DE GRUYTER OPEN

Journal of Official Statistics, Vol. 34, No. 4, 2018, pp. 961–979, http://dx.doi.org/10.2478/JOS-2018-0047

Megatrend and Intervention Impact Analyzer for Jobs: A Visualization Method for Labor Market Intelligence

Rain Opik[1], *Toomas Kirt*[2], *and Innar Liiv*[1]

This article presents a visual method for representing the complex labor market internal structure from the perspective of similar occupations based on shared skills; and a prototype tool for interacting with the visualization, together with an extended description of graph construction and the necessary data processing for linking multiple heterogeneous data sources. Since the labor market is not an isolated phenomenon and is constantly impacted by external trends and interventions, the presented method is designed to enable adding extra layers of external information. For instance, what is the impact of a megatrend or an intervention on the labor market? Which parts of the labor market are the most vulnerable to an approaching megatrend or planned intervention? A case study analyzing the labor market together with the megatrend of job automation and computerization is presented. The source code of the prototype is released as open source for repeatability.

Key words: Labor market; megatrends; big data; visualization; network theory.

1. Introduction

New approaches and tools are needed to understand the complex phenomena of the labor market (e.g., a mismatch between the jobs that job seekers desire or have qualifications for, and actual vacancies), and to analyze the different megatrends impacting the labor market, such as technological change, the future of professions, the automation and computerization of jobs, robots, urbanization, refugee crises, and so on. Megatrends are great forces in societal development that will impact business, economy, society, culture and individual people for the next 10–15 years (Mogensen et al. 2014). Every new megatrend creates the need for a new policy and every successful policy starts an intervention. Therefore, it is necessary to develop methods for visualizing and mapping the implications of megatrends and interventions.

Recent advances in artificial intelligence (LeCun et al. 2015) and automation have raised fears of a significant impact on the job market (Mitchell and Brynjolfsson 2017). For example, it was found that across the OECD countries, on average 9% of jobs are automatable (Arntz et al. 2016). On the other hand, this does not mean that certain jobs are disappearing completely, but rather that they are transformed into other industries and jobs requiring a different set of skills. As Lerman and Schmidt have found regarding the

[1] Tallinn University of Technology, Akadeemia 15A, 12618 Tallinn, Estonia. E-mails: rain.opik@gmail.com, and innar.liiv@gmail.com
[2] Statistics Estonia, Tatari 51, 10134 Tallinn, Estonia. E-mail: toomas.kirt@gmail.com

appearance of the first personal computers in the mid-seventies and in 1983, computer industry jobs in the United States have grown by almost 80% while total employment in the US manufacturing industry has increased by only 4% (Lerman and Schmidt 2005). Yet recent developments in technology affect too many industries simultaneously, potentially causing an accumulation of problems, as was the case with the year 2000 problem (Jones 1997), which required substantial investments to review and upgrade existing computer systems.

We consider the computerization of jobs to be one of the most important megatrends affecting the labor market and have therefore taken this as our case study to exemplify the application of the visual method proposed in this article. Our method and the prototype tool help to visualize the complex structure of the labor market and to link job demand and vacancy data to a published hypothesis on how susceptible different jobs are to computerization (Frey and Osborne 2016). The focus of this article is not on presenting new estimates of computerization, but on developing a visual method for making sense and better understanding the connectedness and the impact of megatrends on the labor market. The presented visualization method and the prototype tool itself are universal and could be used for different data sets of megatrends and interventions.

In this article, we propose a method for representing the complex internal structure of the labor market from the perspective of occupations that are similar based on shared skills. In addition, we have developed, and present here, a prototype tool, together with an extended description of its graph construction and the related necessary data processing for linking multiple heterogeneous data sources. The method is applied to a case study of visualizing the labor market along with external information (i.e., the jobs susceptible to computerization according to Frey and Osborne 2016) in order to understand the interplay between and the joint patterns in several data sets. The source code of the prototype is released as open source for repeatability at (Opik 2017b) and the prototype is available online at (Opik 2017a).

The article includes a detailed description of the steps needed to develop the visual method and implement the prototype tool. In Section 2, we describe the methods and data used, as well as the relations between the data sets. Section 3 provides the details of how we constructed the graph of occupations and how similarity between the occupations is defined. The general technical architecture of the prototype tool and the data processing pipeline is covered in Section 4. This section also discusses the visualization capabilities of the tool and how it can be used to reveal the demand and supply imbalance of occupations. In the final section, the limitations of the prototype tool are explained, followed by conclusions and directions for future research.

2. Methods and Data

The visual method for representing the complex labor market internal structure and the prototype for interacting with the data were developed using a hackathon approach. The word hackathon is combined from the words *hack* and *marathon*, where hack is used in the sense of exploratory and investigate programming (Briscoe and Mulligan 2014).

The main contributions of this article are based on an entry for the European Big Data Hackathon, organized by the European Commission and Eurostat (European Commission

2017b). Teams were gathered from all over Europe to compete for the best data product that combines official statistics and big data to support policymakers in pressing policy questions facing Europe. The policy question for the 2017 hackathon was: "How would you support the design of policies for reducing mismatch between jobs and skills at regional level in the EU through the use of data (European Commission 2017a)?" This article took a more general approach to focus on the interconnectivity of the labor market, the supply and demand in certain segments of the labor market (Weiling and Borghans 2001), and to develop a visual method for representing the complex labor market internal structure from the perspective of similar occupations based on shared skills.

The participants of a hackathon collaborate intensively over a short period of time, and the design of such events encourages and rewards creativity and innovation (Zukin and Papadantonakis 2017). Therefore, despite inherent limitations due to the short time frame, hackathon as a methodology provides feedback and validation mechanisms for ideas and results.

The European Big Data Hackathon 2017 had two independent panels of evaluators – a statistical panel and an industry panel – who were responsible for the evaluation of results presented by the competing teams. The statistical panel was composed of ten members ranging from policymakers with responsibilities in the domain of the policy question (i.e., employment and skills, big data and data economy), official statisticians and academia. The industry panel was composed of ten representatives from all the sponsors of the Hackathon (European Commission 2017d). The evaluation criteria were the same for both panels: relevance, methodological soundness, communication, innovative approach, and replicability (European Commission 2017a).

