Session 3
Knowledge Discovery in Databases (KDD) for LMI

Alessandro Vaccarino – Fabio Mercorio

Big Data for Labour Market Information – focus on data from online job vacancies – training workshop
Milan, 21-22 November 2019
Topics

1. KDD steps for game changing in LMI
2. What you can do with LMI processed? [examples]
3. Issues, limitations and challenges
Is Big Data a game changer in the field of labour market?
Three main Labour Market Sources can support LM Intelligence

(1) Statistical sources
(2) Administrative sources
(3) Web Sources (Big Data 4 LMI)
Quo vadis Labour Market?

LM CHALLENGING FACTORS
1. Skills Evolution
2. New Emerging Occupations
3. Job Automatisation/Replacement
4. Mobility

LM NEEDS
1. Updated information (near-real-time)
2. Data driven decisions (let data speak)
3. Prediction can be done to anticipate trends

Knowledge becomes crucial to support different LM actors and policy makers in understanding LM dynamics and trends.
**Web Labour Market Scenario**

- Job Vacancies frequently posted on specialised Web sources
- Hidden informative power about labour market dynamics
- Heterogeneous sources and different lexicons used in job vacancy texts
- Info about skills, industry sectors, territory, etc expressed as raw text within vacancies

**Stakeholder Needs Identified**

- Near real-time labour market analysis
- Labour Market Occupations/Skills Trend Monitoring
- Evaluate/compare International LM for fact-based decision making
- Analyse LM according to the identified dimensions

**Proposed Research Actions**

- Data scraping from selected sources
- Job vacancy classification via machine-learning
- Multi Language support through the use of Standard Taxonomies
- Query the resulting knowledge base over the identified dimensions
<table>
<thead>
<tr>
<th><strong>LM Source Type</strong></th>
<th><strong>Data Type</strong></th>
<th><strong>Generation Rate</strong></th>
<th><strong>Data Model Paradigm</strong></th>
<th><strong>Quality</strong></th>
<th><strong>Coverage</strong></th>
<th><strong>Analysis Paradigm</strong></th>
<th><strong>Believability</strong></th>
<th><strong>Value</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Statistical</td>
<td>Structured</td>
<td>Periodically</td>
<td>Relational</td>
<td>Owner’s responsibility</td>
<td>Owner’s responsibility</td>
<td>Top Down &amp; Model Based</td>
<td>Owner’s responsibility</td>
<td>intrinsic</td>
</tr>
<tr>
<td>Administrative</td>
<td>Structured or Semi-structured</td>
<td>Periodically</td>
<td>Relational</td>
<td>Owner’s responsibility &amp; User’s responsibility</td>
<td>Top Down &amp; Model Based</td>
<td>Owner’s responsibility &amp; User’s responsibility</td>
<td>intrinsic</td>
<td></td>
</tr>
<tr>
<td>Web</td>
<td>Structured, Semi-structured or Unstructured</td>
<td>Near-real-time or real-time</td>
<td>Relational and Non Relational (NoSQL)</td>
<td>User’s responsibility</td>
<td>User’s responsibility</td>
<td>Bottom up &amp; Data Driven</td>
<td>User’s responsibility</td>
<td>extrinsic</td>
</tr>
</tbody>
</table>
Table 2 Most significant limitations of Big Data architectures

<table>
<thead>
<tr>
<th>Issue (most significant)</th>
<th>Caused by</th>
<th>Conceptual Blocks of Big Data Architectures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schema-free data are out: only structured data sources can be manipulated. Roughly,</td>
<td>Variety</td>
<td>Data ingestion; NoSQL models;</td>
</tr>
<tr>
<td>this means that only data that obey a rigid, well-defined data model can be handled,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>to the exclusion of all “unstructured” data, such as free text, comments and Web</td>
<td></td>
<td></td>
</tr>
<tr>
<td>content in general.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>No adaptability to change: the addition of a new source requires the whole process</td>
<td>Variety, Velocity</td>
<td>Data lake</td>
</tr>
<tr>
<td>to change, and this makes it difficult to scale the architecture over multiple (albeit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>structured) sources.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rigid ETL: the procedures that transform content from source formats to target formats</td>
<td>Variety</td>
<td>Schema free; data-driven approach (bottom-up rather than</td>
</tr>
<tr>
<td>have to be precisely written to fit the desired data structure (e.g., data warehouse).</td>
<td></td>
<td>top-down)</td>
</tr>
<tr>
<td>Time consuming: the larger the volume of data to be processed, the greater the time</td>
<td>Volume, Variety, Velocity</td>
<td>Scale-out rather than scale-up</td>
</tr>
<tr>
<td>needed to complete the process. ETL procedures are usually high time and memory</td>
<td></td>
<td></td>
</tr>
<tr>
<td>consumers, as they need to “scan” all the data sources at any time to transform</td>
<td></td>
<td></td>
</tr>
<tr>
<td>source data.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
How to deal with OJVs at scale?
Web Job Vacancy example

**Job Title:** Data Scientist.

**Description:** We’re looking for a talented Computer Scientist to join our growing development team. Your expertise in data will help us take this to the next level. You will be responsible for identifying opportunities to further improve how we connect recruiters with jobseekers, and designing and implementing solutions. [...] **Required skills and experience:**

- SQL and relational databases;
- Data analysis with R (or Matlab);
- Processing large data sets with MapReduce and Hadoop);
- Real time analytics with Spark, Storm or similar;
- Machine Learning;
- Natural Language Processing (NLP) and text mining;
- Development in C++, Python, Perl;
- Experience with search engines e.g. Lucene/Solr or ElasticSearch advantageous
Web Job Vacancy example

**Job Title:** Data Scientist.

**Description:** We’re looking for a talented Computer Scientist to join our growing development team. Your expertise in data will help us take this to the next level. You will be responsible for identifying opportunities to further improve how we connect recruiters with jobseekers, and designing and implementing solutions. […] **Required skills and experience:**

- SQL and relational databases;
- Data analysis with R (or Matlab);
- Processing large data sets with MapReduce and Hadoop);
- Real time analytics with Spark, Storm or similar;
- Machine Learning;
- Natural Language Processing (NLP) and text mining;
- Development in C++, Python, Perl;
- Experience with search engines e.g. Lucene/Solr or ElasticSearch advantageous
The Process (KDD, Fayyad 1997)
The Process (KDD 4 LMI)

1. Source selection, Raking and data Ingestion
2. Data pre processing, Transformation and cleansing
3. Classification
4. Skills extraction
5. Analysis and Data Visualisation
Which expertises are needed to build a LMI system?
Five steps for turning Big Data into LMI

1. Source selection, Raking and data Ingestion
2. Data pre processing, Transformation and cleansing
3. Classification
4. Skills extraction
5. Analysis and Data Visualisation

Professionals involved in the different steps of the LMI project:

- Statisticians
- Data Engineer and Data scientists
- LM Domain experts – economists and sociologists

These expertise owned by such professionals show the necessity for a multidisciplinary approach in developing a LMI project.
Five steps for turning Big Data into LMI

1. Source selection, Raking and data Ingestion
2. Data pre processing, Transformation and cleansing
3. Classification
4. Skills extraction
5. Analysis and Data Visualisation

This step includes text processing, NLP, data cleaning, denoising, and deduplication

**Statisticians**: Identify measures of data quality, data distribution and significance

**Computer scientist**: Guarantee pre—processing to scale over million items

**LM Domain experts**: How do we identify LM domain synonyms that help in improving data accuracy? How do we identify criteria that characterise missing values and duplicates?
Five steps for turning Big Data into LMI

1. Source selection, Raking and data Ingestion
2. Data processing, Transformation and cleansing
3. Classification
4. Skills extraction
5. Analysis and Data Visualisation

It includes data reduction and projection, which aim at identifying a unified model.

