Fine-tuned Bi-Encoder for Job-Skill Matching: Exploring Curriculum Learning, Hard Negative Mining, and Hybrid Retrieval

Working Notes Paper for the TalentCLEF lab at CLEF 2026

Lakshmi Priya Swaminatha Rao[1,†], Dravid Ranjan[1,*,†], Adithya S[1,†] and Vino Sasi Kumar Roxlin Rani Harrison[1,†]

[1]*Department of Computer Science and Engineering, Sri Sivasubramaniya Nadar College of Engineering, Chennai, India*

Abstract

This paper describes Team Olive's participation in TalentCLEF 2026 Task B, which involves matching job titles to relevant skills from a fixed catalogue. We developed three systems across successive iterations. We began with a retrieval-focused model trained through a three-stage curriculum that included hard negative mining combined with hybrid dense and sparse search. Finding that the hard negative training stage consistently degraded retrieval quality, we simplified to a two-stage curriculum with deeper retrieval as our second attempt. Our final submitted system stepped back further, using a single-stage fine-tuned multilingual model with straightforward dense retrieval, which achieved the strongest performance of all three. Our submitted system achieved an NDCG of 0.7050 and MAP of 0.2025 on the validation set using the official TalentCLEF evaluation script, and $\text{NDCG}_{\text{binary}}$ of 0.6899 and $\text{MAP}_{\text{binary}}$ of 0.1853 on the official test set, both above the TalentCLEF 2026 baseline. We share the lessons learned from each iteration, including why a more complex pipeline did not always lead to better results.

Keywords

job-skill matching, bi-encoder, curriculum learning, hard negative mining, hybrid retrieval, ESCO, TalentCLEF 2026

1. Introduction

1.1. Motivation

Automatically matching job titles to relevant occupational skills is a core challenge in Human Capital Management (HCM). With job postings attracting large numbers of applicants, automated systems that can efficiently identify and rank relevant skills for a given job title are of growing practical importance. Such systems underpin applications including skill-based candidate ranking, personalised up-skilling recommendations, and early detection of emerging skill gaps in the labour market.

The challenge is fundamentally one of semantic matching: a job title such as "software engineer" implies a broad but structured set of competencies, while each skill in a taxonomy like ESCO is described using its own vocabulary. Bridging this vocabulary gap reliably, at scale, and without hand-crafted rules requires learned representations that generalise across diverse occupational domains. Recent advances in dense retrieval with sentence transformers have made this more tractable, but adapting these models to the specific structure of occupational data remains an open and practically important research problem.

CLEF 2026 Working Notes, 21 – 24 September 2026, Jena, Germany

*Corresponding author.

†These authors contributed equally.

lakshmipriyas@ssn.edu.in (L. P. S. Rao); dravidranjan2410209@ssn.edu.in (D. Ranjan); adithya2410422@ssn.edu.in (A. S); vino2410181@ssn.edu.in (V. S. K. R. R. Harrison)

0000-0002-9923-4020 (L. P. S. Rao); 0009-0008-5856-6310 (D. Ranjan); 0009-0006-9283-1478 (A. S); 0009-0007-7427-9736 (V. S. K. R. R. Harrison)

1.2. Task Definition

TalentCLEF 2026 [1] is a shared task at CLEF 2026 focusing on skill and job title intelligence for Human Capital Management. Task B specifically addresses Job-Skill Matching with Skill Type Classification [2]. Given a job title as a query, systems must rank a corpus of ESCO (European Skills, Competences, Qualifications, and Occupations) [3] skills by their relevance to that role. The task is evaluated under two settings:

- **Binary relevance:** Whether a skill is associated with the job (1) or not (0).
- **Graded relevance:** The degree of relevance, distinguishing essential skills from optional skills, where essential skills carry a higher relevance weight.

The primary evaluation metrics are Normalised Discounted Cumulative Gain (NDCG) and Mean Average Precision (MAP) computed under each relevance setting.

1.3. Challenges

Several challenges make this task non-trivial. First, the vocabulary gap between concise job title descriptions and skill alias text means that lexical overlap alone is insufficient for accurate matching. A job title may use informal or domain-specific phrasing that does not surface in the ESCO skill names. Second, the ESCO skill corpus is large and contains many semantically similar skills, making it easy for retrieval models to confuse closely related but distinct competencies. Third, the distinction between essential and optional skills introduces a graded relevance signal that is not always recoverable from textual similarity alone, requiring models sensitive to subtle differences in job-skill association strength. Finally, job titles in the evaluation sets are selected by domain experts and may not appear directly in the training data, introducing a distribution shift that tests the generalisation capacity of any trained system.

1.4. Contribution

We describe three systems developed for TalentCLEF 2026 Task B. Starting from a complex multi-stage curriculum with hard negative mining and hybrid retrieval, we progressively simplified through two iterations, arriving at a single-stage fine-tuned multilingual bi-encoder as our best and submitted system. Our experiments document both what worked and what did not, with particular attention to a hard negative training strategy that consistently degraded retrieval quality on this dataset — a finding we believe is practically useful to the wider community working on dense retrieval over semantically dense occupational corpora. To support reproducibility, our code and the submitted run file are publicly available at https://github.com/amdravidranjan/talentclef2026-taskb.

