UNIVERSITY OF AMSTERDAM

UvA-DARE (Digital Academic Repository)

Bridging the demand and the offer in data science

Belloum, A.S.Z.; Koulouzis, S.; Wiktorski, T.; Manieri, A.

DOI
10.1002/cpe.5200
Publication date
2019
Document Version
Final published version
Published in
Concurrency and Computation: Practice and Experience

License
CC BY

Link to publication

Citation for published version (APA):
Belloum, A. S. Z., Koulouzis, S., Wiktorski, T., & Manieri, A. (2019). Bridging the demand and the offer in data science. *Concurrency and Computation: Practice and Experience*, *31*(17), Article e5200. https://doi.org/10.1002/cpe.5200

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, P.O. Box 19185, 1000 GD Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (https://dare.uva.nl)

Download date:15 Sept 2026

Received: 31 October 2017 | Revised: 9 October 2018 | Accepted: 21 January 2019
DOI: 10.1002/cpe.5200

WILEY

RESEARCH ARTICLE

Bridging the demand and the offer in data science

Adam S.Z. Belloum[1] | Spiros Koulouzis[1] | Tomasz Wiktorski[2] | Andrea Manieri[3]

[1]Institute of Informatics, Amsterdam, The Netherlands
[2]University of Stavanger, Stavanger, Norway
[3]Engineering Ingegneria Informatica SpA, Rome, Italy

Correspondence
Adam S.Z. Belloum, Institute of Informatics, 1098 XH Amsterdam, The Netherlands.
Email: a.s.z.belloum@uva.nl

Funding information
European Union's Horizon 2020 research and innovation programme, Grant/Award Number: 675419

Summary

During the last several years, we have observed an exponential increase in the demand for Data Scientists in the job market. As a result, a number of trainings, courses, books, and university educational programs (both at undergraduate, graduate and postgraduate levels) have been labeled as "Big data" or "Data Science"; the fil-rouge of each of them is the aim at forming people with the right competencies and skills to satisfy the business sector needs. In this paper, we report on some of the exercises done in analyzing current Data Science education offer and matching with the needs of the job markets to propose a scalable matching service, ie, COmpetencies ClassificatiOn (E-CO-2), based on Data Science techniques. The E-CO-2 service can help to extract relevant information from Data Science–related documents (course descriptions, job Ads, blogs, or papers), which enable the comparison of the demand and offer in the field of Data Science Education and HR management, ultimately helping to establish the profession of Data Scientist.

KEYWORDS

career development, data science, data science job market, education

1 | INTRODUCTION

Data Science is an emerging field of science, which is rapidly gaining importance in both academia and business sectors. During the last years, we have registered a sudden increase in the number of universities and industry programs and courses labeled as Data Science (or Big Data) or claiming to offer Data Science related content. A survey has been performed in the context of the EU funded EDISON project[1] and aimed at identifying the skills and knowledge present in these offerings. The survey covered over 300 educational programs and over 100 academic and industry courses; it was primarily based on what was advertised and published on the courses' websites. The information ranged from detailed to limited, which increased the complexity of the analysis. Nevertheless, to our knowledge, it is the most complete and detailed analysis up to date. The inventory is publicly available on the EDISON website[2] and a detailed analysis is available through earlier publication.[3] Another challenge facing the analysis of the Data Science landscape is the absence of commonly accepted definitions for this emerging field; the collected information is highly unstructured and noisy (a term can be used to describe different concepts while multiple terms are used to describe one single concept).

The problem becomes bigger when analyzing the Data Science job vacancies, ie, an initial term extraction process based on 1000 job Ads using a simple term frequency count produced approximately 300 000 terms. One of the few accepted definitions of Data Science is proposed by NIST. NIST Big Data Working Group (NBD-WG) published their first release of Big Data Interoperability Framework (NBDIF) in September 2015,[4] consisting of seven volumes. Volume 1 provides a number of definitions, in particular, those of Data Science, Data Scientist and Data Life Cycle, which have been used as a starting point for this analysis. NIST defines Data Science as a set of multidisciplinary competencies and skills at a very high level of abstraction. However, the definition proposed by NIST is not enough to analyze educational programs, in particular on how well they cover the necessary competencies. Within the EDISON project, a more detailed study showed large discrepancies in the Data Science field, at least in terms of content and focus on required skills.

The majority of Data Science–related programs are offered by Computer Science departments. These programs are the most generic compared to the programs offered by other departments, which are more tuned to a specific field (eg, genomics or bioinformatics) or disciplines

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.
© 2019 The Authors Concurrency and Computation: Practice and Experience Published by John Wiley & Sons Ltd.

(eg, statistics). The study showed the emergence of Data Science programs offered by multiple departments, which is not common in high education. There is quite a variety of contents that is now published under the umbrella of Data Science label. The educational programs covered by the study showed that most of these programs cover Data Analytics, as defined by NIST, to a sufficient extent. However, (computer) engineering competencies are often missing in programs not originating from computer science or computer engineering departments.

In this emerging and dynamically evolving field of science, a number of challenges are still facing the different stakeholders on both the demand and the supply sides, ie, students have to find the right educational program that will lead them quickly to find their first job, human resources departments have to select the right candidate for a given professional profile and describe the right competencies required for a give vacant position, and finally, trainers and educators have to design and adapt Data Science programs to develop the competencies and skills needed for the Business sector as well as Research careers. We analyzed 1000 job Ads related to the Data Science with the following objectives: (1) to create a taxonomy of the terms used in Data Science job Ads and (2) to discover the relationships between the terms to identify skills and competencies required by these job Ads. The taxonomy helps to identify the relationships between the skills and competencies used in Data Science related documents, such as course descriptions, job Ads, blogs, and papers. It will help to map Data Science generic terms used in job ads, CVs, and course descriptions into more concrete skills and competencies, which, in turn, can help to avoid a mismatch between job and CV profiles, courses, and trainings. One primary use of the taxonomy is to validate and update the EDISON Data Science Framework over time, making it easier to maintain beyond the project lifetime.

2 | TERM EXTRACTION

The skills and competencies required from data scientists are hidden in job descriptions, scientific papers, and articles, blogs, or even books. Analyzing millions or even thousands of documents manually to identify skills is labor-intensive, and therefore, we can employ natural language processing and text-mining methods to extract these skills through the analysis of terms used in a given set of documents related with Data Science. The output of this process is a hierarchical taxonomy of the skills required by data scientists[5] described by the terms used to refer to the tasks and output expected for any given skill. Automating the extraction of skills from job Ads and other documents is a key step to being able to address stakeholder needs; in particular, it may help in following trends and thus generating the knowledge needed to define the appropriate career paths or it may be useful to keep curricula up-to-date with job market demand over time. In this section, we describe the approach used to develop the tools to automate the analysis of a large corpus of documents.

To perform the job Ads analysis, we followed a two-step approach. (1) We use the term frequency to extract the most commonly used terms in the job Ads and (2) we extract the relationships between the words to create the Data Science taxonomy.

Term or terminology extraction attempts to identify the body of terms used in a subject or content.[6] A term may be a single/multi-word expression that has a particular meaning within a specific domain.[6,7] There are three main approaches to term extraction, ie, (1) statistical, (2) linguistic, and (3) hybrid.

Statistical approaches produce a ranked list of terms identifying the most important terms extracted from a text and usually start by identifying all the unique words that appear in a text. They also construct all possible n-gram that can be identified. To determine the "term-hood" of each term and rank it accordingly, statistical approaches use several metrics. Term frequency (TF) is one of the most common and simple metrics used for statistical term extraction; it measures how frequently a term occurs in a document. The Inverse Document Frequency (IDF) measures how important a term is. Combining the two provides the term frequency-inverse document frequency (TF-IDF), which is a statistical measure, used to evaluate how important a term is to a document in a corpus. The TF-IDF is the most established measure. However, other metrics that rank candidate terms such as T-score, C-value, Dice coefficient, etc, may offer the E-CO-2 analyzer more accuracy for the term extraction.

