A Two-Stage Multilingual Job Title Matching System: Combining Expert Knowledge and LLM-based Ranking

Notebook for the TalentCLEF Lab at CLEF 2025

Mar Rodriguez, Olatz Perez-de-Viñaspre and Naiara Perez

HiTZ Basque Center for Language Technology - Ixa NLP Group, University of the Basque Country UPV/EHU

Abstract

This paper presents our participation in the TalentCLEF 2025 Shared Task A, which focuses on identifying and ranking job titles similar to a given query across English, German and Spanish. We propose and compare two approaches: (1) an end-to-end LLM-based baseline that performs both retrieval and ranking of job titles in a single step; and (2) a two-step pipeline that first retrieves candidates using the ESCO taxonomy, followed by semantic ranking with an LLM. Our experiments investigate the impact of various preprocessing techniques, including translation and normalization, as well as different retrieval configurations using sentence embeddings. Results show that combining ESCO-based filtering with LLM ranking, especially when using English as a pivot language, improves performance across languages.

Keywords

Job Title Matching, Information Retrieval, ESCO, LLM

1. Introduction

In recent years, the field of Human Resources (HR) has experienced significant transformation, largely driven by advances in Natural Language Processing (NLP) and the emergence of LLMs. These technologies have enabled the development of intelligent systems capable of processing large volumes of unstructured textual data, such as résumés and job descriptions, to identify candidates who best match specific job requirements [1, 2]. As these systems continue to evolve, they are increasingly being integrated into recruitment pipelines. However, practical deployment of NLP systems in HR contexts faces several key challenges [3]. These include multilingualism, ensuring fairness, mitigating bias, and achieving cross-sector adaptability. Multilingual systems must handle semantic differences across languages without losing domain-specific meanings; fairness is essential to prevent discrimination in system outputs; mitigating bias is necessary because training data can reflect or amplify existing societal biases; and cross-sector adaptability is also important, as job semantics vary significantly between professional domains.

In this context, the TalentCLEF 2025 Shared Task [3] Task A challenges participants to develop systems that identify and rank job titles most similar to a given query job title. For each job title in the provided test set, participants must generate a ranked list of similar titles drawn from a specified knowledge base (Figure 1). The task involves English, German and Spanish, making it a multilingual challenge. This paper presents our approach to this task, where we explore the application of Large Language Models (LLM) to the semantic matching of job titles. Our baseline strategy employs an LLM fully end to end, performing both retrieval and ranking of relevant job titles in a single pass. Given the practical challenges of processing large candidate sets directly with LLMs, our main exploration centers on a two-step pipeline: first, we apply a retrieval step based on similarity metrics to ESCO's taxonomy to reduce the candidate pool; then, we use an LLM to perform semantic ranking on the filtered results.

The remainder of this paper is structured as follows: Section 2 presents the datasets and supporting resources used for the task. Section 3 describes our methodology, covering data pre- and postprocessing

CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain

mar.rodriguez@ehu.eus (M. Rodriguez); olatz.perezdevinaspre@ehu.eus (O. Perez-de-Viñaspre); naiara.perez@ehu.eus (N. Perez)

0009-0008-6815-976X (M. Rodriguez); 0000-0002-0933-2461 (O. Perez-de-Viñaspre); 0000-0001-8648-0428 (N. Perez)

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

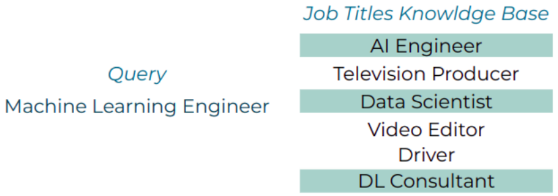


Figure 1: Example of TalentCLEF's Task A - Multilingual Job Title Matching, where the job titles of the knowledge base that match the query are highlighted in a green background. Source: https://talentclef.github.io/talentclef/docs/talentclef-2025/task-summary.

strategies, and our two proposed approaches. Section 4 details the experimental setup, and the results achieved by both systems. Finally, Section 5 discusses the findings, identifies current limitations, and outlines avenues for future research.

2. Data and Other Resources

In this section, we present the occupational classification taxonomies ISCO and ESCO, which served as foundational resources for performing the task. We also describe the dataset provided for the shared task, detailing its structure.

2.1. ISCO and ESCO

In order to carry out the task, ISCO[1] and ESCO[2] codes were provided and subsequently used as part of the methodology. The following subsection offers a brief overview of these codes, outlining their structure and relevance within the context of the task.

ISCO (International Standard Classification of Occupations) is a four-level classification of occupation groups (Major Group, Sub-Major Group, Minor Group and Unit Group). Each group is identified by a title and a numerical code, and is accompanied by a description that defines its scope:

- **Major Group** is denoted by a 1-digit code; e.g., 3 *Technicians and associate professionals*
- **Sub-Major Group** is denoted by a 2-digit code; e.g., 32 *Health associate professionals*
- **Minor Groups** are denoted by 3-digit codes; e.g., 322 *Nursing and midwifery associate professionals*
- **Unit Groups** are denoted by 4-digit codes; e.g., 3221 *Nursing associate professionals*

Each unit group consists of multiple occupations that are highly similar in both skill level and specialization. Since ISCO is a statistical classification, its occupation groups are mutually exclusive. This results in a strictly mono-hierarchical structure, in which every element at level 2 or below has exactly one parent group.

ESCO (European Skills, Competences, Qualifications and Occupations) is a multilingual classification system that works as a dictionary, describing, identifying and classifying professional occupations and skills relevant for the EU labour market. ESCO provides descriptions of 3,039 occupations and 13,939 skills linked to these occupations, translated into 28 languages (namely, all official EU languages plus Icelandic, Norwegian, Ukrainian, and Arabic). Figure 10 in Appendix A shows an example of an occupation at the ESCO level.

