Multilingual Skill Extraction for Job Vacancy–Job Seeker Matching in Knowledge Graphs

Hamit Kavas
NLP Group,
Pompeu Fabra University
Adevinta Spain
hamit.kavas@upf.edu

Marc Serra-Vidal
Adevinta Spain
marc.serrav@adevinta.com

Leo Wanner
NLP Group,
Pompeu Fabra University
Catalan Institute for Research
and Advanced Studies (ICREA), Spain
leo.wanner@upf.edu

Abstract

In the modern labor market, accurate matching of job vacancies with suitable candidate CVs is critical. We present a novel multilingual knowledge graph-based framework designed to enhance the matching by accurately extracting the skills requested by a job and provided by a job seeker in a multilingual setting and aligning them via the standardized skill labels of the *European Skills, Competences, Qualifications and Occupations* (ESCO) taxonomy. The proposed framework employs a combination of state-of-the-art techniques to extract relevant skills from job postings and candidate experiences. These extracted skills are then filtered and mapped to the ESCO taxonomy and integrated into a multilingual knowledge graph that incorporates hierarchical relationships and cross-linguistic variations through embeddings. Our experiments demonstrate a significant improvement of the matching quality compared to the state of the art.

1 Introduction

In the modern labour market, job portals serve as key intermediaries, connecting employers with potential employees by comparing job postings with the CVs of job seekers, to find the best possible matches and liaise both parties.[1] Matching of the required skills in a job posting with the skills outlined in the candidate's experience is of special relevance. For implementation of this matching, recently a number of AI-based techniques have been proposed. Zero-shot recommendation models (Kurek et al., 2024), attention-based scoring mechanisms (Jiang et al., 2020), embedding-based models (Zhao et al., 2021), or tensor decomposition (Mao et al., 2023) have been used, with an emphasis of surface-level features, such as geographic location and user data, including click-through rates and other user engagement metrics, without a deeper analysis of the semantics of the listed skills. Other, more promising, proposals draw upon the skills taxonomy of the *European Skills, Competences, Qualifications and Occupations* (ESCO) classification[2], which is, together with its US equivalent O*Net[3], a primary instrument used in international job markets for job offer and candidate experience annotation. For instance, (Clavié and Soulié, 2023) identify ESCO skill labels in job offers / candidate experiences and provide the reasoning, which leads to the identification of skills by the use of a Large Language Model (LLM), giving guidance to it by detailed prompts, accompanied by manually created examples, but without that the LLM can re-evaluate its reasoning chain. This is different from (Wang and Zhou, 2024), which uses the "Chain of Thought" (CoT) strategy to make the LLM use its own reasoning explanations as in-context examples. However, as other proposals in the field, these proposals neglect the hierarchically organized, contextually diverse information in the ESCO classification.Furthermore, they are prone to biases inherent in LLMs when processing textual data, as they do not involve any human expert control or human-generated ground truth data.

To address these challenges and to improve the quality of skill extraction from job postings and candidate experiences and their matching, we developed a knowledge graph-based multi-step framework. The proposed architecture foresees skill extraction using several state-of-the-art skill selection skills (Zhang and et al., 2024; D'Oosterlinck et al., 2024; Nguyen et al., 2024) to whose output (D'Oosterlinck et al., 2024)'s CoT prompting technique is applied to select the most relevant skills by comparing the extracted skills with the ESCO

[1] In what follows, we use the term "job posting" to refer to the description of a job vacancy and "candidate experience" to refer to the professional history of a job seeker as documented in their CV.

[2] https://esco.ec.europa.eu/en/classification

[3] https://www.onetonline.org/

Proceedings of the Generative AI and Knowledge Graph Workshop (GenAIK), pages 146–155
January 19, 2025.

skill names and their corresponding descriptions. Human labelers provide ground truth data, based on which the model is optimized. The finally selected skills from the job postings and candidate experiences are linked in a knowledge graph with the ESCO skills. The graph is fine-tuned with multilingual (in our case, Spanish) embeddings for effective matching in a multilingual context.

Our tests demonstrate that the developed framework significantly outperforms the state-of-the-art skill extraction and job posting – candidate experience matching. The framework is about to be deployed in a leading European job portal.

It will serve as a complementary instrument for human job recruiters rather than a fully automated service, in order to minimize any risk of faulty recruitment suggestions.

2 Related Work

As already mentioned, a number of works deal with the identification and matching of skills in job postings and candidate experiences (Senger et al., 2024). For instance, (Jia et al., 2018) and (Sayfullina et al., 2018) focus on skill identification through span labeling and binary classification. (Jia et al., 2018) use sequence tagging with LSTM neural networks, and (Sayfullina et al., 2018) employ exact match for skill spans. Both require extensive optimization and are often limited by their dependency on specific languages and datasets. (Goyal et al., 2023) utilizes exact string matching with NLP techniques for label extraction and incorporates a multi-hop job-skill graph neural network with a Graph Isomorphism Network encoder, employing one-vs-many classification with softmax attention and weighting skills based on their frequency in job postings. (Zhang and et al., 2024) emphasize the generation of mention-entity pairs using ESCO skills as ground truth labels, their methodology, which heavily depends on the precise detection of mentions and the disambiguation of contexts, proves insufficient in real-world applications where such extraction methods are not readily available. (Fettach et al., 2024) demonstrates how knowledge graphs can bridge education and employment domains to support job seekers' skill development. They developed JobEdKG, an uncertain knowledge graph embeddings to model relationships between job market and educational entities to predict required skills based on career choices.