Linguistic or contextual approaches attempt to identify syntactical patterns in a text in order to extract terms. Usually, terms tend to have characteristic syntactic structures. Part-of-speech (POS) taggers are used to identify these structures. However, linguistic approaches are language-dependent and therefore are not flexible and adaptable to other languages.

Hybrid approaches use a proper combination of the previous two steps.[8] Most of these approaches depend on statistics and use syntactic rules as a complementary method to filter the appropriate terms. Therefore, in these hybrid approaches, a linguistic analysis is performed to exclude words like pronouns and verbs. This step may also be applied to identify patterns and sequences of part-of-speech and pass these sequences on to statistical measures to rank possible terms. Other approaches include linguistic information in the ranking process.[9] The biggest challenge for any term extraction approach is its validation. Judging the accuracy of any approach involves a human expert that needs to evaluate the results.

2.1 | Word-sense disambiguation

Word-sense disambiguation is the task of identifying which sense (meaning) of a term is used in a sentence or in a set of documents when the term has multiple meanings.[10,11] In general, commonly used approaches for Word-sense disambiguation requires two inputs, ie, a dictionary that contains the senses, which have been already disambiguated, and a corpus of terms to be disambiguated. Dictionary-based methods exploit the hypothesis that words that appear "near" a term are related to each other and that this relation can be observed in the definitions of the terms.[12] Therefore, finding all possible dictionary definitions may disambiguate terms and select the one which has the biggest word overlap between the definition and the related words within a given text.

Algorithm 1 Training process for the text profiler.

```
1: procedure TRAIN(categoryFiles, contextTermsFile)
2:     for all f ∈ categoryFiles do
3:         catTerms = extractTerms(f);            ▷ do hybrid term extraction
4:         contTerms = getTermsContaning(catTerms);          ▷ enrich terms
5:         terms = catTerms.addAll(contTerms);          ▷ combine all terms
6:         for all te ∈ terms do
7:             definition = WSD(te);      ▷ perform WSD to extract definition
8:             definitions.add(definition);      ▷ collect definitions of category
9:         for all df ∈ definitions do
10:            tokens = tokenize(df);
11:        for all tk ∈ tokens do
12:            score = TF-IDF(tk);
13:            wordVector.put(tk, score);          ▷ calculate ifidf for all words
14:        Save wordVector for each category
```

FIGURE 1 Pseudo-code describing the profiling process

Algorithm 2 Process for profiling a document.

```
1: procedure PROFILE(textFile)
2:     tokens = tokenize(textFile);
3:     for all tk ∈ tokens do
4:         score = TF-IDF(tk);
5:         wordVector.put(tk, score);          ▷ calculate ifidf for all words
6:     categoriesVecotrs = loadCategoryVectors();
7:     for all ca ∈ categoriesVecotrs do
8:         similarity = cosineSimilarity(ca, wordVector);
9:         profile.put(ca.name, similarity);
10:    save profile;
```

FIGURE 2 Pseudo-code showing the algorithm for training to obtain the profile of a document for a related subject

2.2 | Text profiling

Based on the techniques and approaches described in the previous sections, a *text profiler* has been implemented and can be trained to provide the profile of a document based on a set of predetermined categories. The first step is to determine the categories to which the documents need to be mapped. For each of the categories, it is necessary to manually set a collection of keywords that are representative of the target category. The collection of representative keywords is further enriched with terms extracted from the context corpus. This context corpus is a collection of documents that are closely related to the subject that encloses the categories to be used as a mapping target. These documents may contain definitions or simply revolve around a specific subject and are used as an input for a term extraction process, which will save the extracted terms in a context terms file. Figure 1 presents the pseudo-code of the profiling process.

When the terms are obtained, a Word-sense Disambiguation on each term is performed and its definition extracted. The set of extracted definitions represent the collective meanings for each category and can be encoded as a vector that numerically represents each category. The constructed TF-IDF vectors are used as features vectors[8] for the following clustering algorithms: K-means,[13] Hierarchical,[14] Filtered Clustering,[15] Farthest First,[16] and EM.[17]

To achieve this goal, we calculate TF-IDF for all words contained in the extracted definitions of each category. These TF-IDF values are saved and used to measure the cosine similarity of the documents. Figure 2 presents the pseudo-code of the training process to obtain the profile of a document for a related subject; we first need to obtain a vector that will numerically represent that document.

3 | COMPETENCIES CLASSIFICATION SERVICE ARCHITECTURE

The COmpetencies ClassificatiOn (E-CO-2) service is a distributed automated service designed to enable Data Science gap analysis. It can identify the similarity of a document against a set of predefined categories. It can, therefore, be used to perform a gap analysis following the EDISON classification to identify mismatches between education offering and business sectors demand. Students, data analysts, educators, and other stakeholders can use this tool to identify the gaps in their skills and competencies and identify the most suitable educational path to fill these gaps.[18] Moreover, by constantly collecting data from sources like job Ads and postgraduate programs, we will be able to identify trends from both the job market and education.

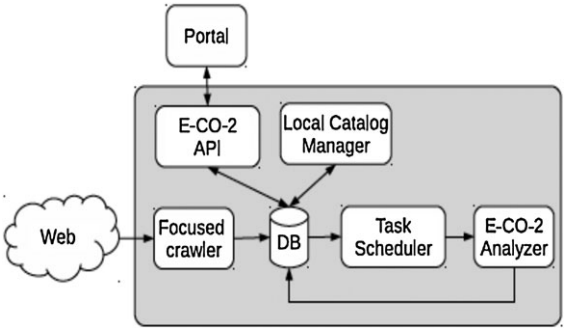

FIGURE 3 E-CO-2 service architecture. E-CO-2 will act as a backend to the community portal 3 through REST API

Figure 3 shows an overview of the architecture of the E-CO-2 service. At the top level, the E-CO-2 API provides methods that allow an API consumer service to analyze documents and retrieve results. The Local Catalogue Manager plugs-in into multiple data sources giving the opportunity to have a unified view of various data sources. Connecting to a larger overlay network, it is possible to discover and request multiple datasets that can be used to perform more analytics. The *focused crawler* is used to collect documents from the web related to the job market and Data Science education. The database is used to efficiently store and query input documents, analysis results, the context corpus, and the category vectors (see Section 3.3). The task scheduler queries the database at regular time intervals for new documents that need to be analyzed or updates in the context corpus and schedules tasks for the E-CO-2 analyzer. The E-CO-2 analyzer is the main analysis component responsible for (1) providing a similarity matrix of an input document against the competence groups derived from the EDSF tables and (2) generating category vectors based on a context corpus.

3.1 | E-CO-2 API

The E-CO-2 service is compliant with the RESTful approach as it is to provide a REST (Representational state transfer) API, which allows other systems to access and manipulate its resources using a uniform set of stateless operations. REST is a stateless protocol with standard operations that help in performance and reliability. It is becoming the de facto standards are REST over HTTP. REST may be less descriptive than SOAP (Simple Object Addressing Protocol) but is less strict, which allows for greater flexibility. REST is also lighter and less complex than SOAP. Other transport protocols may be more suitable for delivering data to a given service. However, HTTP is widely used as a transport protocol because HTTP actions give a clear indication on the service design and functionality. Moreover, the design of commonly used Web portals follows the REST specifications. Therefore, we opted to use REST over HTTP for delivering the service API. The API to analyze documents and retrieve results provides the following methods.

- POST http://host/e-co-2/classification/doc/: This method stores the incoming document in a database. Given that the classification is done asynchronously, this method returns to the caller a unique id that can be used to query the result.
- GET http://host/e-co-2/classification/fidg: Using the unique id obtained from the method described in POST method, this call returns the classification results of a document. If the classification is not finished, this method returns a 202 HTTP code signaling that request has been accepted for processing, but the processing has not been completed.
- GET http://host/e-co-2/jobs/fidg: Using the unique id obtained from the method described POST method, this call returns a sorted list of jobs that are most similar to the analyzed document. If the sorting is not finished, this method returns a 202 HTTP code signaling that request has been accepted for processing, but the processing has not been completed.
- GET http://host/e-co-2/courses/fidg: Using the unique id obtained from the described POST method, this call returns a sorted list of courses that are most similar to the analyzed document. If the sorting is not finished, this method returns a 202 HTTP code signaling that the request has been accepted for processing, but it is still not completed

The E-CO-2 API follows standardized protocols and methods (REST API, JSON, and service-oriented architecture) to make the integration with the portal easier and also provide re-usability and flexibility. This is achieved by separating the business logic from the exposed functionality. Such an approach allows us to replace the text analysis with more efficient implementations (eg, replace MapReduce jobs with Spark or Storm jobs).

