning applications such as TensorFlow. The other languages have largely been displaced by Python within the domain of machine learning, due to its 'fitness' in generating productive 'offspring' technologies.

Summary. The development of digital technologies such as Machine Learning can be described empirically as a co-evolving network based on online platform data. Revealing the changing network relations is important to understand innovations in the digital sphere, as combinatorial possibilities between digital technologies are likely to be conditioned by their proximity in the *Technology Space*. The described network dataset provides a unique perspective on the technology space as it evolves in real-time. This perspective might help to better understand the geographical distribution of digital knowledge [10, 11], work [12], and innovations [13] in the digital sphere.

References

1. C. B. Frey, M. A. Osborne, The future of employment: How susceptible are jobs to computerisation?, *Technological forecasting and social change* 114, (2017) 254-280.
2. E. Brynjolfsson, T. Mitchell, What can machine learning do? Workforce implications, *Science* 358(6370), (2017) 1530-1534.
3. A. De Mauro, M. Greco, M. Grimaldi, P. Ritala, Human resources for Big Data professions: A systematic classification of job roles and required skill sets, *Information Processing Management* 54(5) (2018) 807-817.
4. B. Arthur, *The nature of technology: What it is and how it evolves*, Simon and Schuster, 2009.
5. F. Tria, V. Loreto, V. D. P. Servedio, S. H. Strogatz, The dynamics of correlated novelties, *Scientific reports* 4, (2014) 5890.
6. I. Iacopini, S. Milojevi, V. Latora, Network dynamics of innovation processes, *Physical Review Letters* 120, (2018) 048301
7. S. Thurner, P. Klimek, R. Hanel, Schumpeterian economic dynamics as a quantifiable model of evolution, *New Journal of Physics* 12(7), (2010) 075029.
8. R. Agrawal, T. Imieliski, A. Swami, Mining association rules between sets of items in large databases, in: *Acm sigmod record*, 22(2), (1993) 207-216, ACM.
9. C. A. Hidalgo, B. Klinger, A.-L. Barabási, R. Hausmann, The Product space conditions the development of nations, *Science* 317(5837), (2007) 482-487
10. F. Stephany, F. Braesemann, An exploration of Wikipedia data as a measure of regional knowledge distribution, in: *International Conference on Social Informatics*, 10540 (2017), 31-40, Springer, Cham
11. F. Braesemann, N. Stoehr, M. Graham, Global networks in collaborative programming. *Regional Studies, Regional Science*, 6(1), (2019) 371-373.
12. F. Braesemann,V. Lehdonvirta, O. Kssi, ICTs and the Urban-Rural Divide: Can Online Labour Platforms Bridge the Gap? in: *SocArxiv preprint*, 10.31235/osf.io/wbxd7, (2019)
13. F. Stephany, F. Braesemann, Coding together - coding alone: The role of trust in collaborative programming, in: *SocArxiv preprint*, 10.31235/osf.io/8rf2h, (2019)