Towards Labour Market Intelligence through Topic Modelling

Francesco Colace [†], Massimo De Santo [†], Marco Lombardi [†], Fabio Mercorio[‡], Mario Mezzanzanica[‡], Francesco Pascale [†]
[†] DIIn, University of Salerno, Italy
{fcolace — desanto — malombardi — fpascale}@unisa.it
[‡] Dismeq - CRISP Research Centre, University of Milano-Bicocca, Italy
{fabio.mercorio — mario.mezzanzanica }@unimib.it

Abstract

Nowadays, the number of people and companies using the Web to search for and advertise job opportunities is growing apace, making data related to the Web labor market a rich source of information for understanding labor market dynamics and trends. In this paper, the emerging term labor market intelligence *(LMI) refers to the definition of AI algorithms and frameworks that derive useful knowledge for labor market-related activities, by putting AI into the labor market. At the same time, another branch of AI is developing known as Explainable AI (XAI), whose goal is to obtain interpretable models from current (and future) AI algorithms, given that most of them actually act like black boxes, providing no interpretable explanations of their behavior, as in the case of machine learning. In this paper we connect these two approaches, using a graph model obtained through an NLP-based (Natural Language Processing) methodology for classifying job vacancies. We compare the results obtained with those from a European Project in LMI that employs machine learning for the classification task, to show that our approach is effective and promising.*

1. Introduction

Today, the Web is one of the richest sources of data for many services and domains in our daily lives. This is also true of labor market data (aka labor market information), as a growing proportion of labor market demand is posted on Web job portals and aggregators. In this scenario, AI algorithms and frameworks have been recently applied to labor market data in both academic and industrial contexts in order to perform a variety of tasks, such as classification of job vacancies [1], resumes [2], and labor market trend forecasting [3]. This interest in putting AI into the area of the labor market has contributed to the emergence of the term *labor market intelligence* (LMI). Although there is no unified definition of LMI, it can be regarded as the design and realization of AI algorithms and frameworks to analyze labor market data as support for policy planning and decision-making activities [4, 5, 6].

However, these algorithms, as in the case of machine learning, rarely provide explanations that enable users to understand what the system actually learned, and this could affect the reliability of the algorithms' outcomes when they are used by decision makers. This is one of the main drivers behind the rise of a new branch of Artificial Intelligence called eXplainable AI (XAI) [7, 8], whose aim is to make AI algorithms explainable and thus improve dependability and transparency.

This paper presents a framework that exploits topic modelling to address the task of classification in Labor Market Intelligence, providing explanations for the end users. In particular, compared with classical machine-learning and text-classification techniques, this approach relies on the *explainable* nature of topic modelling, enabling the reasons that guided the system through the classification process to be understood. In this respect, the contribution of the proposed framework is twofold:

(i) (i) first, we define a text-classification framework based on the LDA (Latent Dirichlet Allocation) approach and built using [9]. Specifically, the proposed method introduces a graph-based representation of relevant terms mined from text through a probabilistic topic model. This approach provides the correlation value between words, representing the document as a graph rather than as a simple bag of words. In this way the graph can be used as a filter for classifying documents;

(ii) Second, we apply our approach to a real-life problem, framed within an EU Project [10, 11] in the context of Labor Market Intelligence [5]. Specifically, we show that our approach really can compete with classical machine-learning algorithms in terms of classification accuracy, by comparing our results with the ones presented in [1] on the same dataset. Moreover, we show that the explainable nature

URI: https://hdl.handle.net/10125/59962
ISBN: 978-0-9981331-2-6

of our approach *automatically* provides explanations to motivate its behavior in an interpretable and reliable manner, while classical machine-learning approaches do not. Then, to compare the model obtained through our technique with the one learned from machine learning in [1], we use LIME [12], a system that explains the prediction of any classifiers by learning an interpretable model locally around the prediction.

The paper is organized as follows. In Section 2 we give a background on labor market intelligence. In Section 3 we discuss related work, while in Section 4 we introduce the architecture of the proposed approach and some formal settings. An evaluation is provided in Section 5, while Section 6 concludes the paper and describes future research.

2. Background on LMI

In recent years, several forces and factors have dramatically changed the nature and characteristics of the labor market, in both advanced and developing countries. Certain kinds of jobs are disappearing while new jobs are emerging: some of these are simply variants of existing jobs, others are genuinely new jobs that were non-existent until a few years ago.

In such a dynamic scenario, the problem of monitoring, analyzing, and understanding these labor market changes (i) in a timely manner and (ii) at a very fine-grained geographical level, is becoming a significant issue in our daily lives. *Which occupations will grow in the future and where? How can different jobs be compared across countries?* These are some of the questions facing economists and policy makers (see, for example, the recent work by [3] about occupations that might disappear due to digitalization and robotization). Here, a key role is played by big data relating to the labor market (e.g., job advertisements, CVs posted on the Web, etc.). These data need to be collected, organized, and manipulated to permit real-time monitoring and analysis of labor market dynamics and trends (see, for example, [13, 14, 15]).