The systems by (Zhao et al., 2021; Jiang et al., 2020) approach job-resume matching primarily as a recommender system, incorporating non-textual inputs like geo-location and evaluating performance based on click-through rates. This methodology inherently restricts the comparison to surface-level features shared between candidate experiences and job postings, neglecting the deeper semantic relationships and context that are crucial for accurate skill matching. Similarly, (Kurek et al., 2024; Mao et al., 2023) experiment with zero-shot learning using various embedding models and dimensions. While this approach may be effective for exploring a broad understanding of industrial problems, it overlooks several recent findings. Relying solely on embeddings with zero-shot classification has notable weaknesses, such as the lack of mechanisms to mitigate residual biases in textual data and limited capability to integrate diverse data sources. Moreover, zero-shot classification models have been shown to be outperformed by in-context learning models, particularly LLM, which offer superior contextual understanding and adaptability (Gurusamy et al., 2024; Hasan et al., 2024).

Apart from those that we already cited, in particular, (D'Oosterlinck et al., 2024) needs to be highlighted. (D'Oosterlinck et al., 2024) introduces "in-context learning for extreme multi-label classification" and built on DSPy library (Khattab et al., 2023). It consists of a collection of DSPy programs that enable stateful interactions with LLM. The LLM integrates the CoT methodology with retrieval-augmented generation for skill extraction, where classes—specifically, ESCO skill names—are provided as embeddings. The model learns classification through a novel optimization technique that fine-tunes prompts at each step and generates synthetic examples to enhance in-context learning (ICL) (Wang and Zhou, 2024), thereby aiming to maximize classification metrics. We use (D'Oosterlinck et al., 2024) as foundation. We employ human-labellers to evaluate the skills extracted by this model and introduce enhancements to improve contextual understanding of the LLM beyond the conventional NLP techniques utilized in (Goyal et al., 2023) Additionally, we adapt this model for multilingual applications and incorporate the hierarchical skill relationships defined in the ESCO taxonomy.

3 The Framework

In what follows, we first provide an overview of our approach as a whole and then discuss the skills extraction, filtering of the extracted skills and their subsequent mapping onto the ESCO skills taxonomy, and construction of the knowledge graph as a way of matching the skills extracted from the job postings and candidate experiences.

3.1 Framework overview

Figure 1 illustrates our framework for skill extraction and matching of job postings and candidate experiences.

Job postings and candidate experiences come in structured forms. Skills, their descriptions, and associated job titles are provided in separate fields, such that we first retrieve the content of these fields from the forms[4]. An LLM CoT prompting strategy is applied to normalize the retrieved job titles to unique labels, which are further extended by an index (e.g. ix_000) from the InfoJobs database in which the job postings and candidate experiences are stored. For instance, a generic job title like “administrativo/a" (Spanish term for “administrator") is transformed into a more specific and unique title, such as “organizational support specialist_ix_14662" or “excel data coordinator – international trade support_ix_4664". This prevents inefficient expansion of the knowledge graph at a later stage (see below), since multiple nodes that represent the same concept or entity can lead to redundant data and suboptimal graph performance (Hofer et al., 2023; Zhang et al., 2022b).

The skills field in job postings and candidate experiences often contains unstructured, ambigous, or inconsistent information. For instance, job postings may list skills using a variety of terminologies, such as “project management", “PM", or “leading teams", all of which refer to the same skill in different terms. These variations make the use of skill extraction techniques to standardize relevant information for matching necessary.

For skill extraction, we adapt three state-of-the-art skills extraction techniques, which are applied in parallel to the content retrieved from the job postings and candidate experiences: entity linking (Zhang and et al., 2024), extreme multi-label classification (Khattab et al., 2023), and in-context learning with LLMs (Nguyen et al., 2024). We use three techniques instead of one because each of the three approaches has its pros and cons, such that a combination of them promises to provide the best outcome possible.

The outputs of these three techniques are merged and processed using the *CoT with Hint* reasoning module (D’Oosterlinck et al., 2024) from the DSPy library (Khattab et al., 2023). This module is responsible for both the final selection of the relevant skills and their mapping to the corresponding ESCO skills by introducing a hint during an intermediate reasoning step to prompt stateful LLMs within a multi-step framework. Additionally, the hint also ensures that the extracted skills are translated into English (when the input is in another language, e.g., Spanish)

With the selected skills at hand, we construct a knowledge graph that links ESCO skill labels, job postings, and candidate experiences. The hierarchical relationships between the skills in the ESCO taxonomy are incorporated, along with n-gram matches, to assign initial weights to the edges linking skills with job postings and candidate experiences. The initial edge weights are subsequently refined using Inverse Document Frequency (IDF) scores. To further improve the multilingual capability of the knowledge graph and translate the structural characteristics of the graph into a vector space representation that enables precise comparison of nodes, we fine-tune it on multilingual (so far, Spanish) embeddings. This enables the knowledge graph to account for linguistic variations between the description of the skills in ESCO and in job postings / candidate experiences. Furthermore, language-specific embeddings capture language-specific nuances and cultural context, enabling precise alignment between English and data in the language in question, unlike the initial alignment, which offers a generalized, language-independent representation.