The methodological basis of the presented method is formed by graph theory (West 2001), a node-link representation (Ghoniem et al. 2005), analytic task taxonomy (Amar et al. 2005) and a value-driven framework for visualizations (Stasko 2014). The hackathon format and constant feedback from mentors and co-participants enabled the development of a visualization method that would maximize the number of low-level components of analytical activity (Amar et al. 2005), following guidelines to maximize the value of visualization (Stasko 2014). The method and the prototype tool were designed to support the following low-level components of analytical activity: clustering, finding anomalies, filtering, finding similarities and extrema.

2.1. Connecting Different Data Sets and Classification Systems

To connect all data sets, we needed to convert the US–based O*NET-SOC job classifier into the international system. For that purpose, we used an occupation classifications crosswalk table, which maps an O*NET-SOC occupation to a job in ISCO (Hardy et al. 2016). While jobs in ISCO are organized into a clearly defined set of groups according to the tasks and duties undertaken in the course of the job (International Labor Organization, 2008), the classifier does not explicitly provide a list of those tasks and duties in a machine-readable format.

O*NET-SOC is a taxonomy based on the Standard Occupational Classification 2010 (U.S. Bureau of Labor Statistics 2010), which defines a set of occupations across the working world (U.S. Department of Labor/Employment and Training Administration 2010).

ESCO (European Skills, Competences, Qualifications and Occupations) is a relatively new classification system (European Commission 2013) that provides occupational profiles that show the relationships between occupations, skills, competences and qualifications in an RDF (Resource Description Framework) format. It contains 619 ISCO and 2,950 ESCO occupations, with references for mapping an occupation in the ESCO system to a corresponding job in ISCO. In addition to organizing occupations, ESCO provides a hierarchy of skills and competences. This article has greatly benefited from the 65,814 relationships in the ESCO system, which connects skills to occupations.

2.2. *Different Data Sets*

The following heterogeneous data sources were combined for the visualization method:

- EURES CV and job vacancy data set (European Commission 2017c)
- ESCO classifier in RDF format (European Commission 2013)
- List of jobs susceptible to automation/computerization (Frey and Osborne 2016)
- Occupation classifications mapping table from Occupation classifications crosswalks – from O*NET-SOC to ISCO (Hardy et al. 2016)

The basis of this article is a list of jobs susceptible to automation/computerization (Frey and Osborne 2016), which outlines 702 occupations, classified in SOC (U.S. Bureau of Labor Statistics 2010), along with their probability of computerization in the near future.

For measuring the impact of computerization, we chose to use the EURES data set (European Commission 2017c), which provides insight into the jobs offered by employers and sought by jobseekers across Europe. The EURES data set consists of two main tables, one on anonymized curricula vitae (4.7 million lines) proposed by jobseekers and another on job vacancies (35 million lines) published by potential employers. The vacancy data set was extracted from the EURES database on December 2, 2016. As the organizers of the hackathon did not want to use all the data, the same job vacancies were aggregated. The CV data set included monthly downloaded CVs from the period of March 2015 to November 2016 and contained the data of 297,940 unique jobseekers. Records in the CV table are classified by ESCO occupation identifiers, but the vacancy table is classified by ISCO identifiers.

Figure 1 illustrates how all the data sets are connected.

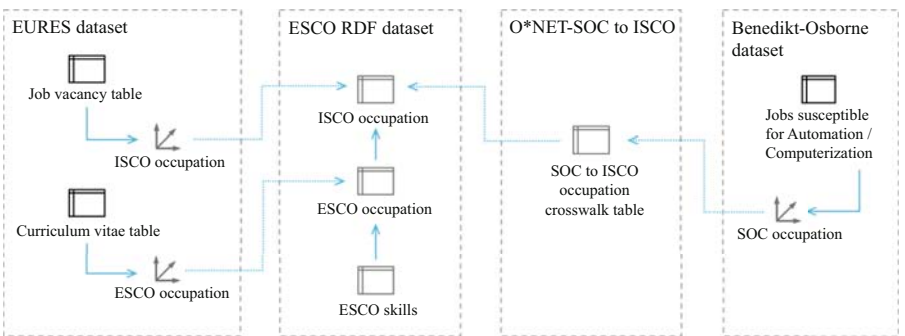

Fig. 1. Overview of data sets and their relationships.

3. Constructing a Graph

In order to visualize the complexity of a labor market, we propose to use graph theory (West 2001) to construct the node-link diagram (Ghoniem et al. 2005) in order to represent the similarity and interrelations between different occupations in the ESCO classification and hackathon data sets, according to shared skills required for a particular job. In the next subsections we will present our approach and data modelling choices for constructing the graph.

3.1. Graph of Occupations

An occupation graph is defined by two entities – node and link. A node in the graph denotes an occupation in the ESCO classifier (e.g., *bus driver*). A link is defined between two nodes (occupations) when they are similar to one another. The link in the occupation graph does not represent a match between jobs and skills, but rather a similarity of occupations based on skill information in the ESCO.

3.2. Linking Similar Occupations

The similarity between two occupations is defined as follows. For a given ESCO occupation o_1, we enumerated all the skills required for that occupation (SK_{o_1}). Then we matched all occupations that require at least one skill from SK_{o_1}, which produces a mapping from ESCO occupation o_1 to ESCO occupation o_2. We define the similarity measure as the ratio of the number of shared skills between two occupations to the number of all skills required by the first occupation (Figure 2).

To avoid ending up with a large number of skills with varying relevance, we only chose skills that were marked as *essential* for the given occupation in the ESCO classifier.

For example, let us take two occupations: *bus driver* (ESCO occupation identifier: *00cee175-1376-43fb-9f02-ba3d7a910a58*) and *private chauffeur* (*e75305db-9011-4ee0-ab62-8d41a98f807e*) and enumerate all the skills that are essential for both occupations (Table 1).

The skills in this table can be divided into three groups:

- Skill that is only required for the first occupation (e.g., *bus driver*)
- Skill that is only required for the second occupation (e.g., *private chauffeur*)
- Skill that is required for both occupations.

When we count the number of distinct skills that are required for both occupations (22 in this example) and divide it by the number of distinct skills required for the first occupation (35), we get a percentage of matching skills, which we use as a similarity measure between these two occupations.

$$similarity(o_1, o_2) = \frac{n\left(SK_{o_1} \cap SK_{o_2}\right)}{n\left(SK_{o_1}\right)}$$

Fig. 2. Occupation similarity.

Table 1. A sample of essential skills for an occupation pair.