This is the case of Web job vacancies, which can be seen as documents consisting largely of a pair of texts: a title and a (full job) description. The title summarizes the working position offered by the employer, while the description usually provides the position details, including all the required skills, depending on the employer's preferences. Analysis of Web job vacancies enables labor market needs to be examined without the time lags found in traditional administrative and survey data sources. In addition, a real-time analysis of the Web labor market offers an anytime snapshot of market demand, providing a useful source of information about specific job requirements, contracts on offer, skills required both hard and soft), the industry sector, etc., which are not covered in any surveys. Finally, the information can be classified with standard taxonomies, which act like a *lingua franca* to overcome linguistic boundaries, such as (i) ISCO (The *International Standard Classification of Occupations*) [16], a four-level classification that represents a standardized system for organizing labor market occupations, and (ii) ESCO [17], the multilingual classification system of European Skills, Competences, Qualifications and Occupations, which is the European standard supporting all labor market intelligence over 28 EU languages.[1]

The Relevance of LMI. In 2016, the EU and Eurostat launched the ESSnet Big Data project [18], involving 22 EU member states with the aim of *"integrating big data in the regular production of official statistics, through pilots exploring the potential of selected big data sources and building concrete applications""*. Previously, in 2014, the EU CEDEFOP agency set up to support the development of European Vocational Education and Training launched a call for tenders for development of a system able to collect and classify Web job vacancies from 5 EU countries [10] The rationale behind the project is to turn data extracted from Web job vacancies into knowledge (and thus value) for policy planning and evaluation through fact-based decision making. Given the success of the prototype, a further call for tenders has been launched to develop a Web labor market monitoring system for the whole EU, including 28 EU country members and all 24 languages of the Union [11]. Focusing on business applications, growing numbers of companies are working on the classification and extraction of meaningful information from job vacancies in order to automatize activities in their Human Resources departments. As a result, many commercial skill-matching products have been developed in the last few years, for instance, BurningGlass, Workday, Pluralsight, EmployInsight, and TextKernel (see, for example, [19]). Worthy of mention is the Google Job Search API, a pay-per-use service announced in 2016 for classifying job vacancies through the Google Machine Learning service over O*NET, the US standard occupation taxonomy.

All these approaches highlight the practical significance of LMI as a growing research field involving all the steps of the KDD approach to extract useful knowledge from labor market data.

[1] Basically, the ESCO data model incorporates the entire ISCO structure, and extends it through (i) a further level of fine-grained occupation descriptions and (ii) a taxonomy of skills and competences

3. Related Work

This section surveys the recent literature related to the topics discussed in this paper, which are text classification of unstructured documents, explainable AI and the use of topic modelling for classifying texts.

Text Classification. In the recent literature, *text classification* (TC) has been shown to give good results in extracting knowledge from much real-life web-based data, for instance, data collected from institutional scientific information platforms [20], or microblogs and other social media platforms [21, 22], in many different research areas such as opinion spam detection [23, 24] and sentiment analysis [25, 26], and, recently, job vacancies and labor market information in general [1, 3]. Specifically, text classification has been an active research topic since the early 1990s. It has been defined as "the activity of labeling natural language texts with thematic categories from a predefined set" [27]. Most popular techniques are based on the *machine learning* paradigm, where an automatic text classifier is created by using an inductive process able to learn, from a set of pre-classified documents, the characteristics of the categories of interest. The case in which one category must be assigned to each document is called *single-label* classification, while *multi-label* classification is the case when many categories may be assigned to the same document.

XAI. Explainable Artificial Intelligence (XAI) is an emerging branch of Artificial Intelligence that investigates new machine learning models that are completely explainable in contexts where transparency is important; for instance, when these models are adopted to handle analysis (e.g., classification) or synthesis tasks (e.g., planning, design), as well as mission-critical applications. DARPA recently launched the Explainable AI (XAI) program for the creation of a suite of AI systems able to explain their own behavior, focusing on ML and deep learning techniques. In fact, to date, most ML research models rarely provide explanations/justifications for the outcomes they offer in the tasks they are applied to. The main driver behind the emergence of explainable AI is the need for (i) trust, (ii) interaction, and (iii) transparency, as recently discussed in [28]. This is also the reason behind the interest in XAI applications among academics and industrial communities (see, for example, [29, 12]). With regard to LMI, autonomous decision-making can be framed as a set of economic problems that need to make evident the rationale behind the action suggested, so that the final decision will appear credible to human decision makers. Even in the LMI field, ML-based approaches do not provide any explanations for their outcomes/predictions, and this might prevent the decision maker from considering the analyses that have been performed sufficiently reliable, thus making the overall process ineffective. In essence, the use of XAI algorithms in LMI fits the challenging issue identified by DARPA as *"machine learning problems to construct decision policies for an autonomous system to perform a variety of simulated missions"*[7].

LDA based classification In the last few years, several research studies have been conducted on text classification based on an LDA method [30] [31] [32]. An interesting approach is presented in [33], which proposes an improved short text classification method based on the Latent Dirichlet Allocation topic model and the K-Nearest Neighbor algorithm. This approach helps to give the texts a greater semantic focus and to reduce sparseness. In this way, text comparisons and topic matching are more efficient.

An LDA method was recently proposed for text classification in a semi-supervised manner with representations based on topic models [34]. The proposed method comprises a semi-supervised text classification algorithm based on self-training and a model, which determines parameter settings for any new document collection. Self-training is used to enlarge the small initial labeled set with the help of information from unlabeled data. The idea behind this work is to exploit an LDA-based approach to classify real-world vacancies in an explainable manner.

4. Architecture

In this section we present our architecture, as shown in Fig. 1. Our framework relies on two distinct modules: MGT, and the Classifier.