2. Related Work

2.1. Encoder Models for Job-Skill Matching

Bi-encoder architectures based on pretrained sentence transformers have become the standard approach for large-scale semantic retrieval. Reimers and Gurevych [4] introduced the Sentence-BERT family of models, demonstrating that fine-tuning transformer encoders with a siamese network structure and contrastive objectives yields dense sentence embeddings well suited to retrieval tasks. These models encode queries and documents independently, enabling efficient large-scale search through pre-computed embeddings and cosine or dot-product similarity.

For multilingual settings, `paraphrase-multilingual-mpnet-base-v2` [4] extends this approach to over 50 languages by training on multilingual paraphrase pairs, making it suitable for the multilingual nature of ESCO, which covers occupational taxonomies across European languages. The BGE model family from BAAI [5] takes a retrieval-specific approach, training models with contrastive objectives on

large-scale retrieval data and recommending the use of task-specific query prefixes to sharpen retrieval performance at inference time.

Prior work on TalentCLEF 2025 [6] established that fine-tuning general-purpose encoders on ESCO job-skill pairs with Multiple Negatives Ranking Loss yields meaningful gains over zero-shot baselines. Ali [7] further demonstrated that augmenting job title inputs with LLM-generated descriptions, which explicitly enumerate relevant skills, can substantially improve matching quality by providing richer semantic context to the encoder. The Task A [8] also helped with different perspective thinking. These findings informed our choice of MNRL as the training objective and motivated our exploration of richer input representations.

2.2. Hard Negative Mining

Standard contrastive training with in-batch negatives treats all non-positive items in a batch as negatives, which are typically easy to distinguish from the anchor. Hard negative mining addresses this by selecting negatives that are highly ranked by the current model but are not true positives, providing more informative training gradients. Xiong et al. [9] formalised this as Approximate Nearest Neighbour Negative Contrastive Estimation (ANCE), demonstrating substantial improvements in dense retrieval for open-domain question answering through iterative hard negative mining. We applied this strategy in our first system and studied its effect on the semantically dense ESCO skill corpus.

2.3. Hybrid Retrieval and Score Fusion

Sparse retrieval methods such as BM25 capture exact lexical matches and are robust to rare or domain-specific terms, while dense retrieval methods capture semantic similarity beyond lexical overlap. Hybrid systems combining both have consistently outperformed either method alone across a range of retrieval benchmarks [10]. Reciprocal Rank Fusion (RRF) [11] is a simple, widely adopted, parameter-free fusion method that combines ranked lists by summing reciprocal ranks, without requiring score normalisation or learned combination weights. We used RRF in our first two systems to combine BGE dense scores with BM25 scores over the skill corpus.

3. Dataset

The TalentCLEF 2026 Task B dataset is derived from the ESCO taxonomy [3], a multilingual European framework for occupations, skills, competences, and qualifications. The training data consists of three components:

- **Job-skill relations** : Maps job IDs to skill IDs with a label of `essential` or `optional` for each pair.
- **Job titles** : Provides multiple names and aliases for each job ID, reflecting the variety of ways the same occupation may be described.
- **Skill names** : Provides multiple names and aliases for each skill ID.

The training set contains 114,699 job-skill pairs in total: 62,480 labelled `essential` and 52,219 labelled `optional`, spanning 3,011 unique job IDs and 13,939 unique skill IDs. This asymmetry between essential and optional pairs directly informed our curriculum training design in Systems 1 and 2.

The validation and test sets are structured as an information retrieval problem. Each consists of a queries file (job titles), a corpus of 9,052 ESCO skill entries with their aliases, and a qrels file providing relevance judgements. The validation set covers 304 query jobs. Crucially, job titles in both the validation and test splits were selected by domain experts and are not necessarily direct entries in the ESCO taxonomy, introducing a distribution shift relative to training. This means that a system cannot simply look up a job in ESCO to retrieve its associated skills; it must generalise from the training distribution to unseen job title phrasings.

All text inputs in our systems are constructed by joining all available aliases for a given job or skill ID, separated by spaces, and truncating to a maximum token length before encoding. Label counts were verified directly from the training data and are summarised in Table 1.

Table 1
TalentCLEF 2026 Task B dataset statistics.

Split / Statistic	Count
Training job-skill pairs (total)	114,699
of which essential	62,480
of which optional	52,219
Unique job IDs (training)	3,011
Unique skill IDs (training)	13,939
Validation query jobs	304
Corpus skills (val / test)	9,052

4. Methodology

We describe our three systems in the order they were developed. The first two systems were exploratory iterations that informed the design of the third. The third system is our final submitted run. All systems use the bi-encoder retrieval paradigm: query and corpus encoders independently produce dense vector representations, and relevance is measured by cosine or dot-product similarity between them. This architecture enables corpus embeddings to be pre-computed offline, allowing fast retrieval at inference without the quadratic complexity of cross-encoder approaches.

4.1. System 1: BGE with Three-Stage Curriculum and Hybrid Retrieval

Our first system was designed around the hypothesis that a structured curriculum — moving from broad coverage to essential-skill sharpening to hard negative discrimination — combined with hybrid retrieval, would yield strong retrieval performance.

4.1.1. Model

We use `BAAI/bge-base-en-v1.5` [5], a 109M parameter encoder model specifically designed and trained for retrieval tasks. At inference, a query prefix (*"Represent this sentence for searching relevant passages:"*) is prepended to all job title queries, as recommended by the BGE authors to activate retrieval-oriented representations. The model was run on a Tesla T4 GPU (15.6 GB VRAM) via Google Colab.

4.1.2. Three-Stage Curriculum Training

Stage 1 (Broad). The model is initialised from the pretrained BