SCODIS: Job Advert-derived Time Series for high-demand Skillset Discovery and Prediction*

Elisa Margareth Sibarani[1] and Simon Scerri[2]

[1] Fraunhofer IAIS and University of Bonn, Germany
elisa.margareth.sibarani@iais.fraunhofer.de
[2] Fraunhofer IAIS, Sankt Augustin, Germany
simon.scerri@iais.fraunhofer.de

Abstract. In this paper, we consider a dataset compiled from online job adverts for consecutive fixed periods, to identify whether repeated and automated observation of skills requested in the job market can be used to predict the relevance of skillsets and the predominance of skills in the near future. The data, consisting of co-occurring skills observed in job adverts, is used to generate a skills graph whose nodes are skills and whose edges denote the co-occurrence appearance. To better observe and interpret the evolution of this graph over a period of time, we investigate two clustering methods that can reduce the complexity of the graph. The best performing method, evaluated according to its modularity value (0.72 for the best method followed by 0.41), is then used as a basis for the SCODIS framework, which enables the discovery of in-demand skillsets based on the observation of skills clusters in a time series. The framework is used to conduct a time series forecasting experiment, resulting in the F-measures observed at 72%, which confirms that to an extent, and with enough previous observations, it is indeed possible to identify which skillsets will dominate demand for a specific sector in the short-term.

Keywords: Graph Mining · Network Clustering · Time-Evolving Network · Graph-based Time-Series Prediction.

1 Introduction

The employment sector is overwhelmed by the rapid changes witnessed in areas of expertise such as big data, data science, and artificial intelligence. A shortage of prospective candidates with relevant skillsets required for on-demand and highly-specific job positions is widely reported at the national, regional, and international levels. In 2016, the gap between supply and demand in careers contributing to the European 'Data Economy' was estimated at 420,000, with a possibility to reach 2 million by 2020[3]. To counteract this phenomenon, there is an urgent need to up-skill and re-skill the combined talent pool to meet the

* Co-funded by the European Union's Horizon 2020 research and innovation programme under the QualiChain Project, Grant Agreement No 822404.

[3] https://ec.europa.eu/digital-single-market/en/news/final-results-european-datamarket-study-measuring-size-and-trends-eu-data-economy

needs of fast-growing employment sectors. To do so effectively, training bodies, corporations, and prospective candidates alike require a dynamic overview of high-demand skillsets being requested in the job market. Early and some recent research for monitoring the labor market and analyzing skills in-demand from job adverts has concentrated on the individual skills based on its frequencies from the static point of view [2, 15, 16], neglecting both the interactions between skills and its dynamic character in nature. Complementary to these efforts, there were also few studies for tracking skills of Information Technology (IT)-related job profiles from job adverts that have been considering the trends and changes of the skills requirements over time [1, 14, 17, 18]; unfortunately, all relied solely on the 'flat' representation and information like individual skills and still do not exploit the relational behavior and network structure naturally encoded within this labor market domain. In this paper, we aim at overcoming these issues utilizing a graph-based method to represent skills in-demand in the labor market. By representing skills that are mentioned in job adverts as complex interaction (or, is called co-occurrence) networks where nodes represent skills and edges mimic the co-occurrences among them. Prior to this reported effort, we have conducted earlier studies both from the static point of view and the dynamic nature of job market networks [5, 13]. The results of this previous research were applied here in this reported study, particularly its information extraction (IE) pipeline for compiling dataset and the generic model for mining evolving networks. In summary, the contributions of this work are as follows:

- We propose a combination of graph-based labor demand model as co-occurrence networks to identify clusters of nodes that are naturally formed, with a task to characterizing each cluster's importance and evolution and use it to identify the skillset which represents the real demand, and finally tracking the changes and predicting the future skillset's importance.
- We present the Louvain algorithm [4], a state-of-the-art large network partitioning approach based on modularity optimization, since identifying clusters is paramount in this approach. In contrast to the Coulter algorithm, which we considered in previous work [13], Louvain is, by definition, better suited for the nature of our task.
- We propose several cluster categorizations: *isolated*, *secondary*, or *principal* (see Section 4.2 for detail), based on its association to other clusters. Also, we propose an additional distinction whose aim is to highlight crucial clusters with a strong ability to structure the skills network, which is called *crossroads*.
- We present a variety of indices, *centrality* and *density*, in order to equip each cluster with a presentation of their respective position and their relative stability. The *centrality* characteristic is used to see the cluster's position by the bundle of links uniting it to other clusters in the network, while *density* is to see a cluster, that is made up of words linked with each other, to define a more or less dense group, and more or less coherent and robust. We need this double analytical perspective, which allows us to give a synthetic and simplified presentation of the network, and provide a stepping-stone for dynamic analysis.

– We provide the skills demand tracking attributes by using each cluster's *centrality* and *density* values and plot it into a strategical diagram. This diagram is obtained by ordering clusters horizontally (along the x-axis), and vertically (along the y-axis), that allows us to classify all aggregates into four general categories, which correspond to the four quadrants of the graph (see Section 3), that is adapted from [8].