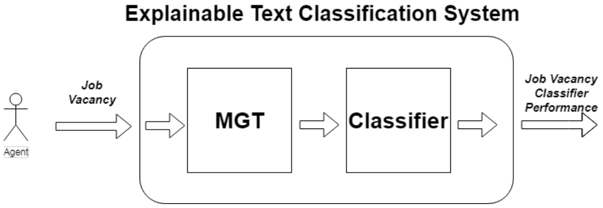
Explainable Text Classification System
Job Vacancy
Agent
MGT
Classifier
Job Vacancy Classifier Performance

Figure 1. System Architecture

MGT represents the Mixed Graph Terms section that provides a graph of terms that will be used by the classification module to generate the opportune masks of terms. This module takes as input raw text and returns a mixed graph of terms with its XML representation, one for each label.

Classifier uses the masks generated by MGT to classify vacancies. Specifically, it receives the XML representation of graph terms as input, divided by labels, and provides in output the masks for classification, a mask for each label.

4.1. Mixed Graph of Terms (mGT)

A mixed Graph of Terms (mGT) is a hierarchical structure composed of two levels of information represented through a directed and an undirected subgraph: the conceptual and word level. Such a graph can be automatically extracted from a document corpus and can be effectively used as a filter to employ in document classification as well as in sentiment extraction problems. Formally, a Mixed Graph of Terms can be defined as a graph. Formally, a Mixed Graph of Terms can be defined as a graph $g = (N, E)$ where:

- $N = \{R, W\}$ is a finite set of nodes, covered by the set $R = \{r_1, ..., r_H\}$ whose elements are the aggregate roots and by the set $W = \{w_1, ..., w_M\}$ containing the aggregates. Aggregate roots can be defined as the words whose occurrence is most implied from the occurrence of all other words in the training corpus. Aggregates are defined as the words most related to aggregate roots from a probabilistic point of view.
- $E = \{E_{RR}, E_{RW}\}$ is a set of edges, covered by the set $E_{RR} = \{e_{r_1 r_2}, ..., e_{r_{H1} r_H}\}$ whose elements are links between aggregate roots, and by the set $E_{RW} = \{e_{r_1 w_1}, ..., e_{r_H w_M}\}$ whose elements are links between aggregate roots and aggregates. As better explained further, two aggregate roots are linked if strongly correlated (in a probabilistic sense):

$$e_{r_i r_j} = \left\{ \begin{array}{ll} 1 & if \psi_{ij} \geq \tau \\ 0 & otherwise \end{array} \right\} \tag{1}$$

Aggregate roots can be also linked to aggregates if a relevant probabilistic correlation is present:

$$e_{r_i w_s} = \left\{ \begin{array}{ll} 1 & if \rho_{ij} \geq \mu_i \\ 0 & otherwise \end{array} \right\} \tag{2}$$

The Feature Extraction module (FE) is represented in Fig. 2. The input of the system is the set of documents:

$$\Omega_r = (d_1, ..., d_M) \tag{3}$$

After the pre-processing phase, which involves tokenization, stopwords filtering and stemming, a Term-Document Matrix is built to feed the Latent Dirichlet Allocation (LDA) [35] module.

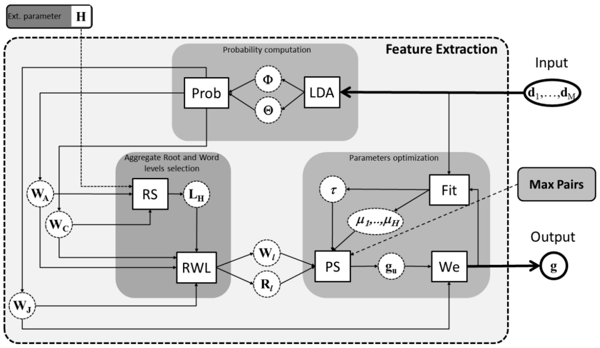

Figure 2. Proposed feature extraction method. A Mixed Graph of Terms g structure is extracted from a corpus of training documents.

The LDA algorithm, assuming that each document is a mixture of a small number of latent topics and each word's creation is attributable to one of the document's topics, provides as output two matrices - $\Theta \; and \; \Phi$ - which express probabilistic relations between topic-document and word-topic respectively. Under particular assumptions [36], LDA module's results can be used to determine: the probability for each word v_i to occur in the corpus $W_A = \{P(v_i)\}$; the conditional probability between word pairs $W_C = \{P(v_i|v_s)\}$; the joint probability between word pairs $W_J = \{P(v_i, v_s)\}$. Details on LDA and probability computation are discussed in [35, 36, 30]. Defining Aggregate roots (AR) as the words whose occurrence is most implied by the occurrence of other words of the corpus, a set of H aggregate roots $r = (r_1, ..., r_H)$ can be determined from W_C :

$$r_i = argmax_{v_i} \prod_{j \neq i} P(v_i|v_j) \tag{4}$$

This phase is referred as Root Selection (RS) in Fig. 2. A weight ψ_{ij} can be defined as a degree of probabilistic correlation between AR pairs: $\psi_{ij} = P(r_i|r_j)$. We define an aggregate as a word v_s having a high probabilistic dependency with an aggregate root r_i. Such a dependency can be expressed through the probabilistic weight $\rho_{is} = P(r_i|v_s)$. Therefore, for each aggregate root, a set of aggregates can be selected according to higher ρ_{is} values. As a result of the Root Word level selection (RWL), an initial Mixed Graph of Terms structure, composed by H aggregate roots s (R_l) linked to all possible aggregates (W_l), is obtained. An optimization phase allows to neglect weakly related pairs according to a fitness function discussed in [36]. Our algorithm, given the number of aggregate roots H and the desired max number of pairs as constraints, chooses the best parameter settings τ and $\mu = (\mu_1, ..., \mu_H)$ defined as follows:

1) τ : the threshold that establishes the number of aggregate root/aggregate root pairs. A relationship between the aggregate root v_i and aggregate root r_j is relevant if $\psi_{ij} \geq \tau$.