**Statisticians**: Measure completeness of unified data model

**Computer scientist**: Guarantee transformation from raw data into target data at scale

**LM Domain experts**: How do we identify the destination data format and taxonomy?
Five steps for turning Big Data into LMI

1. Data pre-processing, Transformation and cleansing
   - Source selection, Raking and data Ingestion

2. Skills extraction
   -  Classification

3. Classification
   - Source selection, Raking and data Ingestion

4. Analysis and Data Visualisation
   - Result evaluation

5. LM Domain experts: Which skills should be selected and which should be discarded?

Statisticians & Computer scientists: Algorithm identification, parameters tuning, implementation

identify appropriate AI algorithms (e.g., classification, prediction, regression, clustering, information filtering), by searching for patterns of interest in a particular representational form, based on the purpose of the analysis.
Five steps for turning Big Data into LMI

1. Source selection, Raking and data Ingestion
   *Data pre processing, Transformation and cleansing*

2. Classification

3. Skills extraction

4. Analysis and Data Visualisation

Employ visual paradigms to visually represent the knowledge obtained, depending on the user's objectives. In the LMI context, it means taking into account the user's ability to understand the data and their main goal.

**Statisticians & Computer scientists:** Identify visualisation and narrative paradigm and implement it.

**LM Domain:**
- How do we deliver appropriate knowledge according to stakeholder needs?
- How do we identify visual navigation paths for each stakeholder?
- How do we retrieve feedback (if any) from LM users?
- How do we put LM knowledge into business?
ITALIAN Real-Time Labour Market Monitor

Labour Market Intelligence

Big Data
AI
Eco+ Stat

OJV since 2013 – 4M+ vacancies unique
EUROPEAN Real-Time Labour Market Monitor

European Union Member States

Labour Market Intelligence

Big Data
AI
Eco+ Stat

28 EU Countries – 32 Languages – more than 6M unique vacancies per month
What you can do with LMI processed?

1. Occupation and Skill Discovery
2. Soft/Digital/Hard Skill Rates
3. New Emerging Occupations
4. Taxonomy Extension
5. Explain how ML works behind the scenes to humans
Occupation and Skill Discovery

Focus on occupations and skills requested by the online-LM
Live demo: Territorial dimension

VIDEOS
Soft/Digital/Hard Skill Rates

How to estimate the impact of digitalization within occupations?
Compute Skill Rates

**Goal:** Estimate the pervasiveness of ICT in both ICT and not ICT-related jobs

**Idea:** Exploit the informative power of Classified OJV for computing The Digita Skill Rate (DSR), Soft skill rate and Hard non digital Skill Rate

DSR estimates the incidence of digital skills in a single profession and comes from observing the pervasiveness of digital skills in all professions whether they are related to the ICT world or not.
Compute Skills Rates

Demand of digital, specialist and soft skills - by sector

Industria
- 13% DSR 2017
- 59% HARD Non Digital
- 28% Soft 2017
- 11% DSR 2014

Servizi
- 14% DSR 2017
- 54% HARD Non Digital
- 32% Soft 2017
- 12% DSR 2014

Commercio
- 9% DSR 2017
- 56% HARD Non Digital
- 35% Soft 2017
- 9% DSR 2014

Ad hoc analyses at different level of granularity, here focusing on the «sector»....

Credits to WollyBI: a trademark of TabulaeX
Compute Skills Rates

- **Applied and Management Skills** = ability to use tools and software to manage both operational and decisional processes
- **ICT Techniques Skill** = very specialized on solutions, platforms and programming languages
- **Basic Skill** = for everyday use of basic IT tools
- **Information Brokerage Skill** = for the use of IT tools aimed at corporate communication

... and more, looking at each occupation...

Credits to WollyBI: a trademark of TabulaeX
Compute Skills Rates [ESCO source WollyBI skills + novel]

<table>
<thead>
<tr>
<th>Occupation</th>
<th>Applied and Management Skills</th>
<th>ICT techniques</th>
<th>Information Brokerage</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Database usage</td>
<td>Front-end Website implementation</td>
<td>Graphic Software Usage</td>
</tr>
<tr>
<td></td>
<td>Programs for draughts man</td>
<td>Web programming</td>
<td>SW markup usage</td>
</tr>
<tr>
<td>Graphic and multimedia designers</td>
<td>25</td>
<td>45</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>35</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

... and more, looking at elementary skills

Credits to WollyBI: a trademark of TabulaeX
New Emerging Occupations on the basis of skill (dis)similarities
Detecting new emerging occupations through AI

1. Classify OJV over ISCO-iv digit
2. Build-up several vector-space representations of words (occupations and skills) to catch lexicon similarities between OJVs
3. Compute similarities between known terms (occupations and skills) and new ones
4. Suggest new potential occupations for Human-AI validation
(Some) New Emerging Occupations