Each ESCO occupation is mapped to exactly one ISCO-08 code (i.e., the 2008 version of the ISCO classification). ISCO-08 can therefore be used as a hierarchical structure for the occupations pillar (Figure

[1] https://ilostat.ilo.org/methods/concepts-and-definitions/classification-occupation
[2] https://esco.ec.europa.eu

Table 1
Dataset composition by language and subset

Subset	File	English	Spanish	German
Training	Pairs of related jobs	28,880	20,724	23,023
Validation	Queries	105	185	203
	Corpus elements	2,619	4,661	4,729
Test	Queries	117	192	227
	Corpus elements	770	1,232	1,510

Table 2
A sample of the English training data

family_id	id	jobtitle_1	jobtitle_2
http://data.europa.eu/esco/isco/C2320	http://data.europa.eu/esco/occupation/d185c066-cddf-4a8c-a931-5eefc21fc2a1	trainer of cabin crew	flight attendant trainer
http://data.europa.eu/esco/isco/C2320	http://data.europa.eu/esco/occupation/d185c066-cddf-4a8c-a931-5eefc21fc2a1	inflight service instructor	cabin crew instructor
http://data.europa.eu/esco/isco/C2320	http://data.europa.eu/esco/occupation/d185c066-cddf-4a8c-a931-5eefc21fc2a1	cabin crew trainer	flight attendant instructor
http://data.europa.eu/esco/isco/C2320	http://data.europa.eu/esco/occupation/d185c066-cddf-4a8c-a931-5eefc21fc2a1	flight attendant trainer	flight service instructor
http://data.europa.eu/esco/isco/C2320	http://data.europa.eu/esco/occupation/d185c066-cddf-4a8c-a931-5eefc21fc2a1	flight attendant trainer	cabin crew instructor
http://data.europa.eu/esco/isco/C2320	http://data.europa.eu/esco/occupation/d185c066-cddf-4a8c-a931-5eefc21fc2a1	instructor of cabin crew	cabin service instructor

11 in Appendix A). ESCO occupations are located at level 5 and lower of the ISCO-08 classification. Some groups of ISCO-08 do not contain ESCO occupations, typically because they represent roles without relevant economic activity in the EU, such as *water* and *firewood collectors*.

2.2. Dataset description

The corpus consists of job titles in three languages: English, German and Spanish. And it covers a wide range of job domains and professional sectors. It is divided into three subsets: training, validation, and test. Table 1 summarizes the number of pairs, queries, target corpora, and labeled relationships across languages and subsets. This multilingual and multi-sector dataset enables robust evaluation of job title similarity methods across both linguistic and professional boundaries. The training data has been compiled using publicly available terminologies. In contrast, both the validation and test sets were annotated by domain experts.

Training Set. For each language involved in the task, a corresponding training dataset is provided in a tabular format (see Table 2), consisting of four columns. The *family_id* column contains the ISCO family identifier, which represents the occupational group to which the job titles belong; the *id* column includes the ESCO identifier indicating the source of the job title pair; and the remaining two columns, *jobtitle_1* and *jobtitle_2*, represent pairs of related job titles, with *jobtitle_2* being semantically or functionally related to *jobtitle_1*.

Validation and Test Set. They are organized into two separate files for each language considered in the task: one for "queries" and one for "corpus elements". The queries file (see Tables 3a and 4a) includes a unique identifier for each query (*q_id*) along with the corresponding job title used as the query (*jobtitle*). The corpus elements file (Tables 3b and 4b) similarly contains a unique identifier (*c_id*) for each element, as well as the associated job title present in the corpus (*jobtitle*).

Table 3
A sample of the English validation set

(a) Queries

q_id	jobtitle
1	nanny
2	food technologist
3	broadcast engineer
4	automation engineer
5	veterinarian
6	loan officer

(b) Corpus elements

c_id	jobtitle
484	university instructor
485	mechanical design engineer
486	social media specialist
487	account manager
488	social media manager
489	digital design engineer

Table 4
A sample of the English test set

(a) Queries

q_id	jobtitle
173017	Mobile Engineer - HVAC/R
730648	Regional Sales Manager
931772	RA Specialist in Medical Devices
878776	Field Representative II, Field Services Support
989207	Finance Analyst
276992	Building Engineer (Mobile)

(b) Corpus elements

c_id	jobtitle
633313	Product Owner Senior Staff Portfolio Leader
494356	Quality Assurance Assistant Manager
202967	Senior Principal Engineer Systems Architect
402004	Senior Engineer
671472	Production Planning Engineer
165965	Program Officer, International Health

3. Methodology

We developed two distinct methodological approaches for the multilingual job title matching task. Our primary approach employs a two-step pipeline (System 1, Subsection 3.3) that first applies ESCO taxonomy-based retrieval to filter candidates, followed by LLM-based semantic ranking. This design addresses the computation constraints imposed by LLM context length limitations when processing large candidate sets directly. As a comparative baseline, we also implemented an end-to-end approach (System 2, Subsection 3.3) that performs both retrieval and ranking in a single LLM pass, though this was only feasible for the smaller test set due to input size constraints. Both systems incorporate preprocessing steps for text normalization and optional translation (Subsection 3.1), with postprocessing steps to handle multilingual label mapping (Subsection 3.4).

3.1. Preprocessing

The text preprocessing stage involved normalization, language standardization and, in some cases, translation. First, known abbreviations such as *QA*, *Sr*, and *AVP* were expanded to their full forms; roman numerals commonly used to indicate seniority levels (e.g., *I*, *II*, *III*) were converted into standardized terms such as *Junior*, *Intermediate*, and *Senior*; the placement of such modifiers was also swapped to follow correct English syntax, for instance, transforming *Engineer Senior* into *Senior Engineer*; unknown acronyms were preserved in uppercase to maintain their distinctiveness; and extraneous punctuation was removed and whitespace normalized.

Regarding the optional translation component of preprocessing, job titles in Spanish and German were translated into English using `Claude 3.7 Sonnet`[3] to facilitate cross-lingual comparison and leverage the generally superior performance of NLP models in English. This translation strategy also provides a mechanism for reducing gender bias inherent in languages with grammatical gender. For example, in Spanish, the titles *Vicepresidente, gerente sénior de planificación de capital* (masculine) and *Vicepresidenta, gerente sénior de planificación de capital* (feminine) both translate to *Vice President, Senior Capital Planning Manager* in English, where the gender distinction that could bias the matching process is entirely neutralized.