2) μ_i : the threshold that establishes, for each aggregate root i, the number of aggregate root/word pairs. A relationship between the word vs and the aggregate root r_i is relevant if $\rho_{ij} \geq \mu_i$.

Note that a Mixed Graph of Terms structure which can be suitably represented as a graph g of terms (Fig. 3). Such a graph is made of several clusters, each containing a set of words v_s (aggregates) related to an aggregate root (r_i), the centroid of the cluster. Aggregate roots can be also linked together building a centroids subgraph.

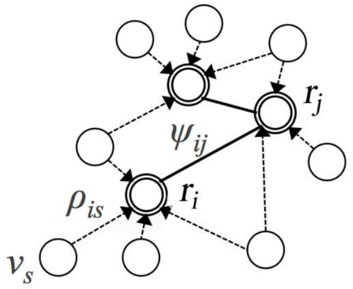

Figure 3. Graphical representation of a Mixed Graph of Terms structure as in [37]

4.2. Classifier

Formally speaking, text categorization aims at assigning a Boolean value to each pair $(d_j, c_i) \in D \times C$ where D is a set of documents and C a set of predefined categories. A *true* value assigned to (d_j, c_i) indicates document d_j to be set under the category c_i, while a *false* value indicates d_j cannot be assigned under c_i. In our scenario, we consider a set of job vacancies $\mathcal{J}$ as a collection of documents each of which has to be assigned to one (and only one) ISCO occupation code. We can model this problem as a text classification problem, relying on the definition of [27]. Formally speaking, let $\mathcal{J} = \{J_1, \ldots, J_n\}$ be a set of Job vacancies, the classification of $\mathcal{J}$ under $|O|$ ISCO occupation labels consists of $|O|$ independent problems of classifying each job vacancy $J \in \mathcal{J}$ under a given ISCO occupation code o_i for $i = 1, \ldots, |O|$. Then, a *classifier* for o_i is a function $\psi : \mathcal{J} \times O \rightarrow \{0, 1\}$ that approximates an unknown target function $\breve{\psi} : \mathcal{J} \times O \rightarrow \{0, 1\}$.

Clearly, as we deal with a single-label classifier, $\forall j \in \mathcal{J}$ the following constraint must hold: $\sum_{o \in O} \psi(j, o) = 1$. To classify job vacancies based on ISCO Code, we propose a framework by applying the MGT method to text classification environment, we obtain a system that provide a classification of job vacancies with the evaluation of performances and explainable of system. The framework used is shown in Fig. 4; it uses of the following elements:

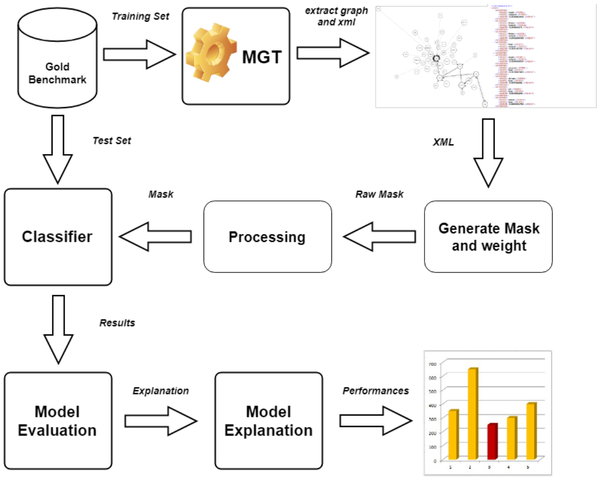

Figure 4. Framework

Gold Benchmark. This is the gold benchmark used to train the system, as presented in [1]. It has been realized by experts that took part in the Cedefop EU Project [10]. Data have been collected and classified by experts belonging to the ENRLMM[2] on the corresponding ISCO code. Clearly, the masks generation has been performed using a k-fold that divides the dataset into 2 training set and test set. The former is used to train the system while the latter is used to assess the obtained masks by the system. This procedure is applied for each occupation, generating a mask for each ISCO Code.

MGT. As described in the previous section, a Mixed Graph of Terms gives a compact representation of a set of documents related to a well-defined knowledge domain. In this way the obtained graph and its related XML files can be considered as a filter to be employed in our classification problem. We assumed as parameters values the ones introduced for the system in the paper [37] (Tab. 1). Max Pairs are

[2] The european network on regional labour market monitoring - http://www.regionallabourmarketmonitoring.net/

Table 1. MGT parameters

Parameter	Value
Max pairs	35
LDA Alpha	0.5
LDA Beta	0.001
LDA Topics	30
LDA Iterations	10,000
K-Means CP	5
K-Means PP	5
K-Means Iterations	100

maximum number of pairs words considered, LDA Alpha represents document-topic density, LDA Beta represents topic-word density, LDA Topic represent the number of topic considered, LDA Iterations represents the number of LDA interact, K-means CP is the number of set that are considered for Ψ_{ij}, K-means PP is the number of set that are considered for ρ_{is}, K-mean iterations are represents the number of k-mean interacts. These parameters have been chosen with these values in reference to [36] [37]. As Number of Concepts (roots) we have considered for each ISCO Code 6-7-8-9-10 roots.