3.2 Skills extraction

As mentioned above, we extract skills from job postings and candidate experiences adapting three state-of-the-art techniques, whose output we then combine for more accurate performance: entity linking, extreme multi-class classification, and few-shot in-context learning.

[4]Prior to their processing, both job postings and candidate experiences are cleaned. Promotional content, sensitive information, specific locations, names, and dates that could reveal private details about the hiring company or job seeker are removed.

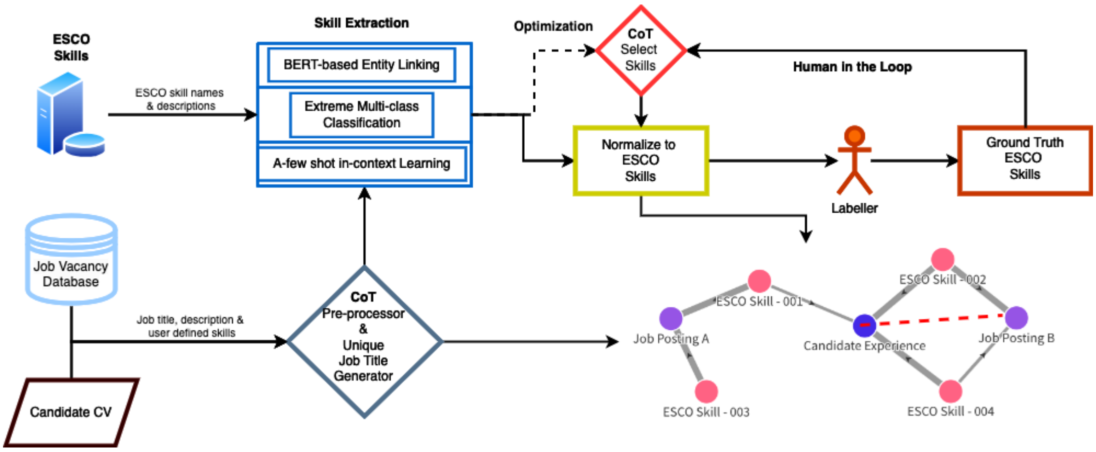

Figure 1: Overview of our framework for skills extraction and matching

3.2.1 Entity linking

Entity linking is used to link fine-grained span-level mentions of skills in candidate experiences to skills in the ESCO classification. We use (Zhang and et al., 2024), which is based on the BLINK model (Wu et al., 2020), adapted to the job domain. In this context, the entities refer to ESCO skill names and their corresponding descriptions. The descriptions are derived from the ESCO taxonomy, which provides detailed contextual information for each skill. BLINK first retrieves potential matches within a dense vector space, where a bi-encoder separately encodes the context of skill mentions and the descriptions of entities, which are two distinct inputs. The first input that represents the mention and its context has the following format:

```
[CLS]ctxt_left[S]mention[E]
                    ctxt_right[SEP]
```

where “mention" refers to the tokens of the actual mention, while “ctxt_left" and "ctxt_right" stand for the tokens of the surrounding contextual information. Special tokens [S] and [E] are used to delineate the mention itself.

The second input, representing the ESCO skill and its description, incorporates the skill’s name (title) and its textual description, separated by a special token.

```
[CLS]skill_name[S]skill_desc[SEP]
```

Following the retrieval stage, a cross-encoder reranks the candidate experience entities by concatenating the mention and entity spans for a precise alignment. The model aims to maximize the similarity, measured by the dot product, of the [CLS] token embeddings for the correct entity relative to other entities in the batch.

Given that our data comprise job titles, descriptions, and skills in plain text, we extract skill mentions along with their surrounding context to utilize this model effectively. First, we employ a continiously pre-trained, domain-adapted skill extraction model[5] proposed in (Zhang et al., 2022a) to identify sequences that most likely contain skills. Sentences containing these identified spans are selected as inputs. Then, we refine the extracted spans by removing, stop words, subword markers, punctuation, and other non-letter characters to ensure coherence and readability. The processed spans are formatted into a structure required by the BLINK model, allowing accurate context and entity-linking. In instances where the span appeared at the beginning or end of a sentence, the context was represented by an empty string, as outlined in (Zhang and et al., 2024).

3.2.2 In-context learning for extreme multi-label Classification

Our second technique for skill extraction is “Infer-Retrieve-Rank" (IReRa) (D’Oosterlinck et al., 2024) for tackling extreme multi-label classification (XMC) problems that involve classifying items into thousands of possible categories (Khattab et al., 2023). IReRa employs a multi-step interaction between LLMs and retrievers, optimizing the process using the DSPy programming model.