Skills required for *bus driver*	Skills required for *private chauffeur*
provide first aid	N/A
manoeuvre bus	N/A
N/A	maintain personal hygiene standards
N/A	park vehicles
drive in urban areas	drive in urban areas
keep time accurately	keep time accurately
provide information to passengers	provide information to passengers

The resulting matrix is very large, as it contains all occupation pairs that are loosely connected by a very generic, albeit essential, skill. For example, both *bus driver* and *physiotherapy assistant* have *use different communication channels* as an essential skill, which connects them in the occupation graph. However, when we calculate the skills match ratio, we get a modest 2%. In addition, the connection between these occupations does not translate into real life, as it is difficult to imagine that a person skilled in operating heavy vehicles could easily apply for a position that requires medical skills.

Not every link in the graph is important, especially when representing the graph visually. To reduce the visual clutter, we decided to prune the graph of weakly connected occupation pairs and take only the three most similar occupations for every occupation. This has also been researched in social network analysis, where the number three has been considered sufficient to represent structurally important connections, while revealing the variation of inter-node relations across the graph and allowing efficient clustering of the graph into subgroups (Burt 1984 and Merluzzi and Burt 2013). Table 2 shows an example of the pruned graph for two selected occupations and Figure 3 contains an illustration of how the graph would look.

When this algorithm is run for all occupations (e.g., *private chauffeur*), we get new links in the graph, yielding at least three links for every node (Figure 4).

3.3. Annotating Occupations With Megatrend and Supply-Demand Data

In its simplest form, a graph node contains only one attribute, which is the title of the occupation. However, we can treat the list of nodes as a data table and attach additional attributes that explain the phenomena being investigated.

Table 2. The pruned occupation graph for two occupations.

From occupation	To occupation	Skill match
bus driver	trolley bus driver	80%
bus driver	tram driver	77%
bus driver	private chauffeur	63%
cargo vehicle driver	dangerous goods driver	60%
cargo vehicle driver	bus driver	55%
cargo vehicle driver	private chauffeur	45%

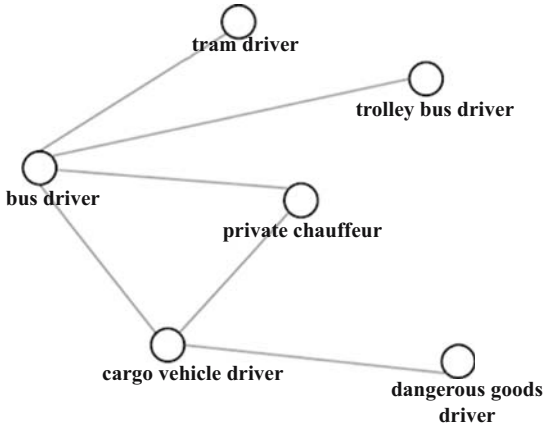

Fig. 3. The graph for two occupations.

First, we want to understand how the megatrend (automation) affects the occupation graph. The list of jobs susceptible to automation/computerization originally had SOC occupation codes. However, our occupation graph was based on occupations classified by ESCO, which can be mapped to ISCO occupation codes. We need a mapping table to link these two data sets. A mapping of ISCO to SOC (Hardy et al. 2016) is unfortunately one-to-many, which means that some ISCO occupations (e.g., *8332 – Heavy truck and lorry drivers*) are associated with several SOC occupations (*53-1031 – Driver/Sales Workers and 53-3032 – Heavy and Tractor-Trailer Truck Drivers*). As a result, they also have different probabilities for automation (0.98 and 0.79 respectively). To solve this ambiguity, we calculated two probabilities, maximum and average.

After knowing which jobs are going to be impacted, we wanted to assess how many people would be affected by this trend. Since we based our tool on the EURES CV and job vacancy data set, we could readily count the number of vacancies and the number of unique persons that have marked this occupation as their desired job. For example, based

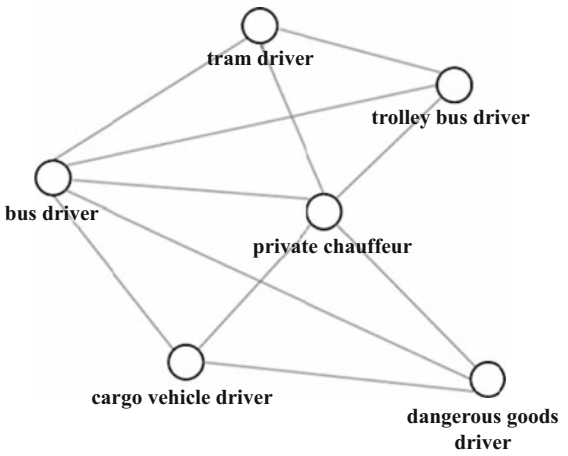

Fig. 4. A fragment of the full occupation graph.

Table 3. Node data attributes for two selected occupations.

Occupation	Prob. of automation	Vacancies total	CVs total	Vacancies in Austria	CVs in Austria	Vacancies in Belgium	CVs in Belgium	. . .
bus driver	0.89	53 936	535	1 426		1 925	5	
cargo vehicle driver	0.79	666 061	13 305	13 305		35 475	15	
. . .								

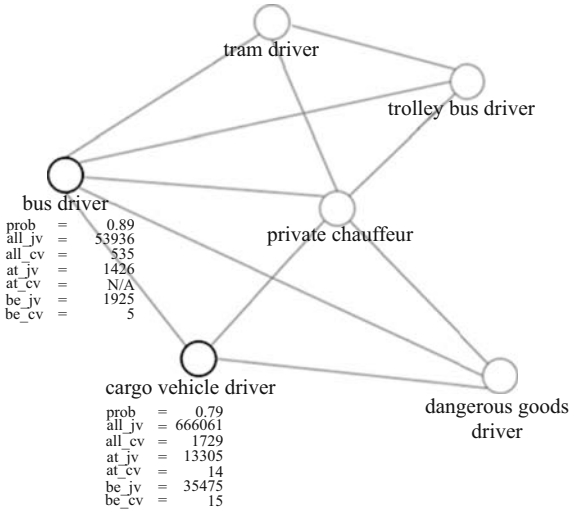

Fig. 5. An occupation graph with annotations.

on the EURES data, there are 1,925 job vacancies for *bus driver* in Belgium and five job seekers have marked *bus driver* as their desired occupation – see Table 3 for examples. Figure 5 shows the annotated nodes in a visual graph representation.