[3] https://www.anthropic.com/claude/sonnet

3.2. System 1: Retrieval and Ranking

This approach employs a two-step pipeline designed to narrow down and order candidate job titles. Initially, a retrieval phase uses similarity metrics derived from ESCO and ISCO taxonomies to select a manageable subset of relevant candidates. Subsequently, an LLM performs filtering and fine-grained ranking on this set. The following subsections describe in detail the retrieval, ranking, and additional filtering strategies implemented in this system.

3.2.1. Retrieval

To perform the initial retrieval of job titles, we leveraged the ESCO and ISCO taxonomies as semantic pivots to identify the most relevant codes for each query and corpus element. We seek to map job titles from different languages and domains to a standardized occupational classification, as it should facilitate cross-lingual and cross-domain matching through shared taxonomic representations. For each job title (query or corpus element), then, we first computed the similarity scores against all taxonomy codes using several methods:

- **Levenshtein distance** [4] as a simple string similarity baseline.
- **Sentence-BERT (sBERT) embeddings** [5], using `all-MiniLM-L6-v2`[4] for English texts and `distiluse-base-multilingual-cased-v1`[5] [6] for German and Spanish.
- **Flair-based document embeddings** [7], combining static word embeddings (GloVe [8] for English; FastText [9] for German and Spanish) with bidirectional Flair embeddings through mean pooling to obtain a fixed-size vector for each job title, following the recommended recipe.[6]
- **RoBERTa-based embeddings**, using `RoBERTa base`[7] [10] for English and `XLM-R`[8] [11] for the other languages.

Based on these similarity scores, we next kept codes with perfect matches, if any, or selected up to 20 most relevant codes for each job title using one of three filtering strategies:

- A fixed similarity **threshold**, below which codes are discarded.
- A **ratio**-based approach, selecting codes within a given percentage of the highest similarity score.
- A **gap**-based strategy, which selects codes until a significant drop in similarity between consecutive codes is detected.

Having thus mapped queries and corpus elements to ESCO and ISCO codes, the retrieval finally consists in selecting, for each query, all corpus elements that share at least one ESCO or ISCO code with the query. This creates a premiliminary semantically filtered candidate set informed by expert knowledge for subsequent LLM-based ranking.

3.2.2. Filtering and Ranking

While the previous step significantly reduces the candidate pool, the resulting sets still contain too many job titles that require further filtering and precise ranking. However, these filtered candidate sets are now manageable in size for processing by LLMs with sufficient context capacity.

We specifically experimented with three different prompts (Figure 2) on the `Llama 3.3 70B Instruct`[9] [12] model, selected for its strong demonstrated multilingual performance. We deployed the model in 2 A100 GPUs using vLLM [13] and applied the model's default generation hyperparameters, except for temperature, which was set to 0.7 instead of 0.6 to encourage more diverse output. The first

[4] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

[5] https://huggingface.co/sentence-transformers/distiluse-base-multilingual-cased-v1

[6] https://flairnlp.github.io/docs/tutorial-embeddings/other-embeddings#document-pool-embeddings

[7] https://huggingface.co/FacebookAI/roberta-base

[8] https://huggingface.co/FacebookAI/xlm-roberta-base

[9] https://huggingface.co/meta-llama/Llama-3.3-70B-Instruct

Ranking Prompt 1 — Relevance

I have this list of professions separated by ';'. I want to sort the professions related to '`{query_label}`', discarding those that are not as closely related. The list should be useful for someone who works as a '`{query_label}`' and is looking for another job. I just want the list, without any explanation. I want the list separated by ';'. `{corpus_labels}`.

Ranking Prompt 2 — Career counselor

Analyze these professions: `{corpus_labels}`. As an expert career counselor, sort them by relevance for a `{query_label}` considering: 1. Skill similarity 2. Industry proximity 3. Career progression paths. Return only the sorted list separated by semicolons.

Ranking Prompt 3 — Skill transfer

For a `{query_label}` considering a career change, rank these options by transferability of skills: `{corpus_labels}`. Most transferable skills first. Only return semicolon-separated list.

Figure 2: Prompts tested for the LLM-based ranking step (System 1)

End-to-End Prompt

I have this list of professions separated by ';'. I want to sort the professions by relevance to '`{query_label}`', and discard those that are not as closely related. This list should be useful for someone who works as a '`{query_label}`' and is looking for another job. I just want the list, without any explanation. Do not add new job titles nor rephrase the ones I gave. Simply discard irrelevant job postings and rerank the relevant ones. I want the list separated by ';'. Here is the original list: `{corpus_labels}`.

Figure 3: Prompt used in the end-to-end baseline (System 2)

prompt ("Relevance") was crafted by the authors, while the second ("Career counselor") and third ("Skill transfer") were proposed by GPT-4o[10] through ChatGPT.[11]

3.3. System 2: End-to-End Baseline

In this simpler approach, the entire task is handled in a single step by an LLM. This approach was applied exclusively to the test set, as the size of the validation set exceeded the input length limit of the Llama 3.3 70B Instruct. Another limitation is that, for Spanish and German, we were only able to test the translated (English) version. The model's tokenizer has greater fertility in non-English languages, making it impossible to fit the original data within the input size constraints.

Furthermore, this baseline does not include a retrieval phase based on ESCO similarity, unlike System 1. Instead, the model receives the full list of corpus job titles directly as input, along with the query title, and is prompted to return a cleaned and ranked list of relevant results. To ensure comparability, all job titles were preprocessed and translated into English using the same steps described in Subsection 3.1.

To better control the model's output, we refined the initial prompts after observing that previous versions occasionally introduced new job titles not present in the input list. The refined prompt (see Figure 3) explicitly instructs the model not to add or rephrase any job titles and to strictly filter out irrelevant ones while reranking the remaining titles by their relevance to the query.