We reproduced the results from the original IreRa framework following the prescribed methodology, including the optimization of the language

[5]https://huggingface.co/jjzha/jobspanbert-base-cased

```python
class NormalizerSignature(dspy.Signature):
        __doc__ = """
        Normalize automatically assigned skills against the ESCO taxonomy based on the provided
        job descriptions. Map only the relevant skills to the most relevant ESCO skill,
        improving the precision of skill normalization. Be concise.
        """
        vacancy = dspy.InputField(
            prefix="Vacancy:",
            desc="The complete job description."
        )
        auto_skill = dspy.InputField(
            prefix="Skill:",
            desc="The skill that has been automatically assigned and needs normalization."
        )
        options = dspy.InputField(
            prefix="Options:",
            desc="List the possible ESCO skills and their descriptions that could correspond
            to the automatically assigned skill."
        )
        output = dspy.OutputField(
            prefix="Normalized Skill:",
            desc="The ESCO skill selected from the provided options that best matches the
            job description and assigned skill."
        )
```

Figure 2: DSPy signature used for Selecting and Normalizing Extracted Skills.

model with examples provided in their repository[6].

3.2.3 Few-shot skill extraction using Large Language Models

As the third technique, we implemented prompting of an LLM for skill extraction through a few-shot in-context learning, as proposed by (Nguyen et al., 2024). We adopted their original proposal to facilitate skill extraction from longer sentences. The exact prompts and few-shot examples from the original proposal. To ensure robustness and mitigate transient issues, we run the extraction up to five times to mitigate errors (e.g., poorly structured output from the LLM).

3.3 Skill selection and ESCO mapping

The skills extracted by the skill extraction techniques from above are further filtered with respect to their relevance, and their final selection is mapped onto the ESCO skills taxonomy, see Figure 2 for the prompt used in the DSPy program. The filtering model is a version of IReRa (D'Oosterlinck et al., 2024) that has been specifically adapted for multilingual skill extraction by incorporating additional context from the ESCO taxonomy. The model is further refined to improve the accuracy of skill selection compared to the ground truth, allowing for evaluation against its variations.

While building upon (D'Oosterlinck et al., 2024), our framework introduces several key enhancements to bridge the language gap between English ESCO skill labels and job postings / candidate experiences in other languages (in our case, Spanish): (1) integration of E5 multilingual embeddings[7] (Wang et al., 2024) to handle cross-lingual variations, (2) incorporation of ESCO skill descriptions into the vector database[8] to provide richer contextual understanding, and (3) development of a novel knowledge graph structure that captures hierarchical relationships between skills. But even if we apply multilingual embeddings for processing the job postings, we retain English for our prompts to acknowledge that maintaining prompts in English can yield advantageous outcomes, even for cases where the data is in another language (Razumovskaia et al., 2024). We use this multilingual adaptation of IReRa with aforementioned enhancements as the baseline model in our experiments.

To further improve (D'Oosterlinck et al., 2024)'s model, we implemented a method similar to few-shot in-context learning (ICL) (Nguyen et al., 2024), using training ground truth data. Annotator-agreed ground truth vacancies and their corresponding skills were provided as examples. To organize examples for in-context learning (ICL), we followed the multilingual alignment proposed by (Tanwar et al., 2023), which states that using semantically similar examples to construct the prompt-context aids the model in in-context inference. The alignment guides the LLM in mapping the source job posting to relevant target examples. To select the most semantically informative examples, we employed cosine similarity between the input job posting and segments of example job postings (Liu et al., 2021). In addition, we optimized the model through bootstrapping few-shot demonstrations, as proposed in (Khattab et al., 2023). For this purpose, we employed a random search method where Claude 3.5 Sonnet served as the teacher model, and Llama-3 served as the student model[9].

3.4 Knowledge Graph Construction

To effectively match the candidate experiences with job postings, we construct a knowledge graph and refine it using multilingual (in our case, Spanish) embeddings.

First, a graph is created with job postings and ESCO skills as nodes, and the links between the ESCO skills and the postings that possess them as weighted edges. The initial weights of the edges are determined by the hierarchical relationships between skills as defined in the ESCO taxonomy. ESCO organizes skills in a three-levels: broad skill groups, intermediate sub-groups, and specific com-

[6] https://github.com/KarelDO/xmc.dspy

[7] Instruction-tuned multilingual E5 text embeddings model employs a multi-stage contrastive learning methodology to obtain high-performance general-purpose embeddings.

[8] In all experiments with vector store, we use the open-source platform https://www.trychroma.com/.

[9] https://ai.meta.com/blog/meta-llama-3/

petencies. In this hierarchy, we utilize parent-child relationships between adjacent levels to reflect how general skills are linked to more specialized skills. Additionally, we introduce an another parent-child relation between skills when an n-gram of a skill is mentioned within another skill, thereby designating the former as the parent. For example, “graphical designer" and “graphical designer intern" illustrate a parent-child relation, with “graphical designer intern" as the child node. Each parent node is assigned an initial weight of 1, while the child skill nodes connected to them receive an initial weight of 0.5. We refine the initial edge weights using Inverse Document Frequency (IDF) scores to account for skill specificity and relevance. This refinement is to distinguish between common skills that appear frequently across many job postings and candidate experiences (receiving lower weights), and specialized skills that are more discriminative (receiving higher weights). The IDF scores are multiplied with the initial hierarchy-based weights, resulting in a weighted graph that better reflects both the hierarchical importance and the distinctiveness of each skill connection.