4. Visualization Tool Prototype

4.1. Technical Architecture

The majority of the integrated data sets were initially received as text files in a CSV format. After estimating the size of the main data set (EURES), which was approximately 26 million records, we decided to use Apache Hive (Thusoo et al. 2009; Huai et al. 2014) for large queries and data aggregation and PostgreSQL (Smith 2010) for more complex queries. We chose PostgreSQL as the primary database engine for data management and exploration. In such a data exploration phase, if there is a relatively small amount of data, relational databases have many benefits over specialized parallelized databases like Apache Hive. The most important advantage provided by PostgreSQL is the ease of ad-hoc querying and the expressiveness of the SQL language. For example, a simple SELECT query on a table with a couple of joins or calculating aggregates such as totals over millions of rows will be more efficient on a specialized big data backend. However, most big data query languages tend to have very limited support for more advanced operations such as subqueries or common table expressions, and the analyst is thus forced to fall back on expressing the query in a programming language. Moreover, evaluating different schema alternatives and developing a suitable data model is an inconvenient task in most big data databases, as ad-hoc schema modifications are slow and cumbersome. For that reason, we decided to perform the data exploration and schema discovery phase in PostgreSQL, and then create the final schema in Apache Hive, where we also ran the main queries for aggregating the occupation data.

The first draft of the occupation graph was drawn with the Python graph-tool (Peixoto 2014), which produced static image files. Since pre-rendered image files give a good overview of the graph, but lack in providing effective methods for filtering and obtaining details, we decided to implement the visualizer in d3.js (Bostock et al. 2011). The d3.js application can be viewed in a modern web browser without any additional dependencies.

The visualizer tool was designed to run without a server backend or online connection to a database. This makes it easy to host the tool on a static website (like GitHub) without any running costs. The final table of similar occupations and the list of all nodes in the graph were exported to text files so they can be served statically.

We used Amazon Web Services to host the infrastructure in a cloud environment. This gave us the flexibility to easily create a computational environment capable of processing the hackathon data sets and dispose of the resources after the computation is completed in order to minimize running costs. Amazon Relational Database Services (RDS) provides various SQL database engines such as MySQL or PostgreSQL, and Amazon Elastic MapReduce (EMR) facilitates running Hadoop workloads with preconfigured big data frameworks such as Apache HBase, Spark or Hive. However, due to our decision to prioritize open-source components, the backend can also be set up in an on-premise datacenter, without relying on cloud service providers.

4.2. *Data Processing*

We built the occupation graph with PostgreSQL queries. The resulting graph was stored in two denormalized tables: *node* containing a list of all occupations and their metadata, and *link* containing connections between similar occupations.

During the construction of the node table, we observed that the amount of data in the EURES data set makes direct querying inefficient – counting the number of unique job seekers and vacancies by occupations and different countries was the most time-consuming task. Since this type of workload is more suitable for databases using the MapReduce programming model, we used Apache Hive to calculate the country-based aggregates for each occupation. This resulted in a tenfold increase in query performance.

The ESCO classifier was originally presented in an RDF format, which is a list of semantic triples in the subject-predicate-object format. While specialized graph databases have support querying data in the triplet format (e.g., SPARQL or Gremlin), writing queries that join data across SQL and a graph database is very inefficient in terms of performance. Therefore, we decided to parse the RDF file and convert it to a relational structure suitable for SQL.

The serverless design of the visualizer mandates that the data files are accessible without a database. We have used flat CSV files for feeding data to the visualizer. Figure 6 shows the data processing pipeline.

4.3. *Calculating Graph Layout*

Our experience with d3.js has shown that real-time calculation of graph layouts (i.e., how to position nodes on a two-dimensional plane) may be slow for graphs with a non-trivial structure. Our occupation graph has 2,950 nodes and 8,838 links and after some experimentation we decided to pre-calculate the positions of the graph nodes. We used the

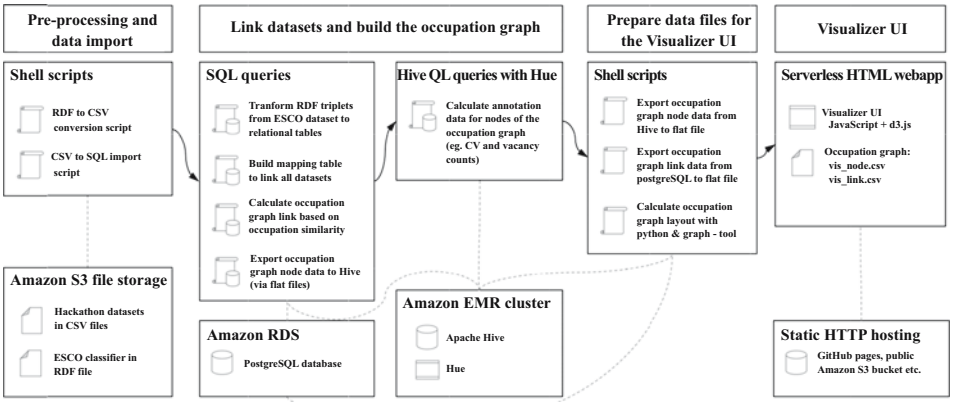

Fig. 6. The data processing pipeline.

SFDP layout algorithm (Hu 2005) from the graph-tool (Peixoto 2014) for calculating the position of the nodes and re-indexing node identifiers to a format that is suitable for a visualizer.

Besides performance gains, this also has a second benefit – the visualization can be easily shared with fellow analysts. Most graph layout algorithms are non-deterministic in nature due to random initialization and produce a different layout after each run. By using

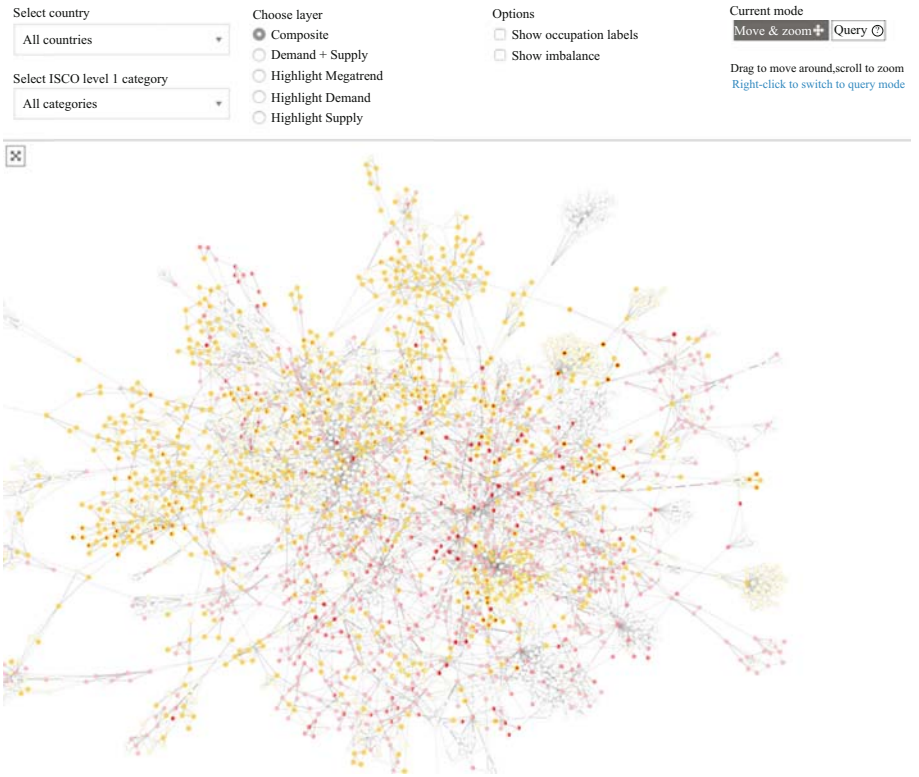

Fig. 7. The visualizer prototype.

pre-calculated node coordinates, we can ensure that the visualizer produces output that looks exactly the same in every browser given the same set of input parameters.

4.4. Visualizer UI

The user interface for the visualizer is built with d3.js, which renders a zoomable and scrollable SVG document for browsing the graph online. The prototype application (Figure 7) can be viewed in a modern web browser, preferably Google Chrome.

4.5. Prototype Interaction Models

Initially the visualizer displays the complete occupation graph. To facilitate the possibility of obtaining more detailed information, the application has two modes:

- Move and zoom mode – an analyst can click and drag the mouse to move around the graph. Scrolling the mouse wheel zooms in and out.
- Query mode – when an analyst moves the mouse cursor over a node, a small tooltip with demand and supply numbers will be displayed. Hovering also highlights connected jobs and fades out the rest of the graph. A click on the right mouse button allows for switching between Move and Query modes. See Figure 8 for an example of a query mode activated for the bus driver node.

The full occupation graph has enough nodes to appear as an impenetrable hairball when zoomed in. To reduce the clutter, we have added a filter tool to show only a relevant subset

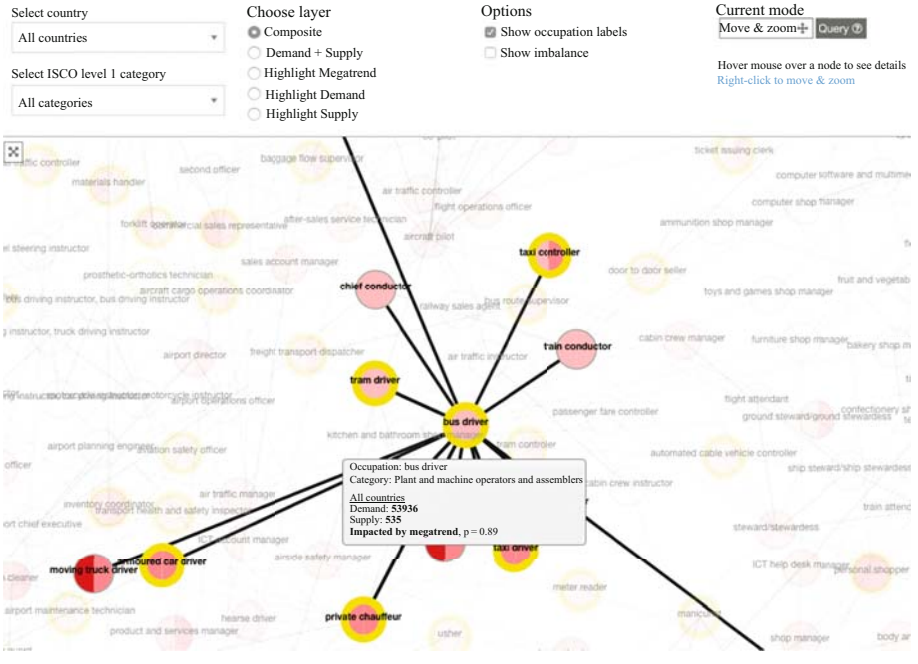

Fig. 8. A query mode is activated for a node.

of occupations. The filter tool allows an analyst to choose an ISCO level 1 occupation category (e.g., *Plant and machine operators and assemblers*) and render only these occupations that have this categorization while hiding the rest of the graph. See Figure 9 and Figure 10 for the effect of the filter tool.

The filter tool is effective due to the nature of the data set – since nodes represent ESCO occupations which can be linked to hierarchical ISCO classification, the top level of the ISCO classifier produces a meaningful subset of the graph with the same semantics.

4.6. Visualizing Node Metadata

We have used color to encode various metadata attributes that were attached to graph nodes. The visualizer supports several types for color coding – we call them layers.

- Composite layer. The left half of a node is colored by the number of vacancies available for that job (demand). Starting with white (no vacancies) to light pink (low demand) and ending with red color denoting high demand. The right half is colored by the number of job seekers who have listed a particular job in their desired job list. Color gradation is similar to the left half. Additionally, the node is marked with a yellow halo when the relevant job is affected by the Megatrend, that is, the job in the

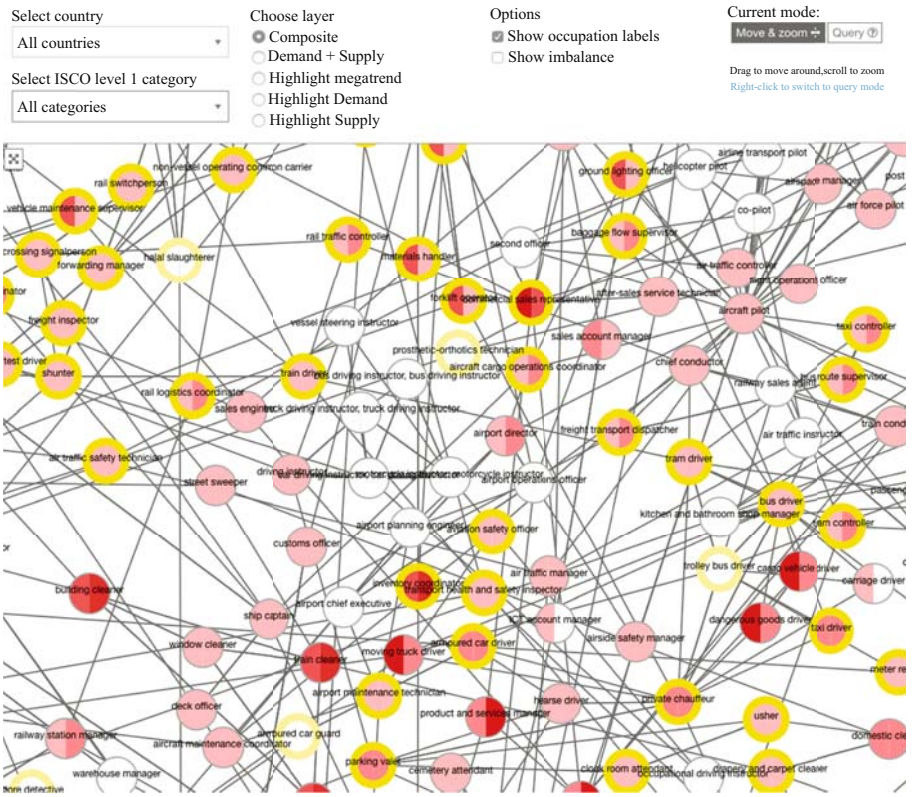

Fig. 9. A close-up of the occupation graph.

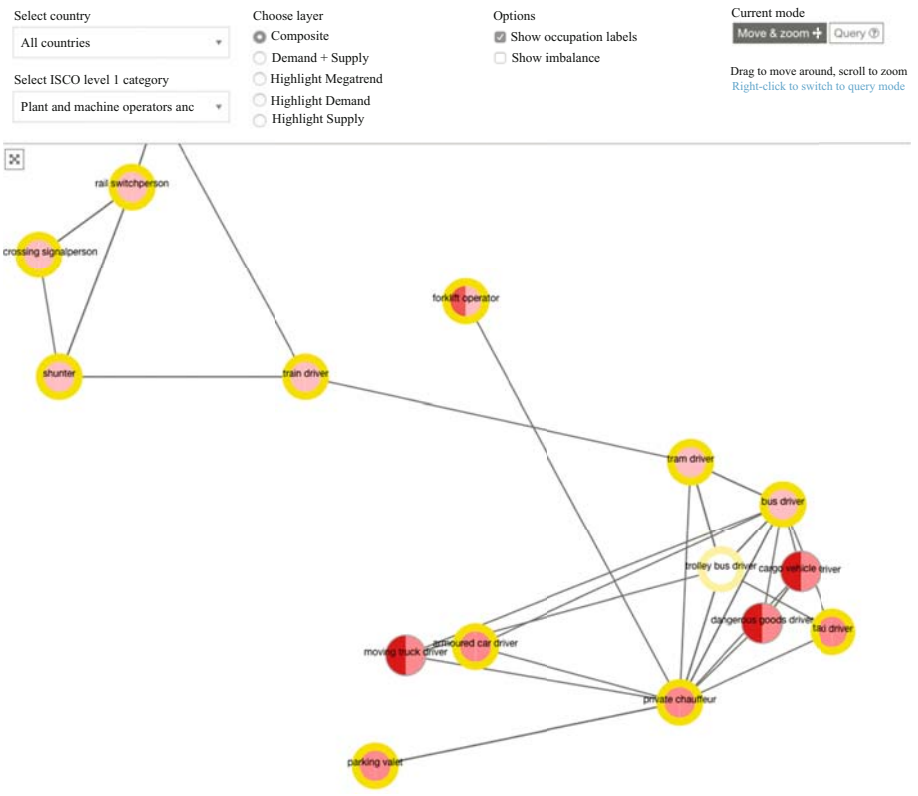

Fig. 10. The filter tool is applied.

list of jobs susceptible to automation/computerization. When the yellow halo is not present, the occupation is unaffected by the Megatrend.

- Demand and Supply layer. This is essentially the same visualization as Composite, except that the Megatrend markers (the yellow halo around the nodes) are not drawn.
- Highlight Megatrend layer. The node is colored red when the job is affected by the Megatrend. Non-affected jobs are colored white.
- Highlight Supply layer. The node is colored red when at least one job seeker has listed this job in their desired job list. White nodes denote jobs that no one desires.
- Highlight Demand layer. The node is colored red when a particular job is listed in at least one job vacancy. White nodes denote jobs with no demand.

4.7. Demand and Supply Imbalance

Color values for the left and right half (demand and supply) are normalized separately due to a huge imbalance in the EURES data. For example, some countries have no job seekers in EURES, while showing lots of vacancies and vice versa.

To overcome this issue, we implemented an alternative way for coloring nodes (refer to Subsection 4.6 for details). The default mode (i.e., *Show imbalance unchecked*) calculates the saturation ("brightness" of the red color) of the left and the right half of the node on the same scale. This helps to identify the most sought-after jobs – the analyst needs to look for

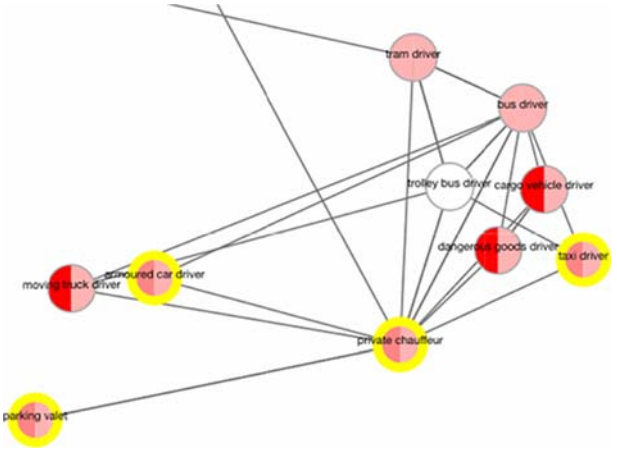

Fig. 11. Default coluring of nodes.

nodes with a bright red left half. Similarly, jobs with the largest supply of job seekers have a bright right half. For example, on Figure 11, the occupation *Private chauffeur* has a total demand (across all EU countries) of 215,677 and a total supply of 1,674. When these counts are normalized across the whole graph, both numbers are assigned the same color.