- Data Scientist
- Cloud Computing
- Cyber Security Expert
- Business Intelligence Analyst
- Big Data Analyst
- Social Media Marketing

9,000 Web Job Vacancies collected related to (some) new emerging occupations (above) between Jan-2014 and August-2019
DATA SCIENTIST – 1,7k vacancies 2014-2019

HARD SKILLS
- **48%** MATH & STAT
  - Data Analysis, Statistical Learning
  - SAS, R
  - SAP & SPSS*
- **39%** COMPUTING
  - SQL, Python, Hadoop
  - BI, Machine-Learning
  - Data Integration
- **13%** BUS & ADM
  - Public Relations
  - Management
  - Clients Relations Management

SOFT SKILLS
- **38%** BEHAVIOURS
- **29%** FOREIGN LANGUAGES
- **13%** PROBLE SOLVING
- **7%** COLLABORATION
- **5%** LEADERSHIP

Variation 2019 vs 2018: +31%
Variation 2019 vs 2017: +149%
SOCIAL MEDIA SPECIALIST – 0.4k vacancies 2014-2019

HARD SKILLS

55% BUS & ADM
- Management
- Public Relations
- Marketing

39% COMPUTING
- Adobe Photoshop & HTML5
- Google Analytics & AdWords*
- CMS (Content Management System)*

16% MATH & STAT
- Data Analysis

SOFT SKILLS

42% BEHAVIOURS

35% FOREIGN LANGUAGES

7% PROBLEM SOLVING
- CREATIVE & ENTREPRENEURIAL THINKING
- INFORMATION & COMMUNICATION

Variation 2019 vs 2018: +105%
Variation 2019 vs 2017: +123%
New Occupations can be compared against traditional ones using different taxonomies (the eCF Competence Framework in such a case)
Taxonomy Extension: How to improve skills/occupations taxonomies through semantic similarities within OJVs?
Main Idea

**KEY Questions: How to...**

1. *maintain the taxonomy up-to-date* with labour market expectation and lexicon?
2. *enrich the taxonomy with those mentions* to corresponding entities in the taxonomy?
3. *Estimate similarities* between all taxonomy entities?
4. *Estimate the relevance of taxonomy entities?*
Can we use big data for skills anticipation and matching?

Taxonomy Extension: Improve skills/occupations taxonomy through semantic similarities within OJVs. Just compute the distance in a vector space.
Compare different Web labour markets [IT, UK and DE here] to perform Skill Gap Analysis using the taxonomy as a baseline.

Question: “starting from a given occupation of the ISCO taxonomy for the ITA labour market, what is the occupation in the UK labour market whose requested skills with a better fit?”

Skills associated to “Web Technician” in ITA are more similar to a “Web and Multimedia Developer” in UK.
Build-up an LMI Ontology

VIDEOS
## Compare Different Countries (Web Technician)

<table>
<thead>
<tr>
<th>Nazione</th>
<th>Top 10 skill</th>
<th>IT</th>
<th>tf-idf IT</th>
<th>tf-idf UK</th>
<th>tf-idf DE</th>
</tr>
</thead>
<tbody>
<tr>
<td>IT</td>
<td>web programming</td>
<td>100.0</td>
<td>38.0</td>
<td>100.0</td>
<td></td>
</tr>
<tr>
<td>IT</td>
<td>analyse software specifications</td>
<td>95.18</td>
<td>36.84</td>
<td>21.72</td>
<td></td>
</tr>
<tr>
<td>IT</td>
<td>computer programming</td>
<td>89.17</td>
<td>48.82</td>
<td>59.35</td>
<td></td>
</tr>
<tr>
<td>UK</td>
<td>Marketing</td>
<td>39.69</td>
<td>100.0</td>
<td>56.56</td>
<td></td>
</tr>
<tr>
<td>UK</td>
<td>online analytical processing</td>
<td>0</td>
<td>94.39</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>UK</td>
<td>adapt to change</td>
<td>69.21</td>
<td>87.21</td>
<td>79.47</td>
<td></td>
</tr>
<tr>
<td>DE</td>
<td>web programming</td>
<td>100.0</td>
<td>38.0</td>
<td>100.0</td>
<td></td>
</tr>
<tr>
<td>DE</td>
<td>use markup languages</td>
<td>80.47</td>
<td>34.03</td>
<td>82.97</td>
<td></td>
</tr>
<tr>
<td>DE</td>
<td>adapt to change</td>
<td>69.21</td>
<td>87.21</td>
<td>79.47</td>
<td></td>
</tr>
</tbody>
</table>
Improve classification through visual explanations: eXplain and Validate

Explainable Labour Market Intelligence (XLMI)

**Goal.** To improve the believability of the analysis and results provided to the final users by explaining the behaviour of AI algorithms used to produce them.

**Idea.** ML algorithms act like a black-box, and there is no way to guess the reason behind a decision. eXplainable AI (XAI - launched by DARPA in 2016) aims at building a new generation of ML systems able to explain their decision in a human manner.

**Benefits.** Improved ability in understanding and monitoring the classification process (even in case this is a complex pipeline of algorithms) and in identifying misclassification to improve both the accuracy and the system understanding.
Why do we need explanations?

No way to guarantee (or at least to make evidence) to users that a system learned the “right model”, but just it learned the “model right”

Explain (in a human-readable format) why a system classified an item on a class would make it more reliable and trustable to the final user

Be able to understand and isolate what forced a system to predict an outcome can be used in a human-in-the-loop validation process to improve:

- **Transparency** to the end users, that have to make decisions on the basis of your outcomes;
- **Believability and utility** of the deployed system;
- **Accuracy** of classification;
- And more (see after)
RA2: Improve classification through visual explanations: eXplain and Validate

Proposal. Use OJV data to text XLMI algorithms

Potential Stakeholders. National associations of Information Technology companies

Research Phases
  o Define a global/local model of Explainability
  o Evaluate XAI algorithms
  o Implement the XLMI algorithm
  o Text the defined XLMI algorithm on the OJV data

Research Output.
  o Working Papers;
  o A research proposal to explain WolliBI’s classification features [working].
Focus: eXplain and Validate

1. Why?
2. Which words led you to this decision?
3. Which alternatives you’ve evaluated and how you’ve scored them?
4. Do the system learnt the right model?

Looking for financial manager that would support....
Focus: eXplain and Validate

Might this help?

Looking for financial analyst that would support....

dirigenti nei servizi finanziari

dirigenti nei servizi finanziari

Dirigenti di filiale nei servizi finanziari e assicurativi
Focus: eXplain and Validate

1. Balck Box Model
   - API to get the predicted class
   - Recommendation for improving feature selections

2. Explain via Surrogate Model
   - Post-Pruning to maximise KPI
   - Improve Model's Interpretability

3. Interpretable Model
   - Generate Analytics from the interpretable model

4. Validate
   - Step 1
   - Step 2
   - Step 3
   - Step 4
Focus: eXplain and Validate

Classify each job vacancy according to the system’s classification criteria

Notice: the preprocessing pipeline is unique modulo the language, this means some terms that do not account for the classification task (eg, senior, milan, etc…) are not discarded as they are useful to derive sectors, experiences, etc.
Focus: eXplain and Validate

Use global interpretable models to explain each class (i.e., one of 400+ ESCO occupations)

The result is a tree that shows the system’s behaviours in classifying vacancies belonging to that ESCO occupation

Software Developer →
Focus: eXplain and Validate

Basically, this is a way to see under the hood of the classification system. It can be seen as a set of nested if-then-else that reveal the system criteria.

Civil Engineer

SW developer
Focus: eXplain and Validate

Which is the «best» tree?
- Too many levels → overfitting
- Few levels → underfitting

IDEA: To define a good enough depth, we should take into account the accuracy (performance), the coverage (interpretability) and the number of leaves (complexity). At the end, you have a tree that represents the criteria used by the system for classifying each ESCO occupation.
Moving from **interpretations** to **explanations**

Once the system has been interpreted, we use visualisations to derive explanations in a human-readable manner.
Focus: eXplain and Validate

No “Developer” 97% OUT

No “php web” 0.2% OUT
No “Financial Services” 1% IN
“Financial Services” 0.1% OUT
No “phphtml” 0.9% IN
“phphtml” 0.1% OUT
Challenge

How to deal with representativity issue?
[live]