As next, candidate experiences are incorporated as nodes, and their skills are connected to the skill nodes in the graph; skills present in candidate experiences that were not previously defined in the graph are omitted from consideration. IDF scores are again used to assign weights to the edges connecting candidate experiences to their respective skills.

To facilitate semantically meaningful connections within the knowledge graph, we fine-tune the knowledge graph with Spanish embeddings since in our implementation, job vacancies and candidate experiences are in Spanish, while ESCO skills are in English. This ensures a more nuanced understanding that goes beyond a simple skills assignment. We first convert pre-trained fastText Spanish embeddings (Cañete, 2019), which have been opted for due to their superior performance on multilingual data into Word2Vec format (Mikolov et al., 2013). This conversion is necessary for the subsequent transformation of the embeddings into a Node2Vec graph (Grover and Leskovec, 2016), which extends Word2Vec representations to graph-structured data. Finally, we fine-tune the graph embeddings using a maximal-entropy biased random walk approach (Sinatra et al., 2010) to optimize the representation. We use biased random walks instead of simple random walks because they have been shown to be more effective, particularly for preferentially exploring certain paths and handling weighted graphs (Liu et al., 2022). As a result, the graph's topological information is translated into a vector space, allowing efficient computation of node similarities and facilitating various graph-based tasks such as classification and clustering.

In formal terms, the knowledge graph is an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes and $\mathcal{E}$ is the set of edges. For a node $v \in \mathcal{V}$, a random walk of length T is a sequence of nodes $\{v_0 = v, v_1, \ldots, v_T\}$, where each node v_i is chosen based on the following transition probabilities:

$$P(v_{i+1} = x \mid v_i = y) = \begin{cases} \frac{1}{p}, & \text{if } x = v_{i-1}, \\ \frac{1}{q}, & \text{if distance}(v_{i-1}, x) = 2, \\ 1, & \text{otherwise.} \end{cases} \quad (1)$$

with the p parameter controlling the likelihood of revisiting the previous node and selected as 0.5 and $q = 1/p$ influencing the exploration of more distant nodes. This process balances local and global exploration of the graph structure.

The sequences generated by the biased random walks are then used to learn node embeddings using the Skip-Gram model, where the goal is to maximize the likelihood of observing a node's neighbors given its embedding:

$$\max_f \sum_{v \in \mathcal{V}} \sum_{u \in \mathcal{N}_v} \log \Pr(u \mid f(v)),$$

where $\mathcal{N}_v$ denotes the context nodes of v, and the probability $\Pr(u \mid f(v))$ is given by:

$$\Pr(u \mid f(v)) = \frac{\exp(f(u) \cdot f(v))}{\sum_{v' \in \mathcal{V}} \exp(f(v') \cdot f(v))}.$$

This optimization ensures that nodes with similar structural roles in the graph have similar embeddings in the vector space.

To quantify the similarity between two nodes $v_i, v_j \in \mathcal{V}$, we compute the cosine similarity between their embeddings.

4 Experiments

4.1 Technical infrastructure

To ensure privacy of companies and users, all models in our experiments were executed locally within an isolated environment.[10] Blackbox models with internet access, which could potentially share sensitive data were restricted to processing only anonymized job postings and were never applied to candidate experiences.

As technical infrastructure, we use the DSPy library (Khattab et al., 2023), which has been designed to create and optimize language model (LM) pipelines. We employed several DSPy programs to structure the data and build our experimental pipeline. One DSPy program processed chunks of CV data to generate synthetic job experiences that ideally match job postings as described in 'Data' section. Another DSPy program, namely skill selection model, see Figure 2, was used to automatically normalize assigned skills against the ESCO taxonomy. This ensured that no extracted skill was omitted. Given the variability in user-generated job titles, another DSPy program was employed to generate unique job titles, thereby maintaining the integrity and uniqueness of the Knowledge Graph.

The knowledge graph is constructed using the StellarGraph library (Data61, 2018). Data is stored in Amazon S3 buckets, with computational tasks executed on an RTX 3090TI GPU for the local LLM and deep learning models. Additionally, the framework utilizes the Claude API via Amazon Bedrock. The entire framework is containerized and fully prepared for testing prior to deployment.

4.2 Evaluating Matching

To evaluate the proposed methodology against SoA skill extraction techniques and skill alignment in the knowledge graph, we use synthetic data. We construct three distinct knowledge graphs with the skills selected by two of the applied skill extraction strategies BLINK (Zhang and et al., 2024) and IReRa (D'Oosterlinck et al., 2024) and our final skill filtering model. The few-shot skill extraction (Nguyen et al., 2024) was not used here because it is not designed for the development of ESCO skills.

For the evaluation of the matching, we use Jaccard and cosine measures. Jaccard similarity is applied to exact skill matches, and cosine similarity to the vectorized graph with Spanish embeddings.

For our biased random walks in the knowledge graph, random walks of length 30 were generated, with each node serving as the root for 10 walks. The bias parameters were set to $p = 0.5$ and $q = 2.0$. Edge weights were determined by multiplying the ESCO hierarchy coefficient (0.5 for child nodes, 1 for parent nodes) with the IDF, and these weights were considered during the walk generation. The process was seeded with a fixed value of 42 to ensure reproducibility. The walks were used to fine-tune a Word2Vec model with the following parameters: vector size of 100, window size of 5, minimum count of 1, and Skip-Gram method (sg=1). The model was initialized with pre-trained vectors and fine-tuned on the generated walks. We computed cosine similarity between node embeddings to evaluate their relational similarity.

4.3 Data

For the evaluation of our skill selection, we use a subset of the job postings and candidate experiences from the InfoJobs database, focusing on records where both descriptions and user-defined skills are included. Specifically, we selected 648 job postings and 1,200 candidate experiences from the database. From this subset, we select ESCO occupations that are particularly ambiguous due to their inclusion in various other occupations. Among the selected occupations are, e.g., "administrative assistant", "office clerk", "sales assistant", "project manager", and "human resources assistant". These labels were assigned to job postings and candidate experiences, following the methodology described in (Kavas et al., 2024). Non-informative elements in job postings, such as company descriptions and promotional content have been removed. Sensitive information in both job postings and candidate experiences was anonymized by a DSPy program that utilizes zero-shot LLM inference.

We used the 1,200 candidate experiences to guide the generation of synthetic experiences by providing real-world language and terminology. These experiences were split into chunks and stored in a vector database, to facilitate similarity searches during the synthetic experience generation process.

For annotation detailed in the 'Ground Truth Creation' section below, we have selected 120 job postings from the obtained dataset characterized by longer descriptions and detailed requirements. The remaining 528 postings were used for testing and validation purposes in our experiments.

To evaluate our knowledge graph-based matching proposal, we generated synthetic job experiences using the local Llama-3 model, implemented through DSPy modules. First, a job posting with predefined skills was randomly selected. We ex-

[10]We use dockerized models from the open-source Ollama library https://ollama.com/ for all experiments.

tracted n-grams from the job posting description, focusing on skill mentions detected by semantic similarity. Candidate experiences were split into chunks and were stored in a vector database. Vector similarity using a multilingual retriever based on the E5 model was employed to identify the most relevant candidate experience chunks. Finally, a DSPy program that employs a CoT methodology and sample Spanish language expressions from candidate experiences was used to guide the generation process for creation of synthetic experiences optimized to match the selected job postings.

4.4 Ground Truth Creation

To evaluate the performance of our skill extraction, we conducted a systematic annotation of job postings, engaging experts from a job market place running company. To mitigate potential biases, we ensured diversity among the annotators in terms of their expertise and backgrounds. The skills presented to the annotators were derived from the union of skills extracted by the three state-of-the-art techniques and normalized, as described above. To enhance the comprehension of the individual ESCO skills, each ESCO skill was accompanied by its corresponding description from the ESCO classification.

The annotation involved 12 annotators: 10 classified the skills in the job postings, while the remaining 2 focused on annotating training data and resolving controversies. For each job posting, about 30 automatically assigned skills were presented. Annotators were instructed to classify each skill as either “essential" or “irrelevant". It is noteworthy that while the job offers were in Spanish, the ESCO skills and their descriptions were presented in English. All annotators were native Spanish speakers with proficiency in English.

We provided detailed guidelines to the annotators, emphasizing objective evaluation based solely on the content of the job postings to reduce personal biases and ensure consistency across annotations. The annotation was conducted in two rounds. In the first round, each of the 10 annotators evaluated 10 job postings from the dataset. At the same time, we employed the Claude 3.5 Sonnet model[11] to perform the same labeling task as the human annotators. We extracted the top 15 job postings where the model and human annotators agreed and selected 15 job postings with the least agreement. Two remaining annotators re-evaluated the 30 annotations. Skills derived from the combined annotations of both rounds were then used in the experiments. To mitigate potential biases, the task instructions were ensured to avoid any implicit suggestion to the annotators. We deliberately refrained from providing specific examples or controversial cases to prevent any influence on the annotators' judgments, allowing them to rely on their professional expertise.

[11] https://www.anthropic.com/news/claude-3-5-sonnet

4.5 Results

Skill Selection Method	P	R	F1	$\bar{X}$
ICL enhanced (claude-3.5)	0.46	0.5	0.48	5.81
Baseline (claude-3.5)	0.58	0.34	0.43	3.12
DSPy Optimized (llama-3)	0.48	0.28	0.35	3.13
ICL enhanced (llama-3)	0.4	0.28	0.33	3.74
Baseline (llama-3)	0.36	0.23	0.28	3.40

Table 1: Comparison of the performance of skill selection techniques (‘$\bar{X}$’ stands for “average number of predicted skills"; as Baseline, we use (D'Oosterlinck et al., 2024))

Table 1 reports the results of our baseline skill selection model and its enhancements, namely in-context learning (ICL) and DSPy optimization with human-annotated skills. Due to inherent limitations of black-box models, which cannot be fine-tuned or optimized with custom scripts, DSPy optimization was applied exclusively to the LLaMA-3 model, with the Claude 3.5 Sonnet model serving as the teacher. We see that particularly the Claude model, when enhanced with in-context learning (ICL), shows an increase in recall by recognizing a broader range of potential skills in the text. However, this increase comes at the cost of precision, as the model becomes more permissive, leading to an increase in false positives. Similarly, the LLaMA-3 model exhibits comparable behavior, with recall improving more substantially than precision under ICL. The optimization proved effective, as it enhanced the F1 score, even if it predicted, in the average, a lower number of skills.

Model	Metric: Score
IReRa SE Model	Jaccard: 0.3084
Blink SE Model	Jaccard: 0.4157
Skill Selection (Baseline)	Jaccard: 0.5002
Skill Selection (Fine-tuned)	Cosine: 0.5761

Table 2: Matching Performance of Knowledge Graph

Table 2 shows the performance of the matching of candidate experiences and job postings. According to the Jaccard scores, the Blink model (Zhang and et al., 2024) surpasses the IReRa model (D'Oosterlinck et al., 2024) in performance. For skill extraction, our integration of E5 multilingual embeddings and CoT-guided filtering shows a significant improvement in recall while maintaining competitive precision. For matching, the incorporation of hierarchical and multilingual embeddings enhances the cosine similarity of candidate experience-job posting pairs, achieving a higher alignment score compared to the baseline method.

The Jaccard score of 0.5002 establishes a baseline for our final model for skill selection. Subsequent fine-tuning, which includes the vectorization of the knowledge graph and incorporation of Spanish embeddings, enhances the cosine similarity score to 0.5761. This indicates that fine-tuning has improved matching accuracy, suggesting a better alignment between job postings and candidate experiences.

5 Conclusions

Our knowledge graph-based framework for candidate experience–job posting matching combines extensions of several state-of-the-art techniques, which leads to a versatile and robust application, apt for industrial use. The integration of entity linking mechanisms (Zhang and et al., 2024) enhances contextual precision, while extreme multi-label classification (D'Oosterlinck et al., 2024) addresses the extensive taxonomy of potential competencies, and the use LLM with few-shot in-context learning methodologies (Nguyen et al., 2024) ensures a high robustness of the skill extraction task. Our extension of (D'Oosterlinck et al., 2024)'s model provides enhanced reasoning capability by utilizing a CoT technique that allows LLM to review its own reasoning. It also enforces the model to respond in the input language, even when provided with context in another language. Furthermore, it normalizes the selected skills against the ESCO taxonomy and incorporates a hint at an intermediate reasoning step, guiding the model to refine its initial output. The extended model significantly improves on the ability to accurately process multilingual data, compared to previous methods that relied on string matching or regular expressions to map extracted skills to the ESCO taxonomy.

Finally, representing job postings, candidate experiences and skills as nodes within a knowledge graph allows for detailed and structured analysis of skill overlap and compatibility, often missed in linear or non-relational models (Yu et al., 2024).

References

José Cañete. 2019. Compilation of large spanish unannotated corpora.

Benjamin Clavié and Guillaume Soulié. 2023. Large language models as batteries-included zero-shot esco skills matchers. *Preprint*, arXiv:2307.03539.

CSIRO's Data61. 2018. Stellargraph machine learning library. https://github.com/stellargraph/stellargraph.

Karel D'Oosterlinck, O. Khattab, François Remy, Thomas Demeester, Chris Develder, and Christopher Potts. 2024. In-context learning for extreme multi-label classification. *ArXiv*, abs/2401.12178.

Yousra Fettach, Adil Bahaj, and Mounir Ghogho. 2024. Jobedkg: An uncertain knowledge graph-based approach for recommending online courses and predicting in-demand skills based on career choices. *Engineering Applications of Artificial Intelligence*, 131:107779.

Nidhi Goyal, Jushaan Kalra, Charu Sharma, Raghava Mutharaju, Niharika Sachdeva, and Ponnurangam Kumaraguru. 2023. JobXMLC: EXtreme multi-label classification of job skills with graph neural networks. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 2181–2191, Dubrovnik, Croatia. Association for Computational Linguistics.

Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. *Preprint*, arXiv:1607.00653.

Bharathi Mohan Gurusamy, Prasanna Kumar Rangarajan, P. Krishh, A. Keerthinathan, G. Lavanya, Meka Meghana, Sheba Sulthana, and Srinath Doss. 2024. An analysis of large language models: their impact and potential applications. *Knowledge and Information Systems*, 66:1–24.

Md. Arid Hasan, Shudipta Das, Afiyat Anjum, Firoj Alam, Anika Anjum, Avijit Sarker, and Sheak Rashed Haider Noori. 2024. Zero- and few-shot prompting with LLMs: A comparative study with fine-tuned models for Bangla sentiment analysis. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 17808–17818, Torino, Italia. ELRA and ICCL.