As a result of this, we are able to provide a comprehensive Skills Cluster Observation and Discovery (SCODIS) framework that presents a quadrant-based categorization of the important and emerging skillsets based on its density and centrality plotted into a strategic diagram. The novelty aspect lies in the whole integrated approach comprises the most fitted clustering algorithm and generates a forecasting model as well as its application to extract the labor market demand as skillsets. We begin by converting an evolving graph into static snapshot graphs at different time points and obtain clusters at each of these snapshots independently. Next, we characterize each cluster and generate a series of clusters to see its transformations, detect the stable clusters, and tracking the changes. Using this framework, we then carry out an experiment to detail a forecasting model for evolving skills networks and demonstrate their application for the task of predicting future high-demand and emerging skillsets.

2 Related Work

There has been enough interest in finding skills requirements for IT as well as non-IT related occupations from job advertisements. However, the majority of these studies [2, 15, 16] have focused on mining individual skills that are highly in-demand from a static point of view (i.e., only a snapshot of the demand from one point in time) based on its frequency of occurrence. Recently, tracking the skills requirements and its trends and changes over several snapshots of different periods have attracted the interest of several groups. Smith and Ali [1] conducted a study to assess the employable skills in programming to guide curriculum decisions. This study concluded that skill is still in demand based on the trend line that remains strong and continues to grow relative to the total number of jobs. Surakka [17] conducted a trend analysis from 1990 to 2004 to identify technical skills for the software developer by merely calculating the frequencies of different phrases. Todd et al. [14] compiled skill requirements over time by manually collecting, classifying, counting, and building an index of keywords from a pilot sample of 200 job adverts. Kennan et al. [18] studied 400 information systems (IS) job adverts over two months-period to get the skills and competencies demanded of early-career IS graduates in Australia. Although they analyze the trends over different time, the result is simply a list of skills which are ranked based on its frequency of occurrence, ignoring associations between them and unable depicting the holistic overview of the structure of the labor market. Most importantly, none of the above works address the skills cluster (or skillset) evolution tracking problem.

The problem of finding appropriate network clustering methods has been studied for some decades in many fields. Girvan-Newman [6] introduced a network clustering algorithm that implements divisive hierarchical clustering based on the breadth-first search. Xu et al. [7] proposed a structural clustering al-

gorithm for networks (SCAN) that is based on the hierarchical agglomerative method. However, despite the high effectiveness of discovering community structure in networks, unfortunately, both algorithms [6, 7] were purposely designed to process an unweighted graph. In contrast, an agglomerative hierarchical clustering algorithm specifically built for a weighted graph with breadth-first search described in [3] (henceforth referred to as the Coulter algorithm) utilizes the number of co-occurrence between pairs of vertices (such as words or noun phrases) in a corpus of text, in order to give a weight to each link. Because it always considers the strongest link every time it visits a vertex according to the breadth-first search order, unfortunately, it tends to find only the cores of clusters and disregard peripheral nodes with no substantial similarity, leading to incomplete observed network structures. Blondel et al. [4] introduced the Louvain algorithm: a heuristic method that is designed for a weighted graph and is based on modularity optimization that unfolds a complete hierarchical community structure for a network. The accuracy is excellent for the top-level hierarchy and extremely fast for networks of unknown sizes, as confirmed by a survey paper [12] that emphasizes its excellent performance and low computational complexity. Because it always visits and includes all links and vertices into the sub-graphs, therefore, the depiction of a complete structure of the network is evident. Recently, mining from dynamic graphs for tracking the clusters' evolution and its dynamic behavior has attracted the interest of several groups [9, 10, 19, 20]. However, none of these reported efforts were applied or evaluated for analyzing labor market demand as well as its changes. Lee et al. [9] proposed an incremental cluster evolution algorithm where its main contribution lies in generating the skeletal graph to summarize the information in the dynamic network in order to find the changes and track its composite evolution behaviors (i.e., merging and splitting). Another effort is by Kim et al. [20], which first clusters individual snapshots into quasi-cliques and then maps them over time using the density of bipartite graphs between quasi-cliques in adjacent snapshots. They mainly focused on handling the evolution behaviors such as birth, growth, decay, death of clusters. Asur et al. [19] focused on identifying a set of critical events that occur and influence the behavior of clusters in real dynamic networks. According to those events, communities can newly form or dissolve (i.e., start or stop) at any time as well as continue with some change (i.e., evolve). Hopcroft et al. [10] introduced a tracking evolving community approach that utilizes the centroid-based agglomerative clustering algorithm based on cosine similarity. Based on the growth rate of the cluster size (i.e., number of nodes), their goal is to detect the emerging clusters and track temporal changes in the underlying structure of a dynamic network. Although the outcomes of these works are cluster evolution behaviors in dynamic networks, unfortunately, none of those can guide users' decisions for determining the significance level of a cluster in the network where it is organized.

3 Problem Definition

Before describing our quadrant-based skills cluster observation framework in detail, we introduce the necessary notations used throughout the paper. As we explained earlier, this study focuses on the evolution of graphs, in particular, to identify the importance and significance of each cluster in the graph over time.

In order to fully understand the temporal evolution of graphs, it is critical to identify all clusters together with its density and centrality attributes and track the importance transformations undergone by the graph at different point of time along the way. In this regard, we use temporal snapshots to examine static versions of the evolving graph at different time points. Definition 1 describes a Skill Graph (or Network), more formally.