Enabling the *Show imbalance* mode normalizes both colors on the same scale. This visualizes the imbalance – when the left half of the node is a brighter red compared to the right, the job has unsatisfied demand. Conversely, a brighter right half marks jobs with an excessive number of job seekers. Figure 12 illustrates this.

Note: the EURES data contains huge discrepancies between supply and demand across different countries. Some countries have no job seekers in EURES while showing lots of vacancies and vice versa. Therefore, the *Show imbalance* mode may reveal only the extremities.

5. Limitations

Currently, the position of the graph nodes is determined by the graph layout algorithm, which generally tends to improve aesthetics by optimizing certain criteria that reduce visual clutter, for example, by minimizing the number of crossings, ensuring the even

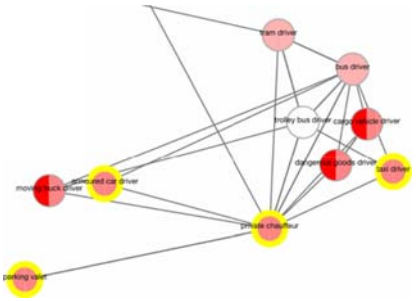

Fig. 12. Show imbalance mode activated.

distribution of nodes, and so on (Battista 1998). For the purposes of analysis, it would be beneficial to utilize the position of the nodes to encode the semantics of the underlying structure. Since the ESCO classifier contains a four-level category structure that effectively clusters the graph, we were able to use this information for the initial positioning of the nodes, providing additional visual cues to the analyst navigating the graph.

The visualizer does not currently distinguish strong links (i.e., high similarity between occupations) from weak ones. Exploring the options for visual representation of similarity (e.g., coloring intra- and inter-category links; different encoding for a link, whether it is based on cross-sector or sector-specific skills; a cut-off point for weak links) and choosing the optimal visualization remains an exercise for the future.

The tool does not, at present, provide a straightforward means of examining the common skills that were the basis for making a connection between occupations. Providing information about common skills inside the visualizer, as shown in Table 1, facilitates understanding why unusual relationships are present in the occupation graph.

6. Conclusion

Rapid changes in society in the information age can pose challenges to the national statistics offices. Registering time series might not be sufficient to face those challenges, and the need is likely to arise for different approaches, which can indicate potential future changes, as well as for using new data sources for this purpose. As novel big data sources are often heterogeneous, there are numerous steps in between to link them, many of which are not known to people entering into the space of big data.

The main contribution of this article is to provide a novel visual representation of all occupations in the labor market, which makes it possible to see similarities and patterns; and the rendering of information about job supply and demand along with external information about the trends on that same visualization. A prototype tool with the necessary data processing is proposed for interacting with the visual representation.

Our method is universal and allows for adding extra layers of information. For instance, what is the impact of a megatrend or an intervention on the labor market? Which segments of the labor market are the most vulnerable to an approaching megatrend or planned intervention?

The computerization of jobs as a megatrend was chosen as an example for using our method. What occupations are the most susceptible to computerization? Is it potentially going to impact labor market demand and skills mismatch, or further increase unemployment? These are only a few of the questions for the exploratory data analysis approach presented in this article. The real value of visualization methods and different visualizations lies in their ability to spur on and discover insights and/or insightful questions about the data (Stasko 2014).

In addition to addressing the limitations highlighted in Section 5, several interesting and different directions for future research are opened up by this work. As a lot of pre-processing of data was done manually here, a production version of such a tool should consider integrating existing data wrangling (Kandel et al. 2011) tools to optimize the time spent on introducing new data sets or scenarios. Frey and Osborne have recently published an opinion (Frey and Osborne 2018) on revisiting their seminal study (Frey and Osborne

2016), which clearly demonstrates that more research and different scenarios for automation and the future of work will be available. Research groups with different assumptions, approaches and methodologies make it ever more difficult to compare scenarios. Investigating the various options of representing several different scenarios using the visual method presented in this article could help researchers and policymakers to grasp the different results. In addition, conducting user studies with policymakers could help enhance the visual method, its interaction and the prototype tool more generally, and researchers could get valuable novel insights from policymakers about computerization or other megatrends. Finding interesting insights from data is always a dialogue and enabling policymakers to visually navigate the complex internal structure of the labor market can introduce wholly new forms of knowledge transfer.

7. References

Amar, R., J. Eagan, and J. Stasko. 2005. "Low-level components of analytic activity in information visualization." IEEE Symposium on Information Visualization, 2005. INFOVIS 2005, October 23–25 2005. 111–117. Minneapolis, MN, U.S.A. IEEE.

Arntz, M., T. Gregory, and U. Zierahn. 2016. "The Risk of Automation for Jobs in OECD Countries." *OECD Social, Employment and Migration Working Papers*. Doi: http://dx.doi.org/10.1787/5jlz9h56dvq7-en.

Battista, G.D., P. Eades, R. Tamassia, and I.G. Tollis. 1998. *Graph Drawing: Algorithms for the Visualization of Graphs*. Englewood Cliffs, NJ: Prentice Hall.

Bostock, M., V. Ogievetsky, and J. Heer. 2011. "D3: Data-Driven Documents." *IEEE transactions on visualization and computer graphics (Proc. InfoVis)*. Available at: http://vis.stanford.edu/papers/d3/ (accessed April 2017).

Briscoe, G. and C. Mulligan. 2014. "Digital Innovation": The Hackathon Phenomenon. Creative works London Working Paper. Queen Mary University of London. Available at: http://qmro.qmul.ac.uk/jspui/handle/123456789/7682 (accessed December 2017).

Burt, R.S. 1984. "Network items in the general social survey." *Social Networks* 6: 293–339. Doi: http://dx.doi.org/10.1016/0378-8733(84)90007-8.

European Commission. 2013. *ESCO – European Classification of Skills/Competences, Qualifications and Occupations – The first public release*. Luxembourg: Publications Office of the European Union. Available at: http://ec.europa.eu/social/main.jsp?catId=738&langId=en&pubId=7676 (accessed April 2017).

European Commission. 2017a. *Description of the European Big Data Hackathon*. Eurostat. Available at: https://ec.europa.eu/eurostat/cros/system/files/european_big_data_hackathon_-_description.pdf (accessed December 2017).

European Commission. 2017b. *European Big Data Hackathon*. Eurostat. Available at: https://ec.europa.eu/eurostat/cros/EU-BD-Hackathon_en. (accessed December 2017).

European Commission. 2017c. *Hackathon Data Catalogue*. Eurostat. Available at: https://ec.europa.eu/eurostat/cros/content/hackathon-data-catalogue_en (accessed December 2017).

European Commission. 2017d. *Panel of evaluators*. Eurostat. Available at: https://ec.europa.eu/eurostat/cros/content/panel-evaluators_en (accessed December 2017).

Frey, C.B. and M.A. Osborne. 2016. "The future of employment: How susceptible are jobs to computerisation?" *Technological Forecasting and Social Change* 114: 254–280. Doi: http://dx.doi.org/10.1016/j.techfore.2016.08.019.

Frey, C.B. and M.A. Osborne. 2018. *Automation and the Future of Work – Understanding the Numbers*. Available at: https://www.oxfordmartin.ox.ac.uk/opinion/view/404. (accessed April 2018).

Ghoniem, M., J.D. Fekete, and P. Castagliola. 2005. "On the readability of graphs using node-link and matrix-based representations: a controlled experiment and statistical analysis." *Information Visualization* 4(2): 114–135. Doi: http://dx.doi.org/10.1057/palgrave.ivs.9500092.

Hardy, W., D. Autor, and D. Acemoglu. 2016. "Occupation classifications crosswalks – from O*NET- SOC to ISCO." [Online]. Available at: http://ibs.org.pl/en/resources/occupation-classifications-crosswalks-from-onet-soc-to-isco/. (accessed April 2017).

Hu, Y. 2005. "Efficient, high-quality force-directed graph drawing." *Mathematica Journal* 10(1): 37–71. Available at: http://www.mathematica-journal.com/issue/v10i1/graph_draw.html (accessed October 2018).

Huai, Y., A. Chauhan, A. Gates, G. Hagleitner, E.N. Hanson, O. O'Malley, J. Pandey, Y. Yuan, R. Lee, and X. Zhang. 2014. "Major technical advancements in apache hive." In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, June 2014*: 1235–1246. New York. NY, U.S.A., ACM.

International Labour Organization. 2008. *ISCO – International Standard Classification of Occupations*. Switzerland Geneva: International Labour Office. Available at: http://www.ilo.org/public/english/bureau/stat/isco/index.htm (accessed April 2017).

Jones, C. 1997. *The Year 2000 Software Problem: Quantifying the Costs and Assessing the Consequences*. ACM Press/Addison-Wesley Publishing Co. Available at: https://dl.acm.org/citation.cfm?id=267961 (accessed December 2017).

Kandel, S., J. Heer, C. Plaisant, J. Kennedy, F. van Ham, N.H. Riche, C. Weaver, B. Lee, D. Brodbeck, and P. Buono. 2011. "Research directions in data wrangling: Visualizations and transformations for usable and credible data." *Information Visualization* 10(4) : 271–288. Doi: http://dx.doi.org/10.1177/1473871611415994.

LeCun, Y., Y. Bengio, and G. Hinton. 2015. "Deep learning." *Nature* 521(7553): 436–444. Doi: http://dx.doi.org/10.1038/nature14539.

Lerman, R.I. and S.R. Schmidt. 2005. *Trends and challenges for work in the 21st century*. Future Work, US Dept. of Labor, The Urban Institute, Washington DC. Available at: https://www.dol.gov/oasam/programs/history/herman/reports/futurework/report.htm (accessed April 2017).

Merluzzi, J. and R.S. Burt. 2013. "How many names are enough? Identifying network effects with the least set of listed contacts." *Social Networks* 35(3): 331–337. Doi: http://dx.doi.org/10.1016/j.socnet.2013.03.004.

Mitchell, T. and E. Brynjolfsson. 2017. "Track how technology is transforming work." *Nature* 544(7650): 290. DOI: http://dx.doi.org/10.1038/544290a.

Mogensen, K.A., K. Brown, A.D. Baedkel, K. Gu, M. Fert-Malka, N.T. Hemmingsen, L. Borgstrom-Hansen, C.S. Petersen, and O. Denysenko. 2014. *Trends for Tomorrow*. Member's Report 4/2014. Copenhagen Institute for Futures Studies. Available at: https://cifs.dk/publications/members-reports/ (accessed October 2018).

Opik, R. 2017a. "The prototype." Available at: https://rainopik.github.io/eubdhack-megatrend/ (accessed October 2018).

Opik, R. 2017b. "The source code of the prototype." Available at: https://github.com/rainopik/eubdhack-megatrend/ (accessed October 2018).

Peixoto, T.P. 2014. "The graph-tool python library." *figshare*. Doi: http://dx.doi.org/10.6084/m9.figshare.1164194.

Smith, G. 2010. *PostgreSQL 9.0: High Performance*. Packt Publishing Ltd.

Stasko, J. 2014. "Value-driven evaluation of visualizations." In *Proceedings of the Fifth Workshop on Beyond Time and Errors: Novel Evaluation Methods for Visualization*. (pp. 46–53). ACM.

Thusoo, A., J.S. Sarma, N. Jain, Z. Shao, P. Chakka, S. Anthony, H. Liu, P. Wyckoff, and R. Murthy. 2009. "Hive: a warehousing solution over a map-reduce framework." *Proceedings of the VLDB Endowment* 2(2): 1626–1629. Doi: http://dx.doi.org/10.14778/1687553.1687609.

West, D.B. 2001. *Introduction to Graph Theory*. New York: Prentice Hall.

Wieling, M., and L. Borghans. 2001. "Descrepancies between supply and demand and adjustment processes in the labour market." *Labour* 15(1): 33–56. Doi: http://dx.doi.org/10.1111/1467-9914.00154.

U.S. Bureau of Labor Statistics. 2010. *Standard Occupational Classification*. Washington DC: Bureau of Labor Statistics. Available at: https://www.bls.gov/soc/. (accessed April 2018).

U.S. Department of Labor/Employment and Training Administration. 2010. *The O*NET-SOC Taxonomy*. Available at: https://www.onetcenter.org/taxonomy.html (accessed December 2017).

Zukin, S. and M. Papadantonakis. 2017. "Hackathons as Co-optation Ritual: Socializing Workers and Institutionalizing Innovation in the 'New' Economy." In *Precarious Work*. (pp. 157–181). Emerald Publishing Limited.

Received June 2017
Revised April 2018
Accepted May 2018