3.2 | Focused crawler

The focused crawler is used to collect document related to the job market and Data Science education. The focused crawler systematically browses specific resources on the Web to retrieve new documents. The behavior of the focused crawler is defined by the following policies.

- Selection policy stating which resources should be searched. In order to prioritize which resources should be visited first, a metric that indicates the importance of a resource is required. This can be a function of the popularity and the number of documents obtained.

- Revisit policy that indicates when to check for updates on each resource. To always have accurate information, the crawler checks the resources that update their documents more often. Therefore, the visiting frequency is proportional to the update frequency.
- Politeness policy that attempts not to overload resources. In some cases, the documents retrieved may require considerable bandwidth. In addition, a crawler can set a heavy load on servers especially if it is implemented with a high degree of parallelism, which may be regarded as an unwanted DDoS attack.

The crawler is composed of two components, ie, a scheduler and a database. The scheduler enforces the policies described above and a document fetcher responsible for obtaining the relevant document. Since resources may have different formats for representing documents and might expose different APIs, we will then have a different fetcher for each resource. These types of APIs are mostly focused on interacting with that particular system. Some web pages make changes to their structure and data while these changes in the data structure would be reflected in the API months later. Moreover, unavailability and downtime at some API endpoints may go unnoticed for days. Since we want to have up-to-date information from many resources, we incorporate the offered APIs to the crawler but also use the publicly available contents while considering the bandwidth limitations and the crawling policies mentioned above.

Database

The database is used to efficiently store and query input documents, analysis results, a context corpus, and category vectors. Queries can be used to perform analytics in relation with trends. For example, given that the database is populated with enough historical data, one can query for the average similarity of job Ads in relation to data analytics. Several parameters can influence the choice of database. Since most data representations these days are in JSON format, a MongoDB may be more appropriate. One of the benefits of using MongoDB is that there is a wide range of tools that allow us to store and query data and map them naturally to object-oriented programming languages. Moreover, the document-based architecture of MongoDB allows our schema to evolve with the requirements.

Task Scheduler

The task scheduler queries the database at regular time intervals for new documents that need to be analyzed or updates in the context corpus and schedules tasks for the E-CO-2 analyzer. This component makes sure that the correct arguments are set and instantiates the E-CO-2 analyzer to execute a MapReduce job. This component is responsible for setting the priority of tasks. For example, the E-CO-2 analyzer may be set to analyses a large set of documents, but at the same time, a user uploads a CV to the portal. In this case, the scheduler should set the user's request for higher priority. In the overall design, it is important to have isolation between components. The task scheduler gives a good separation between the actual analysis, the API communication, and the database implementation. There are several issues when considering task scheduler design; one of them is task cancellation. In many cases, tasks should be canceled either because the results of the task are not needed anymore or because that particular task is using too many resources preventing the completion of other tasks. As mentioned earlier, it is important for tasks to have priorities. However, care must be given to ensure that low priority tasks will not wait forever.

3.3 | E-CO-2 analyzer

Term extraction is one of the key aspects of the E-CO-2 analyzer. Term or terminology extraction attempts to identify the body of terms used in a subject or content of the targeted document. A term may be single or multi-word expressions that have a particular meaning within in specific domain. The E-CO-2 analyzer uses a hybrid approach using POS taggers to filter out words and attempt to reduce the search space.

The E-CO-2 Analyzer based on TF-IDF metric has been implemented as a series of MapReduce jobs to perform two main functions, ie, training and classification. The choice to use MapReduce is motivated by the many published studies in the field of text processing, which show good performance achieved by various MapReduce implementations.[19-22]

Training task (Figure 4A) performs term extraction with the addition of the a priori algorithm and constructs a "category vector" for each of the categories or competences we wish to identify. For each of the categories, it is necessary to manually set a collection of keywords, definitions, and descriptions that are representative of each category. The quality of the classification depends on the accuracy of these keywords, definitions, and descriptions. Therefore, the keywords, definitions, and descriptions have to be concrete, representative, and contain specific terms.

Classification (Figure 4B) compares an input text, which may be a job ad, a CV, or a curriculum description with the set of available category vectors that are created during the training phase. For each of the category vectors, the classification provides a similarity measure that indicates how close the input document is to the available category vectors each representing a category or a competence.

4 | RESULTS

In this section, we present our analyses of both Data Science education and job market to stress the need for a synergy between the two sides. The Data Science education analysis was performed manually and took several months while the job market analysis was automated and data was collected in a few days. It demonstrates a clear need for developing the E-CO-2 as a service to help to accelerate the future gap analyses.[23] We describe two scenarios to show how E-CO-2 service can be used to extract compare Data Science related text.

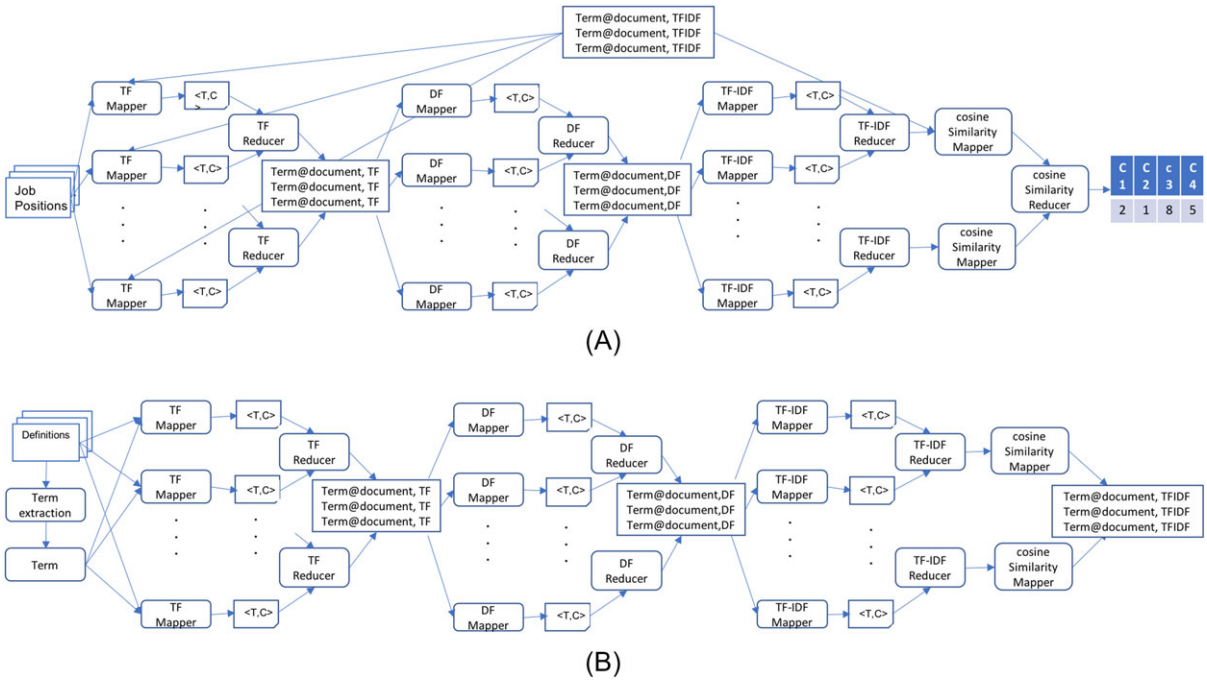

FIGURE 4 E-CO-2 implemented as a series of MapReduce jobs; (A) The training pipeline: Term extraction, Document Frequency, and finally we calculate the TF-IDF metric, which produces the category vectors; (B) The classification pipeline: The category vectors are compared with the output of the TF-IDF metric calculated from the input text; finally, we calculate the cosine similarity

4.1 | Analysis of the data science educational programs

Each program in the EDISON inventory was analyzed in detail to determine to what extent courses in its curriculum cover the identified competence groups. Some courses might naturally cover more than one group. In some cases, especially in the case of project courses (eg, master thesis), they might provide coverage of all areas simultaneously. Such aspects were accounted for during our analysis.

A job market study and analysis for Data Science and Data Science-enabled vacancies was conducted at the initial stage of the project. It resulted in the identification of 3 core competence groups, ie, (1) Data Science analytics (including statistical analysis, machine learning, data mining, business analytics, and others), (2) Data Science engineering (including software and applications engineering, data warehousing, big data infrastructure, and tools), and (3) domain knowledge and expertise (subject/scientific domain related). These core competence groups correspond to the skills groups identified in NIST Big Data Interoperability Framework.[24] Skills and competencies are equivalent terms; however, competencies are more often used in an education setting, whereas the term skill is more common in the professional training setting.

In addition, two meta competence groups we have identified by the project, ie, (1) data management and governance (including data stewardship, curation, and preservation) and (2) research methods for research related professions and business process management for business-related professions. However, we decided not to include these groups as separate in this analysis. Due to the limited quality of data, as most programs do not define competencies well enough, including two additional groups that overlay the three core ones could lead to misleading results. The results were discussed with EDISON Expert Liaison Group, whihch consisted of leading industry and academia representative, to ensure the quality of results.

One should expect, in principle, roughly uniform coverage of each competence group. Balance in covering competence groups is a key to educating successful data scientists. However, small differences in coverage can of course occur. EDISON proposed that the disparity between the most and least covered competence group should not exceed 20 pp. (percent point), so that the program can still cover the whole spectrum of Data Science field. We deliberately avoid using exact points since European and American system operate differently.

This disparity should rather be even lower, but we assumed that a stricter criterion would be misrepresentative at this early stage of Data Science curriculum development. Between 20 pp and 30 pp, we classified programs as having a small imbalance. If the disparity exceeds 30 pp, it effectively means that one of the competence groups cannot be covered at all or only to a marginal extent, while one of the others exceeds 60%, which means it dominates the program. We classified such programs as having a significant imbalance.

Competencies and learning outcomes are seldom defined explicitly. The presented analysis should be seen as an approximation. Simultaneously, considering a large number of programs that were analyzed and that a simple competence group model was used; we believe that the analysis is consequential so long as one is careful about the type of conclusions to draw from it.

We present here a short summary of the analysis. 59% of European (Figure 5A) and 50% of Non-European (Figure 5B) programs are significantly imbalanced. This means that one of the competence groups is not covered properly or not at all. Additional 14% and 15% of programs, respectively, have smaller imbalances. Only 27% and 35% of the programs, respectively, could be considered balanced, despite the fact that the threshold we

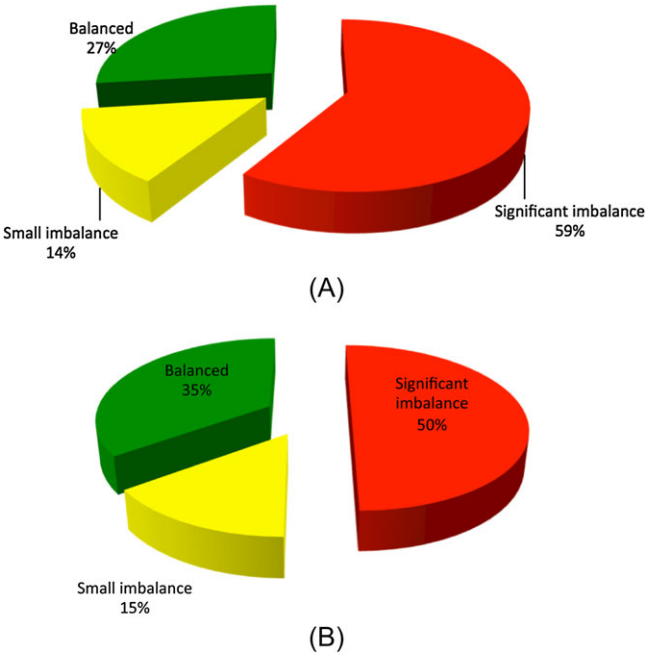

FIGURE 5 A, Balance of European Programs; B, Balance of Non-European Programs

set was relatively low. The distribution of the imbalance between competence groups is not equal. The Data analytics group is usually covered to a sufficient extent in almost all programs. On the other hand, (computer) engineering competencies are often missing in programs not originating from computer science. At the same time, domain knowledge is often overlooked for programs from the aforementioned departments.

Another issue is the uncontrolled flexibility of around 20% of the programs. The way their elective courses are structured might lead to imbalance for a particular student. Flexibility and electives should, of course, be encouraged, but they should be divided into competence groups and students should choose equally from each group.

In a large subset of programs, in which domain knowledge appears to be properly covered, a deeper inspection reveals that the offered courses over emphasize generic management and business skills. There is little conceptual connection between courses offered to cover domain knowledge and those covering other competence groups.

Such courses might be relevant to certain programs and business schools, but it seems they are used as a rushed solution, due to the limited relation of these courses to the rest of the program, to superficially cover missing elements in the program. It is important to note that we excluded from this argument specialized courses in economy, financial analysis, or similar.

Many programs appear to place an equal sign between data scientist and business analyst. While business analysis might be considered a special case of Data Science, the opposite is certainly not correct.

Finally, in Figure 6, we look at balance in programs depending on what type of source they are coming from. We clearly see that, for almost all cases, more than 50% are significantly imbalanced. The only exceptions are programs that come from cross-department collaboration, where more than 50% of programs are balanced. There are some minor differences between other sources, but they should not be over-interpreted in the early stages of Data Science curricula development.

4.2 | Analysis of the job ads in the business sector

To analyze the Data Science job market and identify its needs, we applied well-known Text processing data processing pipeline described in Section 2. The term extraction was performed on a dataset containing 1000 job Ads for data scientist for several experience levels and functions, extracted from LinkedIn®. The term extraction process showed that the majority of terms used Data Science job Ads is related to computer science, math, and statistics, indicating that these skills are relevant in this field. Moreover, specific programming languages and platforms seem to be included in many job Ads and further investigation could reveal which programming languages and platforms are considered important compared with.[25,26] For example, the degree of a hypernym node that connects more hypernym-extracted from a text should give an indication of the term's importance. This way, the graph can become more "balanced" by providing an intuitive level of abstraction in terms.

Figure 7A shows the required years of experience in the job Ads: 82% of job Ads targeting mid-career applicants, while Entry-level opening represents only 10%. Candidates suited for Mid-career opening should have graduated at least five years before the emergence of Data Science as a professional profile, and thus, it is likely that current potential applicants might not have the multidisciplinary background required for the job opening. Figure 7B shows the type of work the applicants will be doing in her/his job.

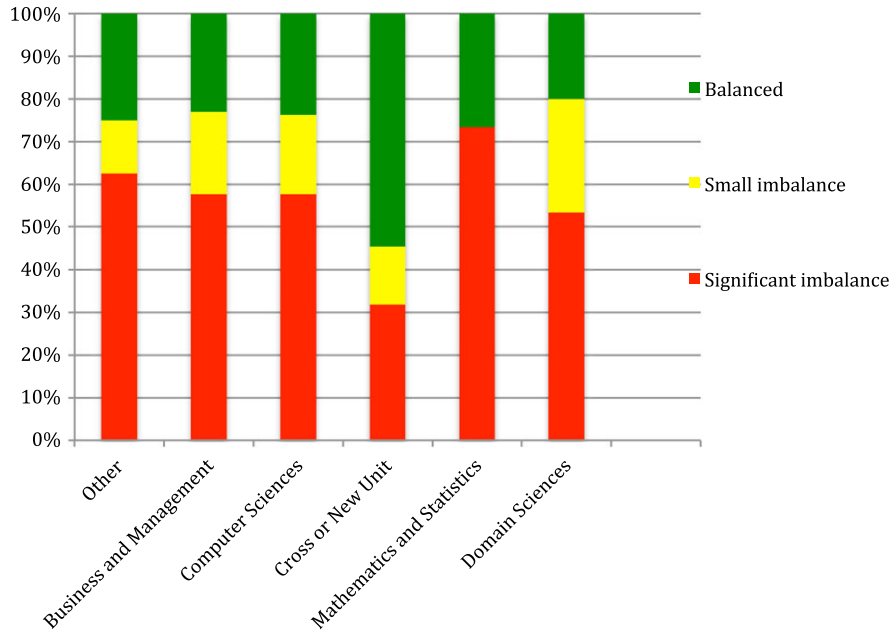

FIGURE 6 Balance of Programs w.r.t. Department, which owns the program

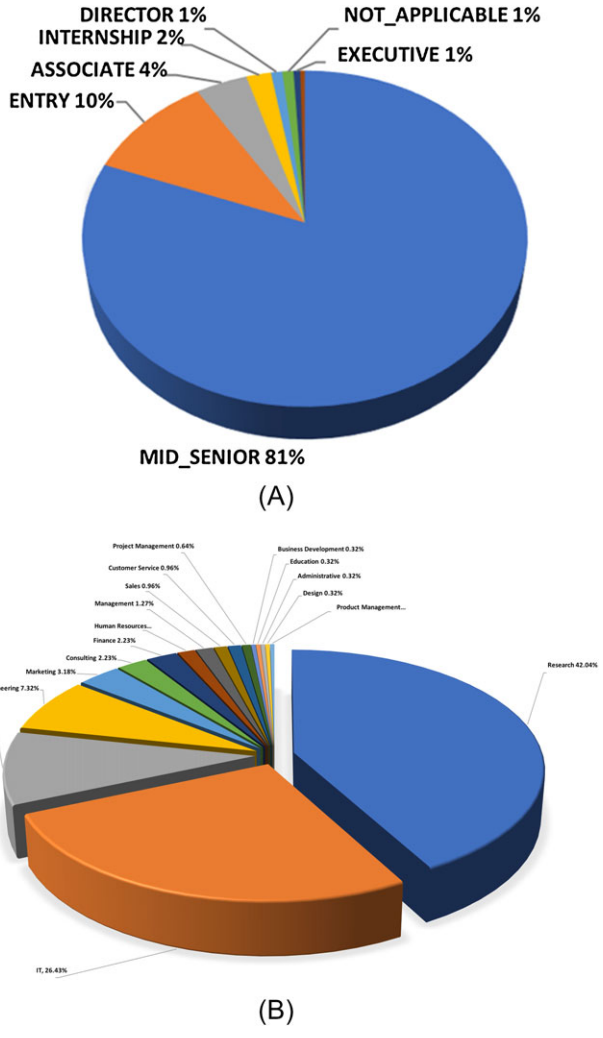

FIGURE 7 Analysis of information collected from the dataset showing the number of year of experience requested in the job Ads (A) and the profile required in the job Ads (B)

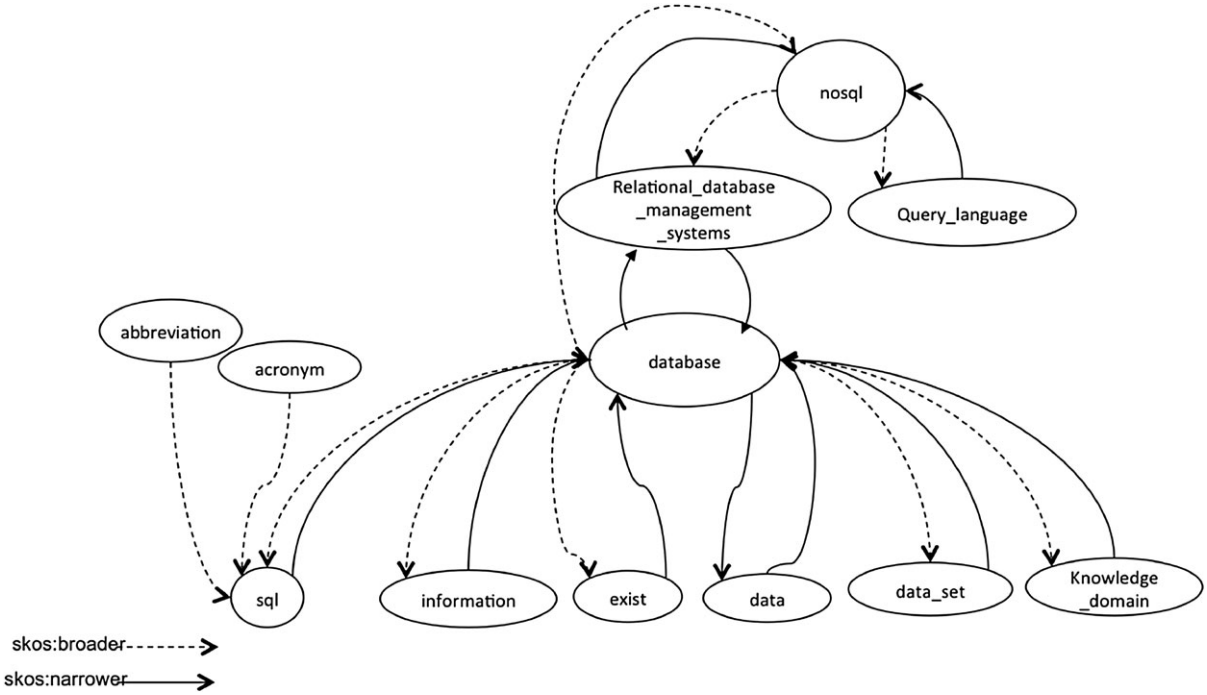

FIGURE 8 Sample of hierarchical taxonomy extracted following the SKOS specification. This is a result for querying the term "database." The entire taxonomy is made up of 773 terms

In Sections 4.1 and 4.2, we presented the analysis of both current Data Science education offerings and the needs of the job market, and immediate conclusion of two analysis is that job market is looking more for senior and mid-career staff (82%) rather than fresh graduates (11%). This suggests that education has to also offer short training education and not only long programs BA, MSs, and PhD. A second important outcome which came out of the analysis is the fact that the job offers are dominated by profiles in Research and IT departments (Research 42%) and (IT 26%); this also suggests that education has to carefully balance between teaching Engineering skill and Scientific research method. In our analysis, only a fraction (27% of EU European Programs and 35% Non- European Programs) offer a well-balanced education.

4.3 | E-CO-2 service for aligning the supply and the demand in data science

4.3.1 | Building the taxonomy

E-CO-2 service helps building taxonomies from a relevant corpus. To move from simple word count to more complex statistical measures, words need to be "lemmatized," which is the process of grouping together different forms of a word so they can be analyzed as the same (eg, "scientist" and "scientists" should be considered as the same word). These processes allow building the term dictionary, which is a list of all unique words used in the corpus. In the next step, we use a set of hybrid term extraction methods to rank the relevant terms. During the relation discovery, we first build non-hierarchical relations, and with the use of hypernym-hypernym relations, we build hierarchical relations within each cluster. The hierarchical relation discovery used hypernym-hypernym relations included in online dictionaries. Hypernym shows the relationship between a generic term and a specific instance. We followed a three-step approach.

- The dataset included job Ads from both SMEs and large companies. The term extraction process produced approximately 300 000 terms. Using statistical methods and we reduced the terms to approximately 50 000.
- As a second step, we used a hybrid method that uses both linguistic (Part-of-speech tagging) and statistical analysis for ranked term extraction.[13]
- The extracted terms are then grouped together with the extracted terms from the hybrid method to form non-hierarchical relations.
- Finally, we performed a hierarchical relation discovery using hypernym-hyponym relations included in online dictionaries. Figure 8 shows a small part of hierarchical taxonomy; the complete taxonomy is available in the Resource Description Framework (RDF) format.[27]

The taxonomy is aimed at validating the outcome of surveys and markets analyses and future updates of the EDSF documents.

4.3.2 | Comparing data science–related text

E-CO-2 analyzer uses as a reference for comparing the various profiles the Data Science competence groups defined in the EDISON Data Science Framework (EDSF).[28] The four competence groups described in the NIST definition are further extended in the EDSF and refined into subgroups of competencies. In total, 30 competencies have been identified and used as a reference for comparing Data Science programs, job Ads, user's

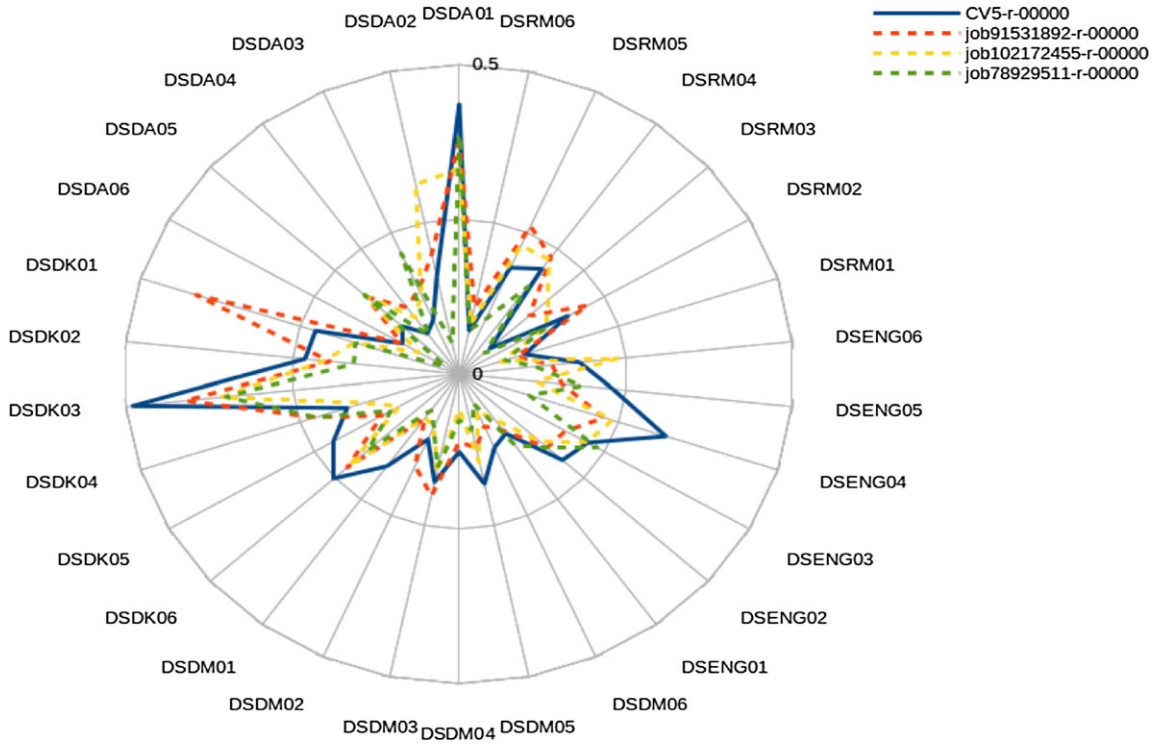

FIGURE 9 Detailed profile of C.V. and individual job profiles for each of the 30 competence groups (definition of competence group see the work of EDISON[28])

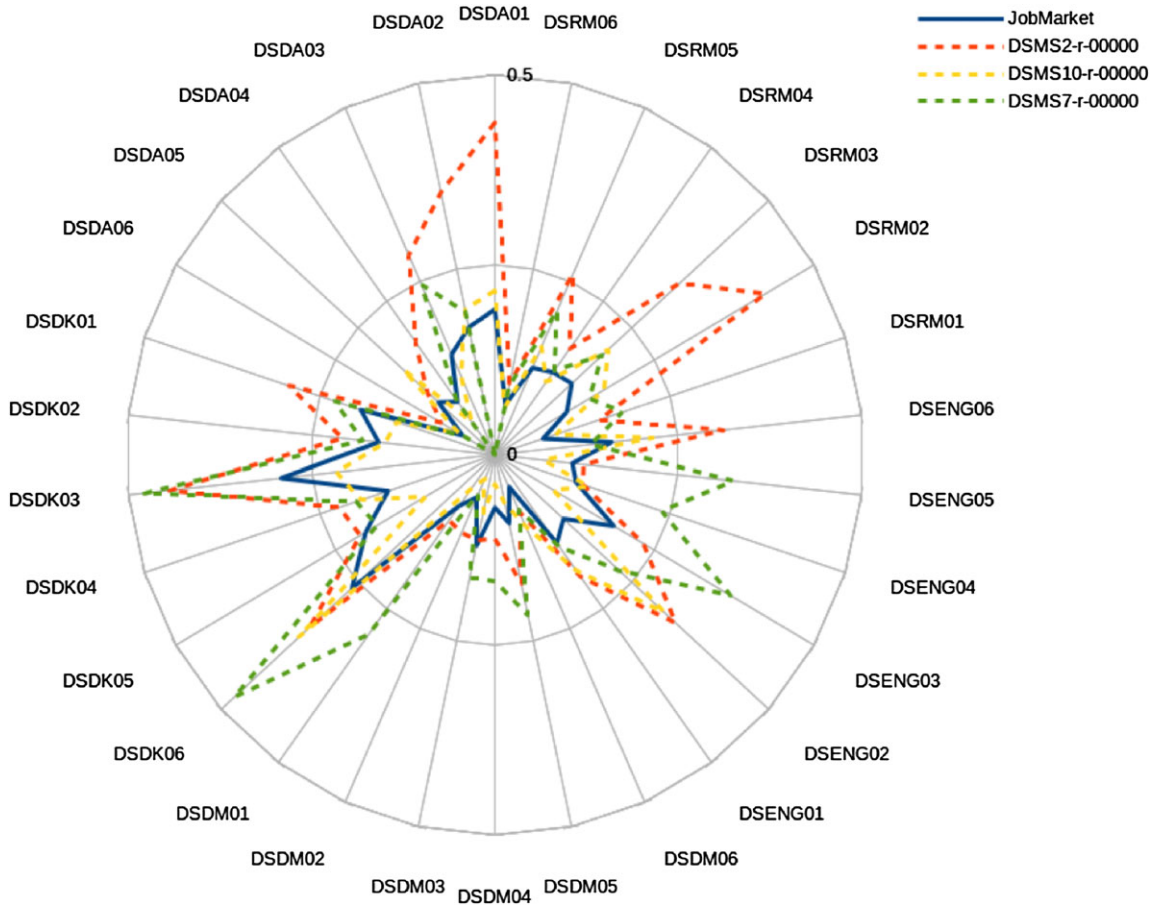

FIGURE 10 Detailed profile of job market and individual courses for each of the 30 competence groups (definition of competence group see the work of EDISON[28])

profiles, etc. The Data Science competencies are divided into five groups, ie, Data Science Analytics (DSDA), Data Science Engineering (DSENG), Data Science Management (DSDM), Data Science Research Methods (DSRM), and Data Science Domain Knowledge (DSDK).[28]

To compare Data Science documents using EDSF, we have implemented the Data processing pipeline described in Section 3. The E-CO-2 service is available as a Docker image that can be downloaded from the E-CO-2 GitHub repository.[23] The Github repository includes the data sets that have been used to generate the results shown in Figures 9 and 10. To construct each of the competence groups vectors (DSDA01, DSRM06, etc), we used the EDSF where it defines each compliance group. From each compliance group, we manually extracted keywords and representative terms. For example, for DSDA01, we extracted terms such as unsupervised machine learning or Data Mining. Next, for each of these keywords and terms, we found the relevant Wikipedia page on which we perform a term extraction and measure the TF-IDF, which allows us to construct a vector for each competence group.

Scenario 1: A student wants to select the program that will help him to develop specific competencies to work on a given Data Science professional profile.

The students provide an up-to-date CV and list of jobs she/he is interested in and E-CO-2 service will return a polar presentation showing which job fits better with the current CV; it will also show the competencies missing to match the other job Ads; the students will have then the choice either to follow courses or trainings to develop the missing skills or apply for the job that fit his current CV. A similar scenario is when the HR department has multiple candidates and wants to shortlist the applicants to select only the two or three most relevant to a given job opening in the company. Figure 9 shows a CV profile compared to three different job Ads. It is clear from the candidate CV fits better the two job Ads and is missing one specific skill for the third one.

Scenario 2: A Course or training developer wants to check whether her/his course or training is still relevant for students who want to develop competencies for a given Data Science professional profile. In addition, she/he wants to compare her/his courses. This is a simplified scenario of a program director that wants to select a set of courses and trainings to create a curriculum targeting given Data Science professional profile. The course developer provides a description of the course and E-CO-2 service provide a measure about the distance of that course content from the job market requests; if more courses or trainings are available, then the course will be also compared to other similar courses.

Figure 10 shows how the profile of three courses compare to the job market profile derived from the data set containing 1000 job Ads (blue).

Using the language-processing technique, we have managed to find similarities between job Ads, CV, and course description using EDISON Data Science competencies. Over time, these lists of competencies are likely to become out of date and will not reflect anymore the need of the job market. The E-CO-2 analyzer can be used to align definitions and terminology used within EDISON with the current status of the job market. As described in previous sections, the E-CO-2 analyzer compares the similarity between the context vector and the word vector. Assuming that the word vector contains a set of terms extracted from a large and representative data set, the terms that appear in the word vector but not in the context vector should be used to better align the descriptions and terminology within EDISON. At the same time, the terms that appear in the context vector but not in the word vector are redundant and therefore need to be reconsidered.

Therefore, if C is the set of terms used in EDSF and W the set of terms used within the job market, then using the E-CO-2 analyzer, we will obtain the set difference, $M = C/W$, to obtain the terms we should include in EDSF. Similarly, by taking the set difference, $R = W/C$, we can obtain the redundant terms used within in EDSF. After obtaining the missing and redundant terms, we use one of the statistical metrics mentioned earlier to rank the terms. This way more importance can be given to the top terms where we will have more impact.

5 | RELATED WORK

Solutions proposed to address job Ads to CV matching often require some extras to work properly like social media data or extra input from the CV owners. A Bilateral recommendation system was developed to improve the match between people and jobs.[29] It is based on task-related and social aspects of human and social capital or person-environment fit. The approach considers two dimensions, ie, (1) matching individuals to task (fitness of individual to job or P-J) and (2) individual to another individual's (fitness of individual to the working environment group, vocational, and organization). Two components are developed, ie, CV-recommender and job-recommender. Both recommenders are based on a probabilistic hybrid recommendation engine based on a latent aspect model that tries to derive individual preferences as a combination of preference factors. For the approach to work, not only CVs and job description but also CV owners were asked to rank the job based on their preference. Obviously, this approach does not scale to a large number of jobs and a large number of CVs. Web Finder is a web application with the aim to match CVs based on skills with respect to a given job. CVs are ranked by comparing the skills from the resume to the skills required in the job description. Web Finder is based on Named Entity Recognition (NER) approach; it uses a statistical classifier to identify named entities; a classifier is trained annotated training set, which contains at least 15000 sentences to work properly.[30] A more recent trend in job-CV matching emerged since 2012, where the information that is extracted from CVs and job description document is complemented by social media data. Bollinger[31] demonstrated that the addition of social media and external data improves the classification accuracy dramatically in terms of identifying the most qualified candidates. Schmitt and Caillou[32] used a deep neural network to match the collaborative filtering representation properties. The aforementioned authors used information inferred from the interactions between job recruiter and job seeks differs from the information that could be extracted from CVs or job announcements. Other similar approaches combine information collected from a LinkedIn account with information from applicant's blogs to match a person to derive the candidate's relevance score for the applied position.[33] E-Gen[34]

is a Natural Language Processing and Information Retrieval system analyses the candidate's answers, which are composed of the cover letter and the CV and computes a relevant ranking of the candidate's application. Comparing to all the CV-matching approaches, we propose a method that could be tuned to specific by selecting the reference against which can compare CVs and job Ads. Many commercial solutions like "Search and Match,"[35] DaXtra search,[36] Match,[37] and Rchilli CV automation[38] based on proprietary solution such as Aspire content Processing platform[34] and aim at automated CV/Resume matching where CVs and job Ads are vectorized in a multidimensional space including job titles, skills, experience, qualification, location, salary range, industry sector, etc. Unfortunately, there is not a lot of information about these tools available to allow a deeper analysis. However, from the dimensions considered for matching, it is clear that these tools are generic and do not consider the domain-specific needs. Our approach focuses on Data Science jobs because it is still not well defined both recruiters and applicants are using different terminology to point to the same or similar skills. Our approach offers a reference based on the Edison Data Science Framework to match skills but also help to identify and rank the skills based on the competence groups relevant to different Data Science job profile. Beyond CV and job Ads matching our approach can help both recruiters, job seekers, and trainers to improve the description and the content of CVs, Job Ads, and trainings to better achieve their respective goals.

6 | CONCLUSIONS

This work has shown that it is possible to develop semi-automatic service that analyzing the huge amount of available data (courses, website, books, open positions, job advertisement, etc) could establish a direct correlation between the skills and competencies the business sector demand and courses or training that education sector offer. This correlation may help both sides to be more (and more quickly) aligned. At the same time, the same conceptual design for the service could support the competency evaluation and gap identification for trainees and students, ie, each of them will be empowered with an instrument that let them taking full control of their educational and career development paths, adapting the course of studies on the quickly changing market landscape and the variability in their personal interests.

The proposed approach may be improved toward two directions, ie, increase the quality and accuracy of the results and speed up the performance of the overall system, to enable quasi-real-time execution.

Some of the extracted terms are not accurately disambiguated due to insufficient information provided by online dictionaries. Combining more resources may help in a more accurate disambiguation and a more complete picture of the skills required. A potential solution to this problem could be the use of existing taxonomies (ie, ACM taxonomy) to compare and validate the proposed results. Taxonomy alignment is a challenging issue, which has to address heterogeneities between different taxonomies, ie, one aspect relates to the lexical heterogeneity, where classes of taxonomies may be semantically equivalent while the terms used for expressing them might differ.[39] Another aspect is their structural heterogeneity, where relationships between concepts of taxonomy are different from those of another one.[40] Using semantic similarity measures, which involve statistical and linguistic approaches, we may be able to identify similar concepts from existing taxonomies.

In order to speed up the overall system execution and willing to pursue some historical data analysis about the variability of skills, new (big data) architecture needs to be considered. As such, an initial evaluation of the Lambda Architecture[41] has been done. Taking into account the amount of data to be evaluated will increase rapidly over the years (both for the increase of produce new data and the need to store historical ones); an approach based on the dual nature of Lambda Architecture may help in addressing both challenges. The successful implementation of this approach may lead to the creation of quasi-real-time observatory about competence and skills not only in Data Science but also in other fields.

ACKNOWLEDGMENTS

This work has been developed in the context of the EDISON project that received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement 675419. The authors would like to thank Michele Sparamonti, University of Perugia, for contribution to the MapReduce implementation and the experimentation of an extension of E-CO-2 for real-time data using a Lambda Architecture.

ORCID

Adam S.Z. Belloum https://orcid.org/0000-0001-6306-6937

REFERENCES

1. Manieri A, Brewer S, Riestra R, et al. Data science professional uncovered: how the EDISON project will contribute to a widely accepted profile for data scientists. Paper presented at: 2015 IEEE 7th International Conference on Cloud Computing Technology and Science (CloudCom); 2015; Vancouver, Canada.
2. EDISON. Data Science educational program inventory. http://edison-project.eu/university-programs-list
3. Wiktorski T, Demchenko Y, Belloum A, Shirazi A. Quantitative and qualitative analysis of current data science programs from perspective of data science competence groups and framework. Paper presented at: 2016 IEEE International Conference on Cloud Computing Technology and Science (CloudCom); 2016; Luxembourg.
4. National Institute of Standards and Technology. NIST big data interoperability framework: Volume 6, reference architecture. Gaithersburg, MD: National Institute of Standards and Technology; 2015. https://doi.org/10.6028/NIST.SP.1500-6

5. Maynard D, Li Y, Peters W. NLP techniques for term extraction and ontology population. In: *Ontology Learning and Population: Bridging the Gap Between Text and Knowledge*. Amsterdam, The Netherlands: IOS Press; 2008:107-127.
6. Pazienza MT. A domain-specific terminology-extraction system. *Terminol Int J Theor Appl Issues Specialized Commun*. 1998;5(2):183-201.
7. Buitelaar P, Cimiano P, Magnini B. *Ontology Learning From Text: Methods, Evaluation and Applications*. Amsterdam, The Netherlands: IOS Press; 2005. *Frontiers in artificial intelligence and applications*; vol. 123.
8. Hatzivassiloglou V, Gravano L, Maganti A. An investigation of linguistic features and clustering algorithms for topical document clustering. In: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR); 2000; Athens, Greece.
9. Frantzi KT, Ananiadou S, Hideki M. Automatic recognition of multi-word terms: the C-value/NC-value method. *Int J Digit Libr*. 2000;3(2):115-130.
10. Mihalcea R. Word sense disambiguation. In: *Encyclopedia of Machine Learning*. New York, NY: Springer Science+Business Media LLC; 2010:1027-1030.
11. Yarowsky D. Unsupervised word sense disambiguation rivaling supervised methods. In: Proceedings of the 33rd annual meeting on Association for Computational Linguistics (ACL); 1995; Cambridge, MA.
12. Navigli R, Ponzetto SP. BabelNet: building a very large multilingual semantic network. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL); 2010; Uppsala, Sweden.
13. Jain AK. Data clustering: 50 years beyond K-means. *Pattern Recogn Lett*. 2010;31(8):651-666. Award winning papers from the 19th International Conference on Pattern Recognition (ICPR).
14. Johnson SC. Hierarchical clustering schemes. *Psychometrika*. 1967;32(3):241-254.
15. Shrivastava V, Arya P. A study of various clustering algorithms on retail sales data. *Int J Comput Commun Netw*. 2012;1(2):68-74. http://warse.org/pdfs/ijccn04122012.pdf
16. Vadeyar DA, Yogish HK. Farthest first clustering in links reorganization. *Int J Web Semantic Technol*. 2014;5(3):17-24.
17. Kersten PR, Lee JS, Ainsworth TL. Unsupervised classification of polarimetric synthetic aperture radar images using fuzzy clustering and EM clustering. *IEEE Trans Geosci Remote Sens*. 2005;43(3):519-527.
18. Hee K, Tolle K, Zicari R, Manieri A. Tailored data science education using gamification. In: Proceedings of the 8th IEEE International Conference and Workshops on Cloud Computing Technology and Science (CloudCom); 2016; Luxembourg.
19. Mayberry T, Blass EO, Chan AH. PIRMAP: efficient private information retrieval for MapReduce. In: Sadeghi AR, ed. *Financial Cryptography and Data Security: 17th International Conference, FC 2013, Okinawa, Japan, April 1-5, 2013, Revised Selected Papers*. Berlin, Germany: Springer-Verlag Berlin Heidelberg; 2016:371-385.
20. Hiemstra D, Hauff C. MapReduce for information retrieval evaluation: "Let's Quickly Test This on 12 TB of Data". In: Agosti M, Ferro N, Peters C, de Rijke M, Smeaton A, eds. *Multilingual and Multimodal Information Access Evaluation: International Conference of the Cross-Language Evaluation Forum, CLEF 2010, Padua, Italy, September 20-23, 2010. Proceedings*. Heidelberg, Germany: Springer-Verlag Heidelberg; 2010:64-69. *Lecture Notes in Computer Science*; vol. 6360.
21. Lin J, Dyer C. *Data-Intensive Text Processing with MapReduce*. San Rafael, CA: Morgan & Claypool Publishers; 2010.
22. Hiemstra D, Hauff C. MIREX: MapReduce information retrieval experiments. 2010. https://arxiv.org/abs/1004.4489v1
23. Skoulozis S. E-CO-2. 2016. https://github.com/skoulouzis/E-CO-2/releases/
24. NIST Big Data Public Working Group Definitions and Taxonomies Subgroup. NIST big data interoperability framework: Volume 1, Definitions. Gaithersburg, MD: National Institute of Standards and Technology; 2015. http://dx.doi.org/10.6028/NIST.SP.1500-1
25. Reali G, Femminella M, Manieri A, Nucci FS. Teaching domain-driven data science: public-private co-creation of market-driven certificate. In: Proceedings of the 2015 IEEE 7th International Conference on Cloud Computing Technology and Science (CloudCom); 2015; Vancouver, Canada.
26. Mukala, P, Buijs J, Leemans M, van der Aalst W. Learning analytics on coursera event data: a process mining approach. In: Proceedings of the 5th International Symposium on Data-driven Process Discovery and Analysis (SIMPDA); 2015; Vienna, Austria.
27. Skoulozis S. E-CO-2 the taxonomy, GitHub repository. https://github.com/skoulouzis/E-CO-2/releases/download/v0.0.1/taxonomy.rdf
28. EDISON. EDSF part 1. Data science competence framework (CF-DS) release 2. Version v0.8. 2017. http://edison-project.eu/data-science-competence-framework-cf-ds
29. Malinowski J, Keim T, Wendt O, Weitzel T. Matching people and jobs: a bilateral recommendation approach. In: Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS); 2006; Kauia, HI. https://ieeexplore.ieee.org/abstract/document/1579569/
30. Kalva TR. *Skill Finder: Automated Job-Resume Matching System*. Logan, UT: Utah State University; 2013. All Graduate Plan B and other Reports. https://digitalcommons.usu.edu/cgi/viewcontent.cgi?article=1343&context=gradreports
31. Bollinger J, Hardtke D, Martin B. Using social data for resume job matching. In: Proceedings of the 2012 Workshop on Data-driven User Behavioral Modelling and Mining from Social Media (DUBMMSM); 2012; Maui, HI.
32. Schmitt T, Caillou P, Sebag M. Matching jobs and resumes: a deep collaborative filtering task. Paper presented at: 2nd Global Conference on Artificial Intelligence (GCAI), Volume 41; 2016; Berlin, Germany. hal-01378589.
33. Faliagka E, Ramantas K, Tsakalidis A, Tzimas G. Application of machine learning algorithms to an online recruitment system. Paper presented at: The Seventh International Conference on Internet and Web Applications and Services (ICIW); 2012; Stuttgart, Germany.
34. Kessler R, Béchet N, Roche M, Torres-Moreno J-M, El-Bèze M. A hybrid approach to managing job offers and candidates. *Inf Process Manag*. 2012;48(6):1124-1135.
35. In Staffing and Recruiting, Time Really IS Money Search & Match Solutions for Recruiters and HR. Retrieved February 2019. https://www.searchtechnologies.com/search-and-match-solutions-recruitment
36. DaXtra Search, Retrieved February 12, 2019, from. http://www.daxtra.com/resume-database-software/resume-matching-software/
37. Accelerate the Match! between candidates and jobs Retrieved Feruary 2019, from https://www.textkernel.com/hr-software/semantic-matching
38. RChilli Semantic Search and Match, Retrieved February 2019, from https://www.rchilli.com/solutions/semantic-search

39. Menczer F. Lexical and semantic clustering by web links. *J Am Soc Inf Sci Technol*. 2004;55(14):1261-1269.
40. Vetere G, Lenzerini M. Models for semantic interoperability in service-oriented architectures. *IBM Syst J*. 2005;44(4):887-903.
41. Bär N. *Investigating the Lambda Architecture* [master's thesis]. Zürich, Switzerland: University of Zurich; 2014.

How to cite this article: Belloum ASZ, Koulouzis S, Wiktorski T, Manieri A. Bridging the demand and the offer in data science. *Concurrency Computat Pract Exper*. 2019;31:e5200. https://doi.org/10.1002/cpe.5200