Definition 1. *A skill graph is said to be evolving if its associations vary over time. For each country a, let* $G_a = (V, E)$ *denote a temporally varying skill graph where* V *represents the total unique skills and* E *the total co-occurrence associations that exist among the skills. We define a temporal snapshot* $S_{i,a} = (V_{i,a}, E_{i,a})$ *of* G_a *to be a graph representing only skills and co-occurrences active in a particular time interval* $[T_i, T_x]$*, called the snapshot interval. In other words, a dynamic skill network of a particular country a,* G_a*, is a sequence of networks* $S_{i,a}(V_{i,a}, E_{i,a})$*, i.e.,* $G_a = \{S_{1,a}, ..., S_{t,a}, ...\}$.

To build a dynamic skill network G_a as a weighted graph, each edge has a weight attached to it, which is what we called strength of association. A simple count of co-occurrence frequency cannot sufficiently measure the strength of association between co-skills, as it favors high-frequency pairs over those with low frequency. Therefore, normalization is necessary. In SCODIS, we apply the equivalent coefficient (based on the systematic study of all such possible indices reported in [8]) as the value of strength, as follows:

$$E_{ij} = \frac{(C_{ij})^2}{(C_i \cdot C_j)}, 0 \leq s_{ij} \leq 1 \tag{1}$$

where, C_{ij} is the number of job adverts in which both skill i and j appear; C_i is the number of job adverts in which skill i appears; C_j is the number of job adverts in which skill j appears.

To investigate the graph's evolution, we need to represent its structure, by identifying clusters from each snapshot of the graph, which is explained in Definition 2. For the clustering task, we have examined the nature of several clustering algorithms thoroughly, which can be found in Section 2. We decided on the Louvain algorithm [4], to be the fittest unsupervised clustering algorithm to obtain all clusters at different timestamps. However, the alternative Coulter algorithm, which we have investigated in the same context (i.e., skills clustering) in the earlier related work [13], is considered to be the state-of-the-art and thus be used as a benchmark for the experiment of this work, see Section 5 for a detail report. The Louvain algorithm does not require a parameter specifying the number of clusters, nor a threshold to forcibly terminate its process. Consequently, the number of clusters for each snapshot may vary depending on the co-occurrence interactions in that time interval.

Definition 2. *For each snapshot from the collection of all i temporal snapshots* $S_{i,a}$ *of graph* G_a *of country a, we partitioned it into k communities or clusters indicated by* $C_i = \{C_i^1, C_i^2, ..., C_i^k\}$*. The* l^{th} *cluster of* $S_{i,a}$*,* C_i^l *is also a graph indicated by* (V_i^l, E_i^l) *where* $V_i^l \in V_{i,a}$*, and* $V_{i,a}$ *is a set of nodes in* $S_{i,a}$*. Hence, for each snapshot* $S_{i,a} = (V_{i,a}, E_{i,a})$*,* $V_i^1 \cup V_i^2 \cup ... \cup V_i^k = V_{i,a}$*. Finally,* E_i^l *indicates the edges between nodes in* V_i^l.

There are several well-known quality measures for graph clustering, such as min-max cut, normalized cut, a measure of agreement (e.g., adjusted rand index (ARI)), and modularity. Among them, min-max cut and normalized cut can only measure the quality of a binary clustering, while ARI is only possible when we have the expected clustering ahead of time. Modularity, which is introduced in [6], however, can measure the quality of clustering with multiple clusters, and when no prior clusters are known, thus, it is more suitable for our problem. The modularity Q is defined as follows:

$$Q = \sum_{s=1}^{k} \left[\frac{l_s}{L} - \left(\frac{d_s}{2L} \right)^2 \right] \tag{2}$$

where, k is the number of clusters, L is the total strength between all pairs of nodes in the graph, l_s is the total strength of a pair of nodes within a cluster s, d_s is the total strength between a node in the cluster s and any node in the graph. To enrich the information about a cluster, we identify its *density* and *centrality*, which are described in Definition 3.

Definition 3. *For a cluster s, a **density** index is a measure of the cluster's internal strength (local context), is calculated by: $s_{den} = \frac{l_s}{x}$, where, x is the number of edges that associate all pairs of nodes within a cluster s and l_s is the total strength of those edges. Whilst a **centrality** value measures the strength of a cluster's interaction with other clusters (global context), that is calculated with: $s_{cen} = \sqrt{\sum_1^n d_n^2}$, where, n is the number of edges that associate all pairs of nodes in cluster s with other nodes outside cluster s, and d_n is the strength value of those edges.*

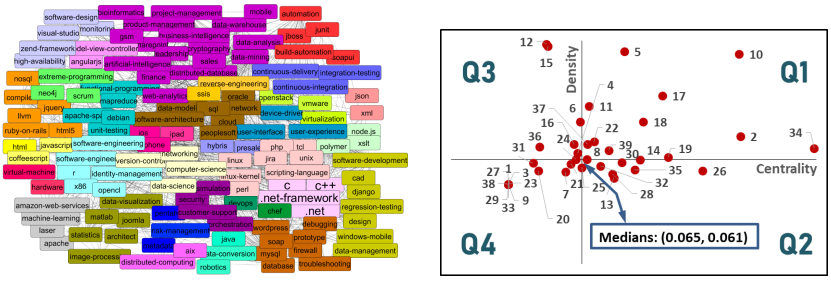
Both indices are very critical in this study. For centrality, the higher is its value, the more a cluster designates a set of skills considered crucial by the employer. On the other hand, the higher a density value is, the more the skills corresponding to the cluster constitute a coherent and integrated whole. Thus, a centrality occupies a strategical position, and density provides a good representation of the cluster's capacity to maintain itself and to develop over time. A cluster also has an additional attribute: a label or a descriptor; that is, a particular skill represents this cluster. The label's candidates must be from all nodes within but also is possible from outside that cluster; however, it is associated with the node inside that cluster. As the members of each cluster are also normalized weighted elements, thus, for each node within cluster Cl or any node in the graph that is connected with the node within cluster Cl, we calculate its weight w which was adopted from [3] as follows:

$$w_{Cl}(a) = \frac{K_{Cl}(a)}{n_{Clin} + n_{Clex}}, 0 < K_{Cl}(a) \leq n_{Clin} + n_{Clex}, 0 < w_{Cl}(a) \leq 1 \tag{3}$$

where, n_{Clin} is the total strength of a pair of nodes within cluster Cl; n_{Clex} is the total strength between a node in the cluster Cl and any node in the graph; and $K_{Cl}(a)$, if node a is within-cluster Cl, then it is the total strength between node a and nodes that are within and outside cluster Cl, else if node a is not within-cluster Cl but is connected to a node in the cluster Cl, then it is the total

strength between node a and nodes within cluster Cl. Hence the keyword with most weight serves to label cluster Cl.

Once all clusters have their density and centrality, we further trace and classify each cluster, plotted it in the form of a strategical diagram. As an example, we randomly chose the graph of country code 'at' G_{at} and depicted its temporal snapshot for period T1 (Jan–Mar 2016) $S_{T1,at}$ in Fig. 1a. In total of 145 nodes and 1029 edges, notice that this figure shows the network after the clustering process, where each color represents one cluster, and the Louvain algorithm identifies 39 clusters.



(a) (b)

Fig. 1: (a) Temporal snapshot period T1 (Jan–Mar 2016) in country code 'at', and (b) Temporal snapshot period T1 country code 'at' Centrality and Density.

Plots of centrality and density for all clusters $\{C^1_{T1}, C^2_{T1}, ..., C^{39}_{T1}\}$ being identified from $S_{T1,at}$ is shown in Fig. 1b. The origin of this diagram is the median of the represent axis values (the horizontal axis represents centrality; the vertical axis represents density). This operation allows us to classify all aggregates into four general groups, which correspond to the four quadrants of the diagram: (1) *Q1*, a quadrant that contains the *core* of the network because it strongly connects to other clusters and has strong internal links; (2) *Q2*, comprises *emerging* clusters that have already shown its importance and their growing maturity although the density of their internal links is relatively low; (3) *Q3* contains *marginalized* clusters with strong internal links yet gradually (if not progressively) become less and less interest due to its low centrality index; and (4) *Q4* comprises clusters that are both peripheral and little developed, represents the margins of the network and might gradually *disappear* from the network.

The most interesting clusters are the ones with both strong density and strong centrality (which is classified as *Q1*) and clusters with strong centrality but less density (is classified as *Q2*). From Fig. 1b, 12 clusters are categorized in Q1, and among them are Clusters-10, -17, and -18, which has a slightly balance high values of both indices, together with 8 clusters which are classified in Q2. On the contrary, Cluster-12 and -15 show strong density but weak centrality values. These attributes play a vital role if and only if it is combined with the cluster classifications (described in Section 4.2, depicted in Fig. 2). Together it will provide the basis for tracking importance (or demand) evolution of clusters from different snapshots under study. Note that when two clusters are close to one another in the strategic diagram, it does not mean that they are closely linked

to each other. The only conclusion that can be drawn from this observation is that their indices of centrality and density have neighboring values.

Algorithm 1 shows the outline of the SCODIS framework we propose. We design a quadrant-based strategy to mine each snapshot of the graph by identifying clusters and its density and centrality transformations over time, referred to as quadrant locations, along with its cluster classifications. These quadrants are then used to generate a forecasting model, to predict high-demanded skillsets.

Algorithm 1 Mine-demands (G_a, T)

Input: Skills graph $G_a = (V, E)$ and T, the number of intervals
Convert graph $G_a = (V, E)$ into T temporal snapshots $S = \{S_1, S_2, ..., S_T\}$.
for $i = 1$ to T **do**
  $C_i \leftarrow$ Cluster (S_i) # $C_i = \{C_i^1, C_i^2, ..., C_i^k\}$
  for each cluster C_i^k in C_i **do**
    $C_i^k \leftarrow$ Define-DensityCentralityLabel (C_i^k)
  end for
  for each cluster C_i^k in C_i **do**
    $C_i^k \leftarrow$ Set-quadrant (C_i^k, MEDIAN (centrality,density) C_i)
    $C_i^k \leftarrow$ Set-classification (C_i^k) [Section 4.2]
  end for
end for
for $i = 1$ to $T - 1$ **do**
  Series = Generate-series (S_i, S_{i+1}) [Section 4.3]
end for
Implement-forecasting (*Series*) [Section 5]

4 The SCODIS Framework

This section contains a comprehensive overview of the Skills Cluster Observation and Discovery (SCODIS) framework, in which we introduce and afford a formal definition to certain cluster classifications and the skillset series that is being identified in evolving graphs. The cluster classifications described in this section are inspired by a similar notion described by Callon et al. [8], in the context of tracking and visualizing clusters.

4.1 Dataset

The dataset used by SCODIS is compiled using a method we have implemented in earlier work [5], but covers a longer period of time (2016-2017) [13]. It is derived from 620,760 job adverts collected from various online job portals, namely Adzuna[4], Indeed[5], and Trovit[6]. The adverts are mostly related to 'Data Science' and 'Data Analysis' and hail from 17 European countries, and were indexed by a total of 1,287,994 skill keywords (a mean of 2.07 per job advert). The OBIE skills extraction process (fully described in [5]) is guided by the *Skills and Recruitment Ontology*[7] (SARO) [11], and utilizes a de-duplication framework to remove multiple entries (posted on multiple portals) and to minimize noise and improve accuracy. The resulting data is transformed to comply with the Resource Description Framework (RDF) W3C standard for data representation[8]. All job

[4] https://www.adzuna.com/
[5] https://de.indeed.com/?r=us
[6] https://de.trovit.com/
[7] https://elisasibarani.github.io/SARO/
[8] https://www.w3.org/RDF/

ad-related information (including skills) are stored as RDF instances, and consist primarily of `JobPosting` and `Skill` instances adhering to the descriptions in the SARO ontology.

We then used the resulted RDF-based job adverts descriptions to generate a skill co-occurrence graph, where each skill is represented as a node, and an edge between two skills corresponds to a joint demand of these two skills mentioned by the employer. Given a graph spanning for two years (2016-2017), we chose the snapshot interval to be a three-months, resulting in 8 consecutive snapshot graphs, for each 17 EU countries. Although this should generate 136 snapshot graphs, due to missing data for some countries, the result is 92 graphs having a total of 14,017 nodes and 122,697 edges. These graphs are then clustered and analyzed to identify the importance, quadrants, and transformations.

4.2 Cluster Classification

The primary benefit of the classification task that we implement for each cluster of a particular snapshot is to group them accordingly to distinguish their importance. Moreover, this task is considered to be crucial in the SCODIS framework to prevent the cluster map resulted from the Louvain algorithm being too cluttered for large networks. Thus, the aim is to trim out all clusters with low-degree node and low connectivity from the evolution tracking and prediction task. It also ensures that only potential clusters are included for the next crucial task, i.e., establish the series of the cluster of skills. For initial distinction, each cluster can be grouped based on three categories, which are as follows:

1. Isolated: a cluster C_i^k of snapshot S_i is *isolated* if all nodes within cluster C_i^k does not associate with any nodes from other clusters of snapshot S_i.

2. Principal and Secondary: a cluster C_i^k of snapshot S_i is a *principal*, if cluster C_i^k has nodes within it which are connected with the node of another cluster C_i^x of snapshot S_i with the strength values higher than the minimum strength of all pair of nodes within cluster C_i^k. This condition also makes cluster C_i^x as a *secondary* to a *principal* cluster C_i^k. The number of connections that satisfy this condition must be at least δ, a predefined link threshold.

Thus based on the above distinct categories, we can confidently decide to omit all *isolated clusters* and instead focus more on the *principal clusters*, which in some sense are the core of a given network, and so are the best candidates to be put under consideration for further processes. To simplify the analysis of characterizing cluster's content and following their evolution, we propose an additional distinction, aiming to highlight clusters with a strong ability to structure the general network, which is called a *crossroads cluster*. As a result, any analysis must therefore start with the *principal clusters* – and in particular *crossroads clusters*.

3. Crossroads: is a *principal cluster* which has y secondary clusters, where y has a value greater than α, a predefined secondary threshold. By their power to connect, *crossroads clusters* play an essential role in the transformation of a network. Setting the threshold at $\delta = 3$ qualifying links for *principal cluster* classification, the classification result for snapshot $S_{T1,at}$ of graph G_{at} depicted in Fig. 1a is shown in Fig. 2, with each node, represents a skills cluster and the legend provides the cluster's label. Some observations include: Cluster-13

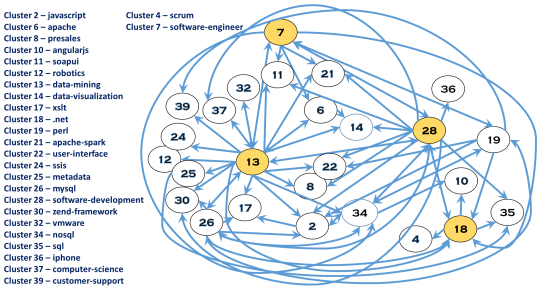


Fig. 2: Cluster Map after classification step for snapshot T1 country code 'at'.

is a principal cluster and a secondary cluster relative to Cluster-6, -7, -26, -28, and -37; and Cluster-4, -10, and -35 are a secondary cluster of principal cluster Cluster-18. Putting this in context of the graphs' contents, we might conclude that:

1. In general, four clusters play a vital role in shaping the skills in-demand for data science, labeled with software-engineer (Cluster-7), data-mining (Cluster-13), .net (Cluster-18), and software-development (Cluster-28).
2. Referring to the strategic diagram in Fig. 1b and this network map, we can conclude that Cluster-18 comprises four skills (.net, .net-framework, c++, c), and is the dominant software development environment during T1 period.

4.3 Skillsets Series Analysis

Once each cluster in all snapshots of a graph has been classified, then SCODIS will continue to establish the series of skills clusters. These generated series represent the evolution of demand given the labor market network, and further can be observed to see the transformation and trends based on the quadrant-strategical position of each cluster. Later, this observation will be the foundation for extracting interesting trends in the labor market that afforded insight into the demand evolution as well as the demand forecasting for the near future. Let S_i and S_{i+1} be snapshots of S at two consecutive time intervals with C_i and C_{i+1} denoting the set of clusters respectively. The framework computes the similarities between clusters in C_i and C_{i+1}, which shows the overlap of two consecutive clusters through a comparative analysis process. The similarity is calculated by using the *Similarity Index* (SI), derived from Callon's Dissimilarity Index [8], to measure the intersection of skills in two given clusters. Although it does not include the corresponding links in clusters directly, all skills in a cluster are nonetheless indirectly linked. Thus the metric can sufficiently capture a portion of cluster similarity. Given two clusters i and j, their similarity is defined as follows:

$$SI(W_i, W_j, W_{ij}) = 2 \times \left(\frac{W_{ij}}{W_i + W_j} \right), 0 < SI \leq 1 \tag{4}$$

where, W_{ij} is the number of skills common to cluster i and cluster j; W_i is the number of skills in C_i; and W_j is the number of skills in C_j. When generating a skills cluster series, the point of departure for each series must be a *crossroads cluster*. The choice for *crossroads*, however, confirms that all identified series are representative and play an essential role in depicting the transformation of a

network. The series creation process is executed until the end of the whole time interval under evaluation (T8) and is formally described in Definition 4.

Definition 4. *For each graph G_a, SCODIS framework generates a number of i series Ser_a in a time-ordered manner, by implementing comparative analysis between two consecutive clusters $C_i^k \in C_i$ from snapshot S_i and $C_{i+1}^x \in C_{i+1}$ from snapshot S_{i+1}, measured by a similarity index. To ensure notable intersection between two clusters, we set a minimum similarity threshold θ that defines the least number of similar skills should be matched between two compared clusters.*

4.4 Trends-based Analysis

The final process in our proposed framework is to projecting the *centrality* and *density* indices of each cluster in each series that has been established for each graph G_a. We use these primary indices to identify more complex trends and behavior, to monitor their stability and strategic position in order to decide which skillset is still on high-demand or currently emerging. In particular, we are interested in capturing the strategical tendencies of skills that contribute to the evolution of the graph. We define these strategical patterns to strengthen further our finding of the stability of the resulted series of clusters. By manually investigating the evolving clusters identified in our experiment, we found four categories of transition curves described as follows:

1. **Category A** represents clusters exhibiting a stable and steady quadrant location.
2. **Category B** displays a gradual shift from being central (Q1) to a less developed but possibly the emerging one (Q2), eventually reaching the margin of the whole network (Q4); or the other way around.
3. **Category C** embodies series that have clusters exhibit a steady and stable quadrant location during period T1 until T3, but then suddenly change track to other quadrants in T4.
4. **Category D** shows cluster series with less consistent trends that reflect a complex curve, with frequent and repeated changes in the clusters' quadrant location.

5 Results and Evaluation

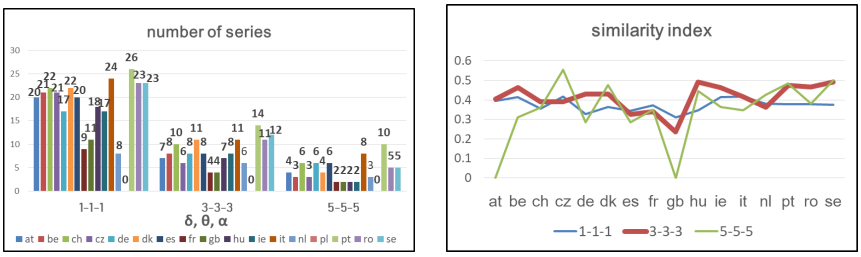
The objectives of our study are two-fold: i) to evaluate the adequacy of the Louvain algorithm in finding the relevant clusters aiming to complement or substitute Coulter algorithm, and ii) to identify whether the proposed method has the potential to correctly track established high-demanded skills clusters and the emergence of new skills clusters identified by their 'quadrant' location. Taking each of the above into consideration will enable us to determine whether we can confirm our hypothesis – that skill demand can be uncovered and predicted based on a sufficient amount of earlier observations. The first exercise returns the quantitative scientific result by determining the modularity gain of each clustering algorithm result, while the second returns the F-measure of the prediction result on each cluster's quadrant location (see Section 3).

Clustering results and evaluation. We implemented SCODIS framework in `Java` environment utilizing *Java Universal Network/Graph*[9] `JUNG` library for

[9] http://jrtom.github.io/jung/javadoc/

graph implementation. We determined the clusters for every 92 graphs constructed from the dataset of job adverts published in 17 countries from the year 2016–17 that were split into eight periods, resulting in a total of 3,310 clusters. We evaluated the clustering results of both the Louvain and Coulter algorithms by calculating each graphs' clustering modularity (Q) using Formula 2 and compared the two results. Our first observation is that on all clustering results for all 92 graphs by the Louvain algorithm, the modularity is consistently higher (average of 0.72) than the results of the Coulter algorithm (average of 0.41). Thus these results suggest that the Louvain clustering can identify a more robust community structure, with a modularity value approaching the maximum of $Q = 1$. In practice, values for such networks typically fall in the range of about 0.3 to 0.7, with higher values considered rare [6]. Second, unlike the Louvain algorithm, the Coulter algorithm generates clusters that only include part of the total nodes and edges from the source graph because it only considers the most 'important' nodes relative to the strongest weights of the edge.