3.4. Postprocessing

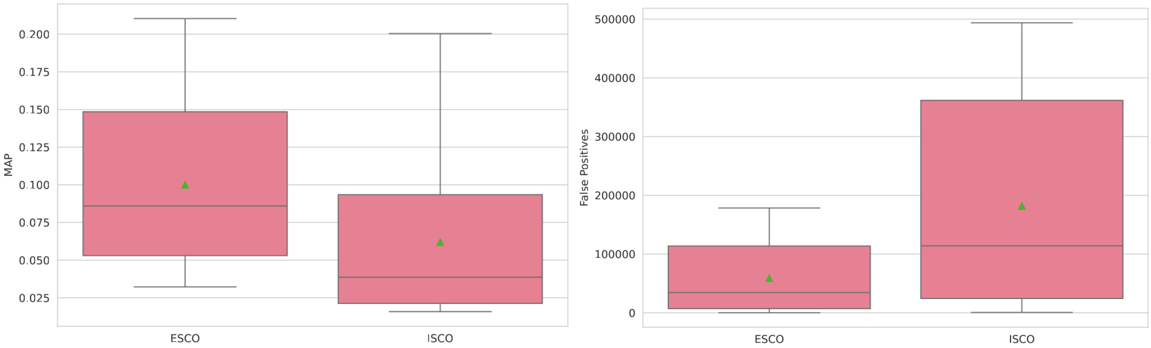
Even after LLM-based filtering and ranking, the output lists could still be excessively long for practical use. When the ranked list exceeded 50 job titles—a threshold determined based on development data distributions—we applied an additional filtering step: we trimmed the list to the top 100 candidates and re-invoked the LLM to obtain a more focused ranking. For Spanish and German datasets, if translation had been applied, we then mapped the English job titles back to their original languages. This step

[10] https://openai.com/index/gpt-4o-system-card

[11] https://chatgpt.com

Table 5
Example from the Spanish-to-English translated test set: mapping of original Spanish job titles and their codes to the shared English label.

Spanish Job Titles	Code	Label Translated to English
Vicepresidente de desarrollo de negocios Vicepresidenta de desarrollo de negocios	712384 222830	Business Development Vice President



(a) MAP

(b) False Positives

Figure 4: Performance of sBERT-based retrieval with ESCO and ISCO taxonomies as pivots. The figure includes aggregated results across the three languages—English, German, and Spanish.

often resulted in list expansion, as a single English label could correspond to multiple original-language variants differing in grammatical gender or phrasing (see Table 5). Next, we mapped the output job titles back to their corresponding corpus codes, since the LLM input contained only candidate job title text. We retained only unique codes to eliminate duplicates resulting from this process. Finally, if the resulting code list still remained excessively long, particularly due to the gender-based expansion in German and Spanish, we arbitrarily retained only the first 80 codes.

4. Experiments and Results

In this section, we present the results for both the validation set (Subsection 4.1) and the test set (Subsection 4.2). The validation results are used to justify the choices made for the final systems submitted to the shared task. We discuss results in terms of precision (or false positives), recall (or false negatives), and Mean Average Precision (MAP), which measures the quality of the ranked lists of job titles by considering both precision and recall at multiple cutoff points.

4.1. Validation Experiments and Results

For System 1, we first explored the full set of possible combinations of the strategies for the **retrieval phase**, as explained in Subsection 3.2. In what follows, we report the most impactful results. It must be noted that, in this step, it is crucial to maximize recall; as this step retrieves the candidates that the LLM will process. At the same time, however, we need to minimize false positives so that the LLM input fits within token limits.

Pivot taxonomy. Figures 4a and 4b compare the performance of ISCO and ESCO as pivots in terms of MAP and false positive rates obtained with sBERT embeddings. As it can be seen, using the **ESCO** taxonomy is more beneficial than using the ISCO taxonomy, as it particularly helps reduce the false positives. Thus, we only report results using ESCO henceforth.

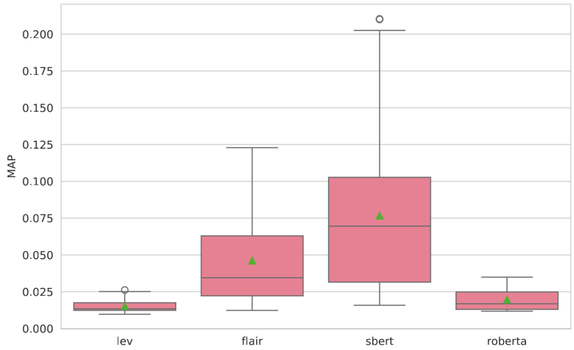


Figure 5: MAPs obtained with different similarity measuring methods. The figure includes aggregated results across the three languages—English, German, and Spanish.

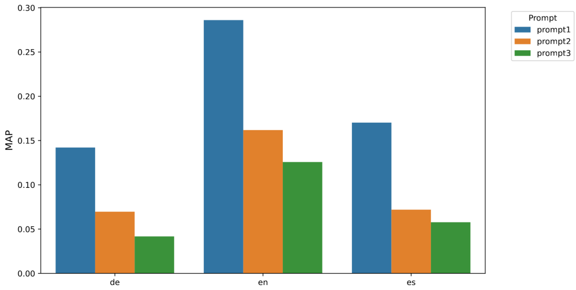


Figure 6: MAPs obtained with different ranking prompts, broken down by language.

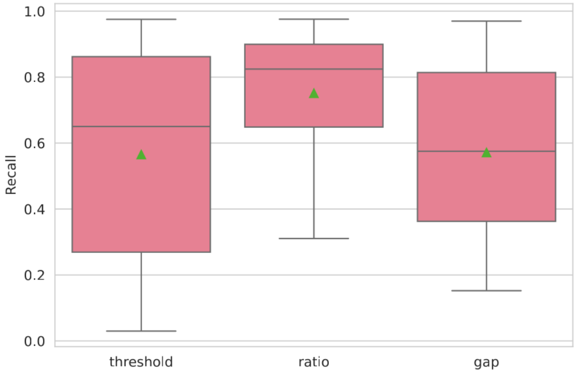


Figure 7: Recall of filtering strategies with sBERT for the validation set. The figure includes results from all entries across the three languages (English, German and Spanish).