Graph and XML. The obtained Graph and its related XML is used to generated the masks and the weight of each words. For each ISCO Code a graph and its related XML representation is generated. In this way we can synthesise five different masks for a given ISCO Code, each of which differs from the others in terms of roots.

This represents the most important phase of our approach that synthesises words and weights from XML files. To generate a mask for a given ISCO code, we consider both root and non-root terms. Then, we consider the number of connecting occurrences exist between (i) a root term and other root terms; (ii) root term and other non-root terms. This allows computing the weight a root term WR_i:

$$WR_i = \frac{\sum_{k=1}^{N} ArcNotRootR_i}{\sum_{k=1}^{N} ArcNotRoot} * \alpha + \frac{\sum_{k=1}^{N} ArcRootR_i}{\sum_{k=1}^{N} ArcRoot} * \beta \tag{5}$$

Where:

- N represents the number of arcs related to the extracted words
- $\sum_{k=1}^{N} ArcNotRootWr_i$ represents the sum of all arcs of non-root words related to the word root R_i
- $\sum_{k=1}^{N} ArcNotRoot$ represents the sum of all arcs of non-root words linked to all the words root R
- $\sum_{k=1}^{N} ArcRootR_i$ represents the sum of all arcs of root words related to the word root R_i
- $\sum_{k=1}^{N} ArcRoot$ represents the sum of all arcs of root words related to all the words root R
- α and β represent the form factors whose sum is 1, which were obtained experimentally as will be described in the chapter on experimentation

The weight makes possible to give importance to certain words rather than to others within the same mask, improving the classification performances.

Processing. Once the masks are obtained, we generate the appropriate output in order to make the masks usable for the classification processes. To this end, words obtained with the relative weights are associated with the relative ISCO Code writing them on a CSV file.

Classifier. This module exploits the masks generated in the previous steps to perform the classification task, assigning each vacancy to one (and only one) label with a given score. To this end, we employ the index of Jaccard as score function (Eq. 6) as well as its version that takes into accounts the weight of each root (Eq. 7). The Jaccard index is a statistic used for comparing the similarity and diversity of sample sets.

$$J(A,B) = \frac{|A \cup B| - |A \cup B|}{|A \cup B|} \tag{6}$$

$$J_w(A,B) = \frac{\sum_{k=1}^{N} min(A_k, B_k)}{\sum_{k=1}^{N} max(A_k, B_k)} \text{ with } N = |A \cup B| \tag{7}$$

To better understand the matter, let us imagine to have two sets of terms A,B to be compared. In our case of study A represents the set of all words that compose the title job vacancies while B represents the set of all words correctly matched with mask. Formula 6 only considers the number of terms and roots to compute the index whilst formula 6 would also consider the weight of matching terms and roots. Using these metrics one can estimate which of our masks fit better with each job vacancies. In the case of we can consider even the weights of every single word of the mask.

Model Evaluation and Explanation. The evaluation of the model is done through Precision, Recall and F1-score, as these metrics give us an estimate of the sensitivity, accuracy and precision of our framework. Then, our the approach here proposed for classifying text basically exploits roots and their respective weights generated by MGT to assign an item to a single class. In essence, the roots generated act like an explanation of the model trained, clarifying which words guided the classification process, and to which extent a feature has been relevant. It is worth noting that the meaning of interpretation or explanation is still an open debate (see, e.g., [38]). Here, we consider an explanation as a human readable interpretation of the features that guided the system in assigning an item to a given class.

5. Experimental Results

In order to evaluate the proposed approach, several experiments were conducted starting from a dataset from [1] were real-life web job vacancies have been collected within the EU Project [10] and published in [1]. The dataset is composed of $35,936$ job vacancies manually labelled using 4 digits ISCO code, covering 271 out of 436 codes of the ISCO taxonomy. In this paper, for the sake of simplicity, we focus on ICT-related professions that are listed in Tab. 2[3].

Table 2. ISCO codes for ICT-related professions

ISCO Code	Description EN
1330	ICT service managers
2152	Electronics engineers
2511	Systems analysts
2512	Software developers
2513	Web and multimedia developers
2522	Systems administrators
3513	Computer network & systems technicians
3514	Web technicians

As a first step, we have divided the dataset into training and test sets, then we have generated the masks using the the titles of job vacancies ngrams (with n=4) starting from the training set with 6-7-8-9-10 roots and finally these masks were used to classify the job vacancies of the test set. To evaluate the performance of our system, we considered precision, recall and f1-score values using masks with both weights info and no weight information, according to Equations 6 and 7. A classical k-fold cross-validation, with k=10 was used, focusing only on job occupations related to ICT, as shown in Tab. 2. Results are shown in Tab. 3.

As one might note, increasing the number of roots to be synthesized does not penalize the scoring function (Eq. 7, if the root weights are considered. Indeed, the 10-root configuration reaches the highest value of 0.86 Weighted F1-score. This is not true for the scoring function that does not consider root weights (Eq. 6, as the configuration with 9 root outperforms the others.

Comparison with results from [1]. We compared our results against the ones obtained by [1], that employs several machine algorithms. Here we report results from the best classification output as reported in [1], that reached a 0.85 for F1-score using a Bag-of-Words preprocessing pipeline and a linear SVM classifier.

However, for the sake of completeness, we have to remark that the work [1] trained a classifier over 256 ISCO codes, and this clearly might affect the classification performances of the system. For obtaining a fair comparison, we used the same pipeline and algorithm used in [1] to train a classifier focusing *only* on the same ICT professions we considered, as described above. This resulted in a higher F1-score, that reached the 0.89, while we reached a 0.88 F1-score in our best settings, that is 10 roots.