Series generation results and evaluation. By taking the clustering result of the Louvain algorithm as an input, we establish the list of the series of skills clusters for each country. The series parameters (δ, θ, α) described in Section 4.2 and 4.3, control the construction of the resulted series. Two variables will help us decide which parameter values give a satisfying series: the amount of series, and the similarity index (SI).



(a) (b)

Fig. 3: The trends of: (a) the number of series, and (b) the similarity index, when increasing δ, θ, α.

Fig. 3a shows the number of established series as δ, θ, α increases from 1 to 5 respectively. We can see that the amount of series decreases when the parameters increase because it is difficult to find a very exact similar cluster. For the data and period (T1–T8) in consideration, there are 302, 135, and 71 established series when the framework uses series parameters 1-1-1, 3-3-3, and 5-5-5, respectively. From those established series, we got only 190, 30, and 10 complete series that have consistently one cluster in four quarterly periods (T1–T4), and 21, 1, and 1 series has a cluster in every eight periods (T1–T8). However, results obtained from calculating the *similarity index* (SI) that is shown in Fig. 3b confirm that the average of SI is higher (see the red line) when the parameters are based on the median value of the internal links, internal nodes, and secondary amount of each cluster (i.e., 3-3-3), are being used. Therefore, we set $\delta = 3$, $\theta = 3$, $\alpha = 3$ as

a trade-off between the number of series one hand and the *similarity index* on the other, proven better outcomes based on manual observation on the resulted series.

Prediction results and evaluation. To return a quantitative scientific result for the discovery and prediction of skills in high demand, the second objective of our evaluation considers the forecast quadrant location of established clusters against their next known location. An F-measure is thus computed to determine the forecast accuracy. Repeating the exercise based on the method presented in [13], we implement three traditional statistical prediction models that can sufficiently handle the trend characteristic and variability of our dataset, namely Naive, Simple Exponential Smoothing (SES), and Holt's linear trend method. A walk-forward validation is carried out to compare forecast values with actual values observed later than the period in question. Forecast calculations were conducted based on the 30 series (each containing four versions of an evolving skills cluster), starting by using two data points (T1-T2) to train the model, then the previous two to forecast the third, and the previous three to forecast the fourth. Thus, a total of 60 cluster forecasts are generated. We then compare the quadrant location of each cluster based on the forecast result to its actual quadrant in order to assess the performance of the forecasting method. By extension, the evaluation also determines whether our proposed framework is feasible to identify and monitor labor market skills demand. The precision, recall, and F-measure were computed for 60 clusters between the two datasets and is summarized in Table. 1.

Table 1: Prediction F-measure for each quadrant location in **all** investigated series.

Quadrant Label	**Precision**			**Recall**			**F-measure**		
	Naive	**SES**	**Holt**	**Naive**	**SES**	**Holt**	**Naive**	**SES**	**Holt**
Q1	69%	60%	57%	76%	86%	83%	**72%**	70%	68%
Q2	71%	72%	71%	67%	43%	40%	**69%**	54%	51%

Overall the F-measure ranges between 68%–72% for Q1 and 51%–69% for Q2, with no prediction potential observed for Q3 (0 clusters) and Q4 (1 cluster). Since both Q1 and Q2 are the focus of this study in which in-demand and emerging skills clusters are contained; thus, these strategical clusters are considered to be the answer to the question of this study. The F-measure result for the Q1 label is quite reliable and seems to indicate that the proposed framework is suitable for predicting highly-demanded and emerging skills. Interestingly, for both Q1 and Q2 label, the highest F-measure (69% precision, 76% recall) is achieved by the Naive method, and so confirming that the series of clusters are indeed stable and steady over time.

Furthermore, we fix our investigation on the series, which shows a stable transformation during the period being studied. By using four categories (**A**, **B**, **C**, and **D**) we have identified in Section 4.4, we re-calculate the F-measure based on these groupings. The results as shown in Table. 2, focus on the predictions for Q1 and Q2, i.e., we only consider the category-based predictions to determine which of the skill clusters are expected to shift to Q1 (most stable

and developed) and Q2 (emerging and become more important). The second column shows the distribution (ratio) of clusters within the series assigned to these four categories. The results indicate that forecasts are most accurate when the clusters are already within a stable and steady series (category A), followed by those in category B, and finally D, however no established series found in category C.

Table 2: F-measure results organized by identified category (and percentage of series belonging to each category).

Category	Percentage	Quadrant	Precision			Recall			F-measure		
			Naive	SES	Holt	Naive	SES	Holt	Naive	SES	Holt
A	**63.3%**	Q1	63%	48%	46%	86%	93%	86%	**73%**	63%	60%
		Q2	89%	91%	91%	71%	42%	42%	**79%**	57%	57%
B	**16.7%**	Q1	100%	100%	100%	60%	80%	80%	75%	**89%**	**89%**
		Q2	0	0	0	-	-	-	-	-	-
D	**20%**	Q1	57%	57%	50%	80%	80%	80%	**67%**	**67%**	62%
		Q2	60%	60%	50%	50%	50%	33%	**55%**	**55%**	40%

By looking at the ratio, the proposed framework can generate a stable and steady series of clusters more than half of the whole generated series. In summary, demand trends ascribed to category A (which represents 63.3% of the total), the highest F-measure is achieved by the Naive method at 73% and 79% for Q1 and Q2 respectively. Therefore it is clear that the proposed framework can generate a series of steady and stable clusters and to use it further to track established high-demanded skills clusters and the emergence of new skills clusters identified by their 'quadrant' location.