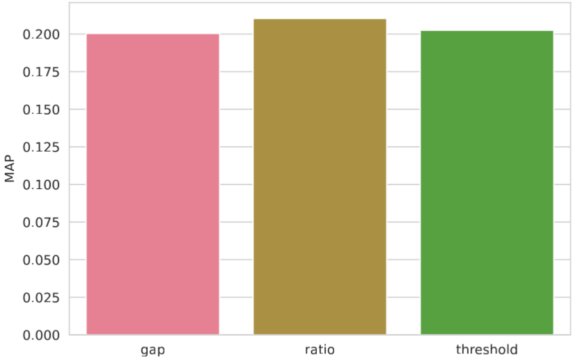


Figure 8: MAP of filtering strategies with sBERT for the validation set. The figure includes results from all entries across the three languages (English, German and Spanish).

Embeddings. Figure 5 shows the MAP results obtained with the different tested methods to measure the similarity between job titles and ESCO labels. We observe that **sBERT** clearly outperformed the other methods, followed by Flair. RoBERTa-based embeddings yielded extremely poor results, as did the Levenshtein distance. Hence, in what follows, we only report results with sBERT.

Filtering strategy. Figures 7 and 8 show the recall and MAP, respectively, of the different strategies. In this case, the **ratio-based** method provided the best trade-off between recall and precision, compared to the fixed value threshold and the gap-based strategy. We further experimented with various ratio values to assess their impact on retrieval performance (Table 6). Lower ratios, such as 0.2 and 0.4, achieved good recall while keeping false positives manageable. However, they tended to be overly permissive. Among them, 0.4 slightly outperformed 0.2. In contrast, higher ratios like 0.8 led to a sharp drop in recall. Mid-range values, particularly 0.6 and 0.7, offered a better balance between recall and precision, with 0.7 showing a slight advantage. Based on these findings, we selected **0.4** and **0.7** for our final configuration. Detailed results for all ratios using sBERT across languages are provided in Appendix A, Table 11.

Having optimized the retrieval phase, we next explored the **ranking phase**, were our only hyperparameter is the prompt passed to Llama 3.3 70B Instruct. In this exploration, we experimented with 4 ratio values for a wider view of the options and the effect of limiting the recall: 0.4, 0.6, 0.7 and 0.8.

Table 6
MAP, Precision, and Recall scores for the different filtering strategies and their corresponding values, evaluated on the English validation set using the sBERT model.

Strategy	Value	MAP	Precision	Recall
threshold	0.4	0.0761	0.0566	0.9066
threshold	0.5	0.0865	0.0601	0.8955
threshold	0.6	0.1228	0.0772	0.8306
threshold	0.7	**0.2025**	0.1616	0.6409
threshold	0.8	0.1998	0.4644	0.2988
threshold	0.9	0.0805	0.5974	0.0748
ratio	0.4	0.0762	0.0565	0.9074
ratio	0.5	0.0854	0.0578	0.9012
ratio	0.6	0.1075	0.0663	0.8727
ratio	0.7	0.1629	0.0924	0.8161
ratio	0.8	**0.2104**	0.1776	0.6128
ratio	0.9	0.2100	0.3742	0.3831
gap	0.01	0.1580	0.5668	0.1963
gap	0.025	0.1582	0.4458	0.2599
gap	0.05	**0.1768**	0.2353	0.3930
gap	0.1	0.1729	0.1140	0.6095
gap	0.2	0.1231	0.0707	0.8273

Table 7
MAP scores for the different prompts and ratios across languages for the validation set

Prompt	Ratio	Avg. MAP	MAP en	MAP es	MAP de
Prompt 1	0.4	0.175	0.257	0.136	0.131
	0.6	0.191	0.268	0.157	0.148
	0.7	0.208	0.304	0.179	0.141
	0.8	**0.224**	**0.314**	**0.210**	**0.148**
Prompt 2	0.4	0.090	0.140	0.056	0.073
	0.6	0.082	0.125	0.053	0.067
	0.7	0.103	0.176	0.071	0.062
	0.8	0.130	0.207	0.107	0.077
Prompt 3	0.4	0.049	0.079	0.035	0.032
	0.6	0.057	0.100	0.037	0.034
	0.7	0.078	0.137	0.056	0.042
	0.8	0.116	0.187	0.103	0.059

Ranking prompt. Figure 6 shows the impact of each prompt, broken down by language, with Prompt 1 producing the most accurate and relevant results. Table 7 presents the corresponding MAP scores for different prompt configurations and ratio thresholds across languages. Another important factor was execution time. Prompt 1 was consistently faster across all cases, typically requiring between 30 minutes and 3.5 hours, depending on the ratio and language (see Table 12 in Appendix A). In contrast, Prompt 2 generally ranged from 1.5 to 9.5 hours, while Prompt 3 required between 3.5 and over 14 hours. Due to these limitations, we applied **Prompt 1** exclusively in our final system.

In this final experiments over the validation dataset, the last ranking phase using LLMs showed a beneficial effect, as we obtain an increase of more than 0.1 in the MAP metric.

Regarding the preprocessing step (Subsection 3.1), no validation results are available, as it was applied exclusively to the test set. This decision stems from the fact that the need for normalization arose when we observed significant discrepancies between the test data and the formats found in official ISCO and ESCO taxonomies. Likewise, job title translation into English was incorporated for two main reasons: (1) English yielded better performance in preliminary experiments, and (2) translation helped reduce issues related to grammatical gender present in languages like Spanish and German. Due to time constraints, we could not apply these enhancements to the validation set, and their impact was only evaluated during the final testing phase.