Model Explanation. In essence, our research reveal that our approach is comparable with classical machine-learning techniques, that were tested on the same training/test dataset. In addition, our approach would give *explanations* to the final users, in order to explain the rationale that guided the decision whilst the classification process in [1] does not provide any explanations about the reasons that guided the classification. This behaviour might affect the reliability of the obtained results, especially in a domain where classified data are used for decision making activity. To this end, we employed the LIME [12] algorithm that provides the local interpretable model of a text classifier. Roughly, LIME takes as input (1) a classifier that learnt an unknown decision function f for classifying items over several classes, and (2) an item to be classified synthesising an interpretable model by sampling instances, then it gets the predictions using f, and it weights them by the proximity to the instance being explained. This results in an interpretable model that is locally (but not globally) faithful.

In our approach the role of explanation is played by *roots*, as they are returned by MGT along with a weight that describes the relevance of that term within the document collection. This would allow us having a *global* interpretation of our model, as shown in Tab. 4, where top 10 root terms have been retrieved using MGT for each occupation code, along with the weight of each root.

As one might note, features obtained through LIME confirm that the model learnt by SVM linear is really comparable to the one built by MGT but, in contrast, MGT is built in a forward manner whilst the features derived from the SVM algorithm through LIME are synthesized in a backward manner. Here, Jaccard index computed over the terms for each ISCO code is always above the 0.7 and often it reaches the 0.9, showing that

[3] Notice that each ISCO/ESCO concept can be accessed through the URI composed by a prefix and the code of the profesion (e.g., `http://data.europa.eu/esco/isco/C1330`) accesses to profession 1330, etc.

Table 3. Evaluation of Precision, Recall and F1-Score. Bold values indicate the highest value for each column.

# Roots	Precision	Weighted Precision	Recall	Weighted Recall	F1-score	Weighted F1-score
6	0.83	0.87	0.80	0.86	0.79	0.84
7	0.84	0.86	0.80	0.85	0.80	0.85
8	0.84	0.86	0.80	0.85	0.80	0.85
9	**0.85**	0.86	**0.82**	0.85	**0.82**	0.85
10	0.84	**0.88**	0.80	**0.87**	0.80	**0.86**

the learnt models are really comparable.

6. Conclusions and Future Steps

In this paper, a novel and explainable approach to Web job vacancy classification has been presented. The proposed system is built on top of the Mixed Graph of Terms algorithm. The proposed technique has been tested in a real-life settings framed within the research activity of a EU project, that aims at building the EU labour market intelligence system that collects, classifies vacancies and extracts skills from them [11]. Our results - in terms of classification performances - are comparable with ones obtained using machine-learning algorithm [1], that used real data from the previous EU project on the same field [10]. Specifically, we obtained a 0.86 F1-score with a configuration that uses 10 roots for the classification step, that is in line with the 0.85 F1-score obtained by [1] by using a linear SVM algorithm and the 0.89 F1-score value obtained by linear SVM focusing on ICT-related professions (here considered).

In contrast with classical machine-learning algorithm, our approach is preferable as it provides explanations for the classification task in a straightforward manner, as we have shown above. Indeed, the roots and their relative weights provided by MGT act like an explanation that allows end-users to understand the rationale behind the classification and to *validate* the results as well.

Further developments as a research part of the project involve (1) the application of the proposed approach by expanding the dataset to all existing ISCO codes and (2) the translation of masks from one language to another in order by applying the *cross-language text classification* process (see, e.g., [39]).

References

[1] R. Boselli, M. Cesarini, F. Mercorio, and M. Mezzanzanica, "An AI planning system for data cleaning," in *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2017, Skopje, Macedonia, September 18-22, 2017, Proceedings, Part III*, pp. 349–353, 2017.

[2] H. Li, Y. Ge, H. Zhu, H. Xiong, and H. Zhao, "Prospecting the career development of talents: A survival analysis perspective," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 917–925, ACM, 2017.

[3] C. B. Frey and M. A. Osborne, "The future of employment: How susceptible are jobs to computerisation?," *Technological Forecasting and Social Change*, vol. 114, no. Supplement C, pp. 254 – 280, 2017.

[4] UK Commission for Employment and Skills, "The importance of LMI, available at `https://goo.gl/TtRwvS`," 2015.

[5] M. Mezzanzanica and F. Mercorio, "Big data enables labor market intelligence," in *Encyclopedia of Big Data Technologies*, pp. 1–11, Springer International Publishing, 2018.

[6] UK Department for Education and Skills, *LMI Matters!* 2004.

[7] DARPA, "Explainable artificial intelligence (xai) program. `http://www.darpa.mil/program/explainable-artificial-intelligence`. full solicitation at `http://www.darpa.mil/attachments/DARPA-BAA-16-53.pdf`," 2016.

[8] O. Biran and C. Cotton, "Explanation and justification in machine learning: A survey," in *IJCAI-17 Workshop on Explainable AI (XAI)*, p. 8, 2017.

[9] F. Clarizia, F. Colace, M. De Santo, L. Greco, and P. Napoletano, "Mixed graph of terms for query expansion," in *Intelligent Systems Design and Applications (ISDA), 2011 11th International Conference on*, pp. 581–586, IEEE, 2011.

[10] CEDEFOP, "Real-time labour market information on skill requirements: feasibility study and working prototype". cedefop reference number ao/rpa/vkvet-nsofro/real-time lmi/010/14. contract notice 2014/s 141-252026 of 15/07/2014 `https://goo.gl/qNjmrn`," 2014.

[11] CEDEFOP, "Real-time labour market information on skill requirements: Setting up the eu system for online vacancy analysis ao/dsl/vkvet-grusso/real-time lmi 2/009/16. contract notice - 2016/s 134-240996 of 14/07/2016 `https://goo.gl/5FZS3E`," 2016.

[12] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should i trust you?: Explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144, ACM, 2016.

[13] R. Boselli, M. Cesarini, S. Marrara, F. Mercorio, M. Mezzanzanica, G. Pasi, and M. Viviani, "Wolmis: a labor market intelligence system for classifying web job vacancies," *Journal of Intelligent Information Systems*, pp. 1–26, 2017.

[14] S. Marrara, G. Pasi, M. Viviani, M. Cesarini, F. Mercorio, M. Mezzanzanica, and M. Pappagallo, "A language modelling approach for discovering novel labour market occupations from the web," in *2017 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2017)*, pp. 1026–1034, 2017.

[15] Lovaglio, Cesarini, Mercorio, and Mezzanzanica, "Skills in demand for ict and statistical occupations: Evidence from web-based job vacancies," *Statistical Analysis and Data Mining*, vol. 11, no. 2, pp. 78–91, 2018. cited By 0.

[16] I. L. Organization, "Isco:the international standard classification of occupations, available at `http://www.ilo.org/public/english/bureau/stat/isco/`," 2017.

[17] European Commission, "Esco: European skills, competences, qualifications and occupations, available at `https://ec.europa.eu/esco/portal/browse`," 2017.

[18] EuroStat, "The essnet big data project - available at `https://goo.gl/EF6GtU`," 2016.

[19] P. Neculoiu, M. Versteegh, M. Rotaru, and T. B. Amsterdam, "Learning text similarity with siamese recurrent networks," *ACL 2016*, p. 148, 2016.

[20] J. Koperwas, Ł. Skonieczny, M. Kozłowski, P. Andruszkiewicz, H. Rybiński, and W. Struk, "Intelligent information processing for building university knowledge base," *Journal of Intelligent Information Systems*, pp. 1–23, 2016.

[21] S. Andrews, H. Gibson, K. Domdouzis, and B. Akhgar, "Creating corroborated crisis reports from social media data through formal concept analysis," *Journal of Intelligent Information Systems*, vol. 47, no. 2, pp. 287–312, 2016.

[22] T. Kanan and E. A. Fox, "Automated arabic text classification with p-stemmer, machine learning, and a tailored news article taxonomy," *JASIST*, vol. 67, no. 11, pp. 2667–2683, 2016.

[23] N. Jindal and B. Liu, "Opinion spam and analysis," in *Proceedings of the 2008 International Conference on Web Search and Data Mining*, pp. 219–230, ACM, 2008.

[24] M. Viviani and G. Pasi, "Credibility in Social Media: Opinions, News, and Health Information - A Survey," *WIREs Data Mining and Knowledge Discovery*, 2017.

[25] A. Bifet and E. Frank, "Sentiment knowledge discovery in twitter streaming data," in *International Conference on Discovery Science*, pp. 1–15, Springer, 2010.

[26] D. Vilares, M. A. Alonso, and C. Gómez-Rodríguez, "On the usefulness of lexical and syntactic processing in polarity classification of twitter messages," *JASIST*, vol. 66, no. 9, pp. 1799–1816, 2015.

[27] F. Sebastiani, "Machine learning in automated text categorization," *ACM computing surveys (CSUR)*, vol. 34, no. 1, pp. 1–47, 2002.

[28] M. Fox, D. Long, and D. Magazzeni, "Explainable planning," *arXiv preprint arXiv:1709.10256*, 2017.

[29] O. Biran and C. Cotton, "Explanation and justification in machine learning: A survey," in *IJCAI-17 Workshop on Explainable AI (XAI)*, p. 8, 2017.

[30] F. Clarizia, F. Colace, M. De Santo, M. Lombardi, F. Pascale, and A. Pietrosanto, "E-learning and sentiment analysis: A case study," in *Proceedings of the 6th International Conference on Information and Education Technology*, ICIET '18, (New York, NY, USA), pp. 111–118, ACM, 2018.

[31] H. B. Yalamanchili, S. J. Kho, and M. L. Raymer, "Latent dirichlet allocation for classification using gene expression data," in *2017 IEEE 17th International Conference on Bioinformatics and Bioengineering (BIBE)*, pp. 39–44, Oct 2017.

[32] A. K. Diop, S. Meza, M. Gordan, and A. Vlaicu, "Lda based classification of video surveillance sequences using motion information," in *2018 20th International Conference on Advanced Communication Technology (ICACT)*, pp. 1–1, Feb 2018.

[33] Q. Chen, L. Yao, and J. Yang, "Short text classification based on lda topic model," in *2016 International Conference on Audio, Language and Image Processing (ICALIP)*, pp. 749–753, July 2016.

[34] M. Pavlinek and V. Podgorelec, "Text classification method based on self-training and lda topic models," *Expert Systems with Applications*, vol. 80, pp. 83 – 93, 2017.

[35] A. Y. N. David M. Blei and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn.*, vol. 3, pp. 993–1022, 2003.

[36] L. Greco, F. Colace, M. D. Santo, and P. Napoletano, "Query expansion through weighted word pairs," in *22nd Italian Symposium on Advanced Database Systems, SEBD 2014, Sorrento Coast, Italy, June 16-18, 2014.*, pp. 399–406, 2014.