6 Conclusion

In this paper, we have proposed SCODIS: a quadrant-based framework that can observe the evolution of dynamic skills graphs and forecast which skillsets that are being requested. In fast-evolving sectors impacted by technology, their labor market tends to be large and, at the same time, very dynamic with rapid changes in the skills being referenced, thus a scalable and realistic evolutionary demand tracking method is required. The framework is based on the use of certain critical indices (*centrality*, *density*) and cluster categories (*isolated*, *secondary*, *principal*, *crossroads*), that facilitate our ability to compute and reason about novel strategical-oriented observations, which can offer new and interesting insights for the identification of dynamic requirement of such labor market domain. We have presented a dense and central-based model for constructing a series of skills clusters and have shown the use of quadrant location patterns for demand prediction. We have demonstrated the efficacy of our framework in predicting our job adverts dataset. The application of the quadrant-based method we obtained to a skillset demand prediction (emerging or re-inforced skillsets) scenario provided favorable results with sufficient prior observations. A critical next step for us is to extend our framework for comparing clusters over time. In this context, we would like to improve the performance of series generation by leveraging the knowledge graph for semantic similarity, which currently relies only on the contents from each cluster. We would also like to extend our framework to predict in-demand skillsets using other types of the learning-based forecasting model.

References

1. D. Smith and A. Ali, "Analyzing computer programming job trend using web data mining," Issues in Informing Science and Information Technology, vol. 11, pp. 203–214, 2014.
2. M. S. Sodhi and B.-G. Son, "Content analysis of OR job advertisements to infer required skills," Journal of the Operational Research Society, vol. 61, pp. 1315–1327, 2010.
3. N. Coulter, I. Monarch, and S. Konda, "Software engineering as seen through its research literature: A study in co-word analysis," Journal of the American Society for Information Science, vol. 49, pp. 1206–1223, November 1998.
4. V. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," Journal of Statistical Mechanics: Theory and Experiment, 2008.
5. E. M. Sibarani, S. Scerri, C. Morales, S. Auer, and D. Collarana, "Ontology-guided job market demand analysis: A cross-sectional study for the data science field," in SEMANTiCS, 2017, pp. 25–32.
6. M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," Physical Review E, vol. 69, pp. 026113, 2004.
7. X. Xu, N. Yuruk, Z. Feng, and T. A. J. Schweiger, "Scan: A structural clustering algorithm for networks," in KDD, 2007, pp. 824–833.
8. M. Callon, J. P. Courtial, and F. Laville, "Co-word analysis as a tool for describing the network of interactions between basic and technological research: The case of polymer chemistry," Scientometrics, vol. 22(1), pp. 155–205, 1991.
9. P. Lee, L. V. S. Lakshmanan, and E. E. Milios, "Incremental cluster evolution tracking from highly dynamic network data," in ICDE, 2014, pp. 3–14.
10. J. Hopcroft, O. Khan, B. Kulis, and B. Selman, "Tracking evolving communities in large linked networks," Proceedings of the National Academy of Sciences, vol. 101 (suppl 1), pp. 5249–5253, April 2004.
11. A.-S. Dadzie, E. Sibarani, I. Novalija, and S. Scerri, "Structuring visual exploratory analysis of skill demand," Web Semantics, vol. 49(0), 2018.
12. S. Fortunato and A. Lancichinetti, "Community detection algorithms: a comparative analysis," in the Fourth International ICST Conference on Performance Evaluation Methodologies and Tools, 2009, pp. 27:1–27:2.
13. E. M. Sibarani and S. Scerri, "Generating an evolving Skills Network from Job Adverts for high-demand Skillset Discovery," in WISE, 2019.
14. P. A. Todd, J. D. McKeen, and R. B. Gallupe, "The Evolution of IS Job Skills: A Content Analysis of IS Job Advertisements from 1970 to 1990," MIS Quarterly, vol. 19 (1), pp. 1–27, 1995.
15. I. A. Wowczko, "Skills and Vacancy Analysis with Data Mining Techniques," Informatics, vol. 2, pp. 31–49, November 2015.
16. A. Aken, C. Litecky, A. Ahmad, and J. Nelson, "Mining for Computing Jobs," IEEE Software, vol. 27 (1), pp. 78–85, Jan.-Feb. 2010.
17. S. Surakka, "Analysis of Technical Skills in Job Advertisements Targeted at Software Developers," Informatics in Education, vol. 4 (1), pp. 101–122, January 2005.
18. M. A. Kennan, P. Willard, D. Cecez-Kecmanovic, and C. S. Wilson, "A Content Analysis of Australian IS Early Career Job Advertisements," Australasian Journal of Information Systems, vol. 15 (2), pp. 169–190, May 2009.
19. S. Asur, S. Parthasarathy, and D. Ucar, "An Event-based Framework for Characterizing the Evolutionary Behavior of Interaction Graphs," in KDD, pp. 913–921, 2007.
20. M.-S. Kim and J. Han, "A Particle-and-Density Based Evolutionary Clustering Method for Dynamic Networks," in VLDB, pp. 622–633, August 2009.