Table 8
Configuration summary of the five experimental runs conducted on the test set

Run	Translation	Normalization	Retrieval Model	Ratio	Final Ranking
Run1	–	Yes	sBERT	0.7	`LLaMA-3.3-70B-Instruct`
Run2	`Claude 3.7 Sonnet` (to en)	Yes	sBERT	0.7	`LLaMA-3.3-70B-Instruct`
Run3	–	Yes	sBERT	0.4	`LLaMA-3.3-70B-Instruct`
Run4	`Claude 3.7 Sonnet` (to en)	Yes	sBERT	0.4	`LLaMA-3.3-70B-Instruct`
Run5	`Claude 3.7 Sonnet` (to en)	Yes	–	–	`LLaMA-3.3-70B-Instruct`

Table 9
MAP scores for the test set across different runs and language pairs

Run ID	Avg. MAP (en, es, de)	MAP(en-en)	MAP(es-es)	MAP(de-de)
run2 (translation + sBERT + ratio 0.7 + LLM)	**0.18**	0.199	**0.173**	**0.169**
run4 (translation + sBERT + ratio 0.4 + LLM)	**0.18**	**0.200**	0.166	0.168
run5 (translation + LLM)	0.15	0.159	0.150	0.150
run1 (sBERT + ratio 0.7 + LLM)	0.14	0.199	0.122	0.110
run3 (sBERT + ratio 0.4 + LLM)	0.14	**0.200**	0.115	0.106

Table 10
Sample from predicted job titles for the query *reliability engineer* using ratio 0.7 and sBERT from English test set

query_id	query_label	similar_job_titles	shared_ESCO_codes
130916	reliability engineer	supplier development engineer	2141.8
130916	reliability engineer	engineer junior	2141.8
130916	reliability engineer	computational engineering student	2141.8
130916	reliability engineer	product development engineer	2141.8
130916	reliability engineer	technical writer manufacturing systems department	2141.8
130916	reliability engineer	process manufacturing engineer	2141.8
130916	reliability engineer	engineer technical operations	2141.8
130916	reliability engineer	senior manager application maintenance and support	2141.8
130916	reliability engineer	senior site reliability engineer	2141.8, 2149.7, 2512.7
130916	reliability engineer	principal engineer	2141.8

4.2. Test Results

Based on the results for the validation set outlined in the previous subsection, we conducted five distinct experimental runs on the test set by implementing different combinations of preprocessing steps, translation, filtering ratios, and ranking strategies. These runs were designed to evaluate the impact of various configurations on the system's performance in identifying and ranking job titles across English, German and Spanish. The five runs performed are as follows (Table 8):

In Table 9, the MAP scores obtained for each run across the main monolingual language pairs can be seen: English-English (en-en), Spanish-Spanish (es-es), and German-German (de-de). Runs that include translation tend to perform better on Spanish and German, highlighting the benefit of cross-lingual alignment through English. Notably, Run2 and Run4 achieve the highest average MAP scores across the three languages, indicating that combining translation with sBERT retrieval and LLM ranking yields superior results. Run5, which skips sBERT retrieval, exhibits markedly poor performance and is consistently outperformed by the runs that include filtering. In Table 10 a sample of the predicted most similar job titles for the query *reliability engineer* is presented.

We observe a significant drop in performance on the test set compared to the validation set. Furthermore, the influence of the ratio used to select ESCO codes appears less pronounced in the test set. To investigate this discrepancy, we analyzed the classification of ESCO codes and found that it was less effective than in the validation set. Specifically, similarity scores were generally lower, particularly for the most similar codes. Figure 9 presents violin plots comparing the distributions of similarity scores for both sets. The results indicate that the sBERT-based similarity scores align more closely with the

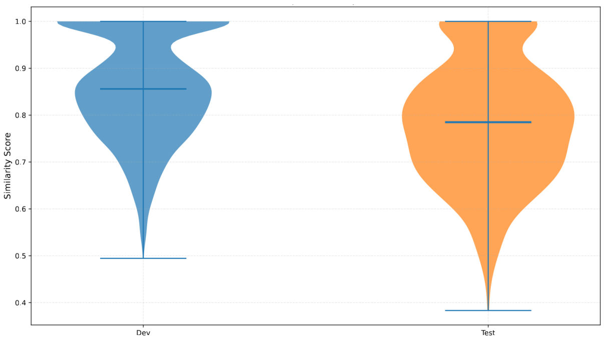


Figure 9: sBERT distributions of the codes for each of the sets (validation and test)

ESCO taxonomy in the validation set than in the test set. Consequently, our method may not be the most suitable approach for the current test set.

5. Discussion and Future Work

Our initial strategy was using LLMs in a fully end-to-end manner, performing both filtering and ranking of relevant job titles. However, due to the context-length limitations of current LLMs and the size of the validation data, we evaluated this direct end-to-end approach only on the test set. The other approach we explored was a two-step pipeline: first, we applied a pre-filtering phase based on ESCO taxonomy similarity to reduce the candidate pool; then, we used an LLM to perform semantic ranking on the filtered results.

Although the MAP scores obtained on the test set were generally modest, the methodology we followed proved to be the most effective among the configurations tested. Particularly, the two-step approach combining ESCO-based retrieval with LLM-based ranking. The incorporation of semantic filtering using sBERT embeddings provided a strong foundation for narrowing down the candidate set, significantly outperforming simpler approaches such as Levenshtein distance or document-level embeddings based on Flair and RoBERTa.

One of the most impactful design choices was the decision to translate Spanish and German job titles into English prior to further processing. This translation step not only improved MAP scores in the corresponding language pairs, but also contributed to mitigating gender bias inherent in languages with grammatical gender. As discussed in Section 3.4, English helped neutralize gender-specific job titles by lacking this grammatical feature. Nonetheless, the translation served as a valuable cross-lingual normalization mechanism that enhanced the LLM's ability to generalize across languages.

The analysis of filtering strategies revealed that ratio-based selection was the most balanced among the three evaluated alternatives. Unlike fixed thresholds or abrupt similarity gaps, ratio-based filtering allowed for dynamic cutoffs that preserved both precision and recall. Empirical results showed that a ratio of 0.7 offered a particularly good trade-off, leading to better final rankings than more permissive (0.4) or restrictive (0.8) values. Moreover, sBERT emerged as the most robust method for computing semantic similarity, justifying its exclusive use in the final configurations.

Regarding the ranking phase, the experiments confirmed the utility of prompting LLMs to reorder and refine candidate lists. Prompt 1 not only delivered better alignment with the query job title but also operated more efficiently in terms of execution time compared to the other alternatives. Even in the end-to-end baseline, the LLM proved capable of performing effective filtering and ranking, though less reliably than when supported by ESCO-based filtering.