[37] F. Colace, M. D. Santo, L. Greco, and P. Napoletano, "Text classification using a few labeled examples," *Computers in Human Behavior*, vol. 30, pp. 689 – 697, 2014.

[38] Z. C. Lipton, "The mythos of model interpretability," *arXiv preprint arXiv:1606.03490*, 2016.

[39] N. Bel, C. H. Koster, and M. Villegas, "Cross-lingual text categorization," in *International Conference on Theory and Practice of Digital Libraries*, pp. 126–139, Springer, 2003.

Table 4. Top-10 explanations (ngrams) for the ISCO codes considered limited to ICT-related professions as listed in Tab. 2. mgt refers to the classification pipeline we introduced. SVM+LIME refers to the linear SVM trained only on ICT-related professions, and explained using the LIME tool [12]. Values within parentheses indicate the weight of the term in the corresponding approach. Light (dark) gray cells refer to terms introduced by mgt (SVM+LIME) and not identified by SVM+LIME (mgt).

Code	Method	Ngram 1	Ngram 2	Ngram 3	Ngram 4	Ngram 5	Ngram 6	Ngram 7	Ngram 8	Ngram 9	Ngram 10	Jaccard
1330	mgt	manag (0.28)	ict manag (0.19)	ict work manag (0.07)	ict categori manag (0.06)	head (0.08)	project manag (0.08)	global (0.05)	ict serv (0.06)	ict (0.06)	servic manag (0.04)	0.764
	SVM+LIME	manag (0.07)	ict manag (0.03)	ict (0.02)	project manag (0.01)	head (0.01)	director (0.001)	ict work manag (0.009)	procur manag (0.005)	account manag (0.0506)	ict categori manag (0.004)	
2152	mgt	electron engin (0.30)	hardwar engin (0.13)	network architect (0.085)	electron (0.08)	uat electron engin (0.08)	electron hardwar engin (0.08)	princip electron engin (0.06)	infrastructur engin (0.08)	electron engin light (0.06)	system engin (0.05)	0.947
	SVM+LIME	electron engin (0.11)	electron (0.034)	engin (0.01)	hardwar engin (0.01)	infrastructur engin (0.01)	electron design engin (0.01)	uat electron engin (0.01)	princip electron engin (0.01)	electron hardwar engin (0.007)	network architect (0.006)	
2511	mgt	system analyst (0.2)	system architect (0.07)	engin (0.17)	busi analyst (0.11)	system consult (0.05)	analyst (0.05)	system support (0.10)	system engin (0.07)	specialist (0.07)	financ system (0.07)	0.994
	SVM+LIME	busi analyst (0.04)	system (0.002)	analyst (0.017)	system engin (0.01)	system analyst (0.014)	system support (0.007)	technic architect (0.005)	system architect (0.005)	system consult (0.004)	financ system (0.004)	
2512	mgt	develop (0.2)	softwar engin (0.18)	softwar develop (0.06)	uat (0.12)	java develop (0.07)	php develop (0.09)	agil (0.09)	javascript (0.09)	embed softwar engin (0.07)	net develop (0.04)	0.994
	SVM+LIME	develop (0.07)	softwar engin (0.028)	softwar develop (0.025)	java develop (0.015)	net develop (0.009)	php develop (0.005)	php develop (0.005)	embed softwar engin (0.005)	android develop (0.004)	java (0.003)	
2513	mgt	web develop (0.26)	develop (0.17)	javascript (0.14)	front end develop (0.11)	web (0.06)	php web develop (0.06)	web applic develop (0.06)	front end develop html (0.06)	asp (0.04)	html css javascript (0.02)	0.95
	SVM+LIME	web develop (0.11)	front end develop (0.01)	front end web develop (0.01)	web applic develop (0.01)	web (0.008)	net develop asp mvc (0.008)	front end develop html (0.006)	web develop html css (0.006)	asp (0.005)	php web develop (0.004)	
2522	mgt	system administr (0.23)	linux system administr (0.22)	support (0.09)	sql (0.05)	manag (0.08)	vmware (0.08)	develop system administr (0.08)	servicenow system administr (0.08)	system administr linux window (0.04)	technic consult (0.06)	0.9
	SVM+LIME	system administr (0.15)	linux system administr (0.07)	develop system admin (0.009)	system administr linux windows (0.006)	servicenow system administr (0.006)	windows system (0.006)	windows system administr (0.004)	system administr cloud (0.003)	admin (0.003)	system administr devop (0.002)	
3513	mgt	network technician (0.25)	network engin (0.14)	specialist (0.1)	school ict technician (0.07)	twork technician (0.07)	network infrastructur engin (0.08)	helpdes advisor (0.08)	autocad technician (0.07)	twork (0.06)	engin (0.06)	0.842
	SVM+LIME	network engin (0.07)	network technician (0.04)	technician (0.02)	specialist (0.014)	twork technician (0.014)	network infrastructur (0.014)	twork engin (0.014)	network infrastructur engin (0.014)	twork technician (0.011)	school ict technician (0.011)	
3514	mgt	web analyst (0.15)	web editor (0.2)	web content editor (0.12)	websit (0.11)	websit statist assist (0.09)	web (0.09)	web analyt (0.08)	assist (0.05)	onlin cont (0.05)	sitecor (0.05)	0.888
	SVM+LIME	webist (0.03)	web editor (0.03)	web analyst (0.028)	websit administr (0.02)	websit content editor (0.022)	web (0.019)	web analyt (0.019)	websit manag (0.01)	websit editor (0.013)	web content manag (0.013)	