Marvin Hofer, Daniel Obraczka, Alieh Saeedi, Hanna Köpcke, and Erhard Rahm. 2023. Construction of knowledge graphs: State and challenges.

X. Jia, Y. Zhang, and Y. Wang. 2018. Representation of job-skill in artificial intelligence with knowledge graph analysis. In *Proceedings of the 2018 IEEE International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*, pages 1–8.

Junshu Jiang, Songyun Ye, Wei Wang, Jingran Xu, and Xiaosheng Luo. 2020. Learning effective representations for person-job fit by feature fusion. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, CIKM '20, page 2549–2556, New York, NY, USA. Association for Computing Machinery.

Hamit Kavas, Marc Serra-Vidal, and Leo Wanner. 2024. Job offer and applicant cv classification using rich information from a labour market taxonomy. *Elsevier*. Available at SSRN: https://ssrn.com/abstract=4519766 or http://dx.doi.org/10.2139/ssrn.4519766.

O. Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. 2023. Dspy: Compiling declarative language model calls into self-improving pipelines. *ArXiv*, abs/2310.03714.

Jarosław Kurek, Tomasz Latkowski, Michał Bukowski, Bartosz Świderski, Mateusz Łępicki, Grzegorz Baranik, Bogusz Nowak, Robert Zakowicz, and Łukasz Dobrakowski. 2024. Zero-shot recommendation ai models for efficient job–candidate matching in recruitment process. *Applied Sciences*.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2021. What makes good in-context examples for gpt-3? In *Workshop on Knowledge Extraction and Integration for Deep Learning Architectures; Deep Learning Inside Out*.

Renming Liu, Matthew Hirn, and Arjun Krishnan. 2022. Accurately modeling biased random walks on weighted graphs using *Node2vec+*. *Preprint*, arXiv:2109.08031.

Yu Mao, Yuxuan Cheng, and Chunyu Shi. 2023. A job recommendation method based on attention layer scoring characteristics and tensor decomposition. *Applied Sciences*, 13:9464.

Tomas Mikolov, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *International Conference on Learning Representations*.

Khanh Cao Nguyen, Mike Zhang, Syrielle Montariol, and Antoine Bosselut. 2024. Rethinking skill extraction in the job market domain using large language models. *ArXiv*, abs/2402.03832.

Evgeniia Razumovskaia, Ivan Vuli'c, and Anna Korhonen. 2024. Analyzing and adapting large language models for few-shot multilingual nlu: Are we there yet? *ArXiv*, abs/2403.01929.

Liliya Sayfullina, Carlos Leiva, Erik Garcia, Xiang Niu, and Yang Zhao. 2018. Learning representations for soft skill matching. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2060–2071.

Elena Senger, Mike Zhang, Rob van der Goot, and Barbara Plank. 2024. Deep learning-based computational job market analysis: A survey on skill extraction and classification from job postings. In *Proceedings of the First Workshop on Natural Language Processing for Human Resources (NLP4HR 2024)*, pages 1–15, St. Julian's, Malta. Association for Computational Linguistics.

Roberta Sinatra, Jesús Gómez-Gardeñes, Renaud Lambiotte, Vincenzo Nicosia, and Vito Latora. 2010. Maximal-entropy random walks in complex networks with limited information. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 83 3 Pt 1:030103.

Eshaan Tanwar, Manish Borthakur, Subhabrata Dutta, and Tanmoy Chakraborty. 2023. Multilingual llms are better cross-lingual in-context learners with alignment. *ArXiv*, abs/2305.05940.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Multilingual e5 text embeddings: A technical report. *ArXiv*, abs/2402.05672.

Xuezhi Wang and Denny Zhou. 2024. Chain-of-thought reasoning without prompting. *Preprint*, arXiv:2402.10200.

Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. 2020. Scalable zero-shot entity linking with dense entity retrieval. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6397–6407, Online. Association for Computational Linguistics.

Xiao Yu, Jinzhong Zhang, and Zhou Yu. 2024. Confit: Improving resume-job matching using data augmentation and contrastive learning. *ArXiv*, abs/2401.16349.

Mike Zhang and et al. 2024. Entity linking in the job market domain. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 410–419, St. Julian's, Malta. Association for Computational Linguistics.

Mike Zhang, Kristian Jensen, Sif Sonniks, and Barbara Plank. 2022a. SkillSpan: Hard and soft skill extraction from English job postings. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4962–4984, Seattle, United States. Association for Computational Linguistics.

Qinggang Zhang, Junnan Dong, Keyu Duan, Xiao Huang, Yezi Liu, and Linchuan Xu. 2022b. Contrastive knowledge graph error detection.

Jing Zhao, Jingya Wang, Madhav Sigdel, Bopeng Zhang, Phuong Hoang, Mengshu Liu, and Mohammed Korayem. 2021. Embedding-based recommender system for job to candidate matching on scale. *ArXiv*, abs/2107.00221.