While our approach proved effective within the tested configurations for the validation set, it may have relied too heavily on the ESCO taxonomy as a backbone for candidate filtering. This dependence might have constrained the system's flexibility and limited the exploration of alternative methods for initial retrieval, such as using multilingual semantic search directly. Additionally, although we had prepared the necessary code to support cross-lingual configurations (e.g., EN–ES, EN–DE), time constraints prevented us from running these experiments. As a result, our evaluation was restricted to monolingual settings.

Future work could work on relaxing the dependency on ESCO by investigating multilingual embedding-based retrieval strategies and assessing system performance in cross-lingual scenarios. Further exploration into end-to-end LLM approaches with improved context management may also unlock more streamlined solutions.

Acknowledgments

We acknowledge the support of the HiTZ Chair of Artificial Intelligence and Language Technology (TSI100923-2023-1), funded by MTDFP, Secretaría de Estado de Digitalización e Inteligencia Artificial, ENIA, and by the European Union-Next Generation EU / PRTR; the Spanish Ministry for Digital Transformation and of Civil Service, and the EU-funded NextGenerationEU Recovery, Transformation and Resilience Plan (*ILENIA, 2022/TL22/00215335*); *Disargue (TED2021-130810B-C21)* project (funded by MCIN/AEI /10.13039/501100011033 and European Union NextGeneration EU/PRTR) and the *TRAIN (PID2021-123988OB-C31)* project (funded by MCIN/AEI /10.13039/501100011033 and "ERDF A way of making Europe").

Declaration on Generative AI

During the preparation of this work, the author(s) used GPT-4 in order to: Grammar and spelling check. After using these tool(s)/service(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] C. Qin, L. Zhang, Y. Cheng, R. Zha, D. Shen, Q. Zhang, X. Chen, Y. Sun, C. Zhu, H. Zhu, et al., A comprehensive survey of artificial intelligence techniques for talent analytics, arXiv preprint arXiv:2307.03195 (2023).

[2] N. Otani, N. Bhutani, E. Hruschka, Natural language processing for human resources: A survey, in: W. Chen, Y. Yang, M. Kachuee, X.-Y. Fu (Eds.), Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 3: Industry Track), Association for Computational Linguistics, Albuquerque, New Mexico, 2025, pp. 583–597. URL: https://aclanthology.org/2025.naacl-industry.47/.

[3] L. Gasco, H. Fabregat, L. García-Sardiña, P. Estrella, D. Deniz, A. Rodrigo, R. Zbib, Overview of the TalentCLEF 2025 Shared Task: Skill and Job Title Intelligence for Human Capital Management, in: International Conference of the Cross-Language Evaluation Forum for European Languages, Springer, 2025.

[4] V. I. Levenshtein, et al., Binary codes capable of correcting deletions, insertions, and reversals, in: Soviet physics doklady, volume 10, Soviet Union, 1966, pp. 707–710.

[5] N. Reimers, I. Gurevych, Sentence-BERT: Sentence embeddings using Siamese BERT-networks, in: K. Inui, J. Jiang, V. Ng, X. Wan (Eds.), Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3982–3992. URL: https://aclanthology.org/D19-1410/. doi:10.18653/v1/D19-1410.

[6] N. Reimers, I. Gurevych, Making monolingual sentence embeddings multilingual using knowledge distillation, in: B. Webber, T. Cohn, Y. He, Y. Liu (Eds.), Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online, 2020, pp. 4512–4525. URL: https://aclanthology.org/2020.emnlp-main.365/. doi:10.18653/v1/2020.emnlp-main.365.

[7] A. Akbik, D. Blythe, R. Vollgraf, Contextual string embeddings for sequence labeling, in: E. M. Bender, L. Derczynski, P. Isabelle (Eds.), Proceedings of the 27th International Conference on Computational Linguistics, Association for Computational Linguistics, Santa Fe, New Mexico, USA, 2018, pp. 1638–1649. URL: https://aclanthology.org/C18-1139/.

[8] J. Pennington, R. Socher, C. Manning, GloVe: Global vectors for word representation, in: A. Moschitti, B. Pang, W. Daelemans (Eds.), Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Doha, Qatar, 2014, pp. 1532–1543. URL: https://aclanthology.org/D14-1162/. doi:10.3115/v1/D14-1162.

[9] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, Enriching word vectors with subword information, Transactions of the Association for Computational Linguistics 5 (2017) 135–146. URL: https://aclanthology.org/Q17-1010/. doi:10.1162/tacl_a_00051.

[10] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, RoBERTa: A robustly optimized BERT pretraining approach, arXiv preprint arXiv:1907.11692 (2019).

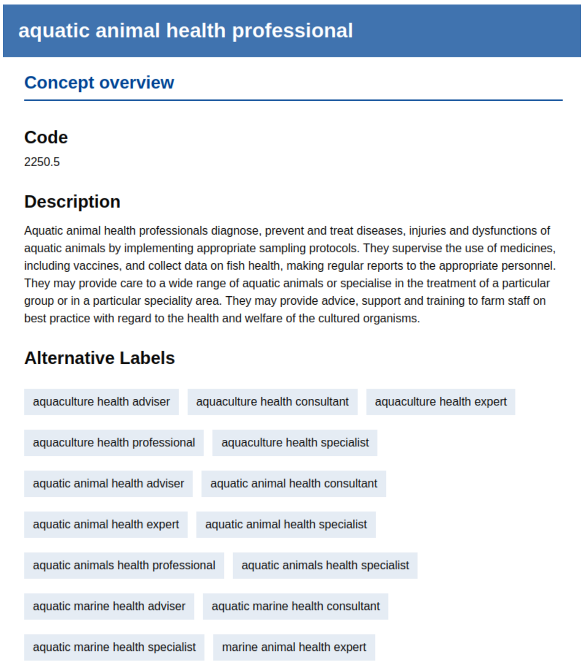
[11] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, in: D. Jurafsky, J. Chai, N. Schluter, J. Tetreault (Eds.), Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 8440–8451. URL: https://aclanthology.org/2020.acl-main.747/. doi:10.18653/v1/2020.acl-main.747.

[12] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, et al., The llama 3 herd of models, arXiv preprint arXiv:2407.21783 (2024).

[13] W. Kwon, Z. Li, S. Zhuang, Y. Sheng, L. Zheng, C. H. Yu, J. Gonzalez, H. Zhang, I. Stoica, Efficient memory management for large language model serving with pagedattention, in: Proceedings of the 29th Symposium on Operating Systems Principles, SOSP '23, Association for Computing Machinery, New York, NY, USA, 2023, p. 611–626. URL: https://doi.org/10.1145/3600006.3613165. doi:10.1145/3600006.3613165.

A. Appendix

This appendix provides additional resources to support the main content of the paper. It includes two figures illustrating the structure and content of the ESCO and ISCO taxonomies: an example occupation entry with its ESCO code, description, and alternative labels, and an overview of the occupations pillar hierarchy. Additionally, two tables report extended experimental details: one presents the full evaluation metrics of the sBERT model across languages and ratio-based thresholds for the validation set, and the other details execution times for each prompt configuration across languages and ratios using the validation set.



aquatic animal health professional

Concept overview

Code

2250.5

Description

Aquatic animal health professionals diagnose, prevent and treat diseases, injuries and dysfunctions of aquatic animals by implementing appropriate sampling protocols. They supervise the use of medicines, including vaccines, and collect data on fish health, making regular reports to the appropriate personnel. They may provide care to a wide range of aquatic animals or specialise in the treatment of a particular group or in a particular speciality area. They may provide advice, support and training to farm staff on best practice with regard to the health and welfare of the cultured organisms.

Alternative Labels

aquaculture health adviser | aquaculture health consultant | aquaculture health expert | aquaculture health professional | aquaculture health specialist | aquatic animal health adviser | aquatic animal health consultant | aquatic animal health expert | aquatic animal health specialist | aquatic animals health professional | aquatic animals health specialist | aquatic marine health adviser | aquatic marine health consultant | aquatic marine health specialist | marine animal health expert

Figure 10: figure

Example of an occupation at the ESCO level, including its ESCO code, description, and alternative labels. Source: https://esco.ec.europa.eu/en/classification/occupation_main

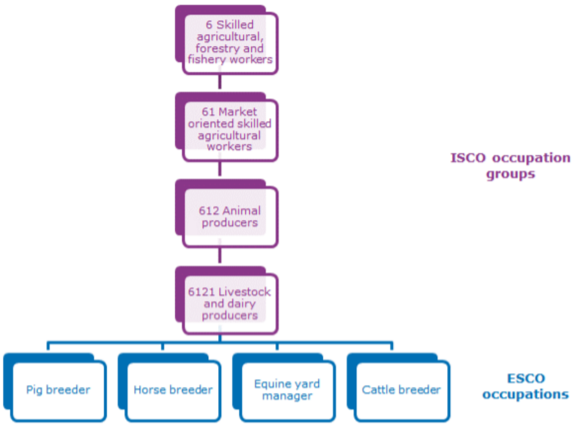

6 Skilled agricultural, forestry and fishery workers
61 Market oriented skilled agricultural workers
612 Animal producers
6121 Livestock and dairy producers
ISCO occupation groups
Pig breeder
Horse breeder
Equine yard manager
Cattle breeder
ESCO occupations

Figure 11: figure

Structure of the occupations pillar hierarchy. Source: https://esco.ec.europa.eu/en/about-esco/escopedia/escopedia/international-standard-classification-occupations-isco

Table 11

Performance of the sBERT model across languages (English, German and Spanish) and different ratio-based filtering thresholds for the validation set. Reported metrics include Precision, Recall, F1-score, True Positives (TP), Duplicates (DP), False Negatives (FN), and MAP.

Language	Ratio	Precision	Recall	F1	TP	FP	FN	MAP
en	0.4	0.0565	0.9074	0.1064	2196	36657	224	0.0762
	0.5	0.0578	0.9012	0.1086	2181	35568	239	0.0854
	0.6	0.0663	0.8727	0.1232	2112	29764	308	0.1075
	0.7	0.0924	0.8161	0.1660	1975	19407	445	0.1629
	0.8	0.1776	0.6128	0.2754	1483	6866	937	0.2104
	0.9	0.3742	0.3831	0.3786	927	1550	1493	0.2100
es	0.4	0.0451	0.7506	0.0852	5689	120351	1890	0.0486
	0.5	0.0458	0.7489	0.0864	5676	118162	1903	0.0503
	0.6	0.0510	0.7224	0.0953	5475	101862	2104	0.0566
	0.7	0.0712	0.6628	0.1286	5023	65530	2556	0.0912
	0.8	0.1373	0.5042	0.2159	3821	23999	3758	0.1513
	0.9	0.3251	0.3126	0.3187	2369	4917	5210	0.1853
de	0.4	0.0311	0.6801	0.0595	5724	178420	2693	0.0332
	0.5	0.0311	0.6801	0.0595	5724	178178	2693	0.0337
	0.6	0.0316	0.6783	0.0605	5709	174718	2708	0.0342
	0.7	0.0352	0.6387	0.0667	5376	147406	3041	0.0423
	0.8	0.0468	0.5118	0.0858	4308	87742	4109	0.0840
	0.9	0.0972	0.3107	0.1481	2615	24293	5802	0.1206

Table 12
Execution time (hh:mm:ss) for each prompt across different ratios and languages for the validation set.

Language	Ratio	Prompt 1	Prompt 2	Prompt 3
en	0.4	00:38:00	03:04:00	05:31:00
	0.6	00:28:00	02:40:00	04:44:00
	0.7	00:28:00	01:33:00	03:36:00
	0.8	00:09:00	00:27:00	00:47:00
es	0.4	03:27:00	09:01:00	13:41:00
	0.6	03:38:00	09:36:00	14:17:00
	0.7	02:41:00	07:58:00	12:38:00
	0.8	01:38:00	03:56:00	06:40:00
de	0.4	01:29:00	04:48:00	07:39:00
	0.6	01:53:00	05:35:00	08:42:00
	0.7	02:06:00	04:04:00	09:10:00
	0.8	01:47:00	05:29:00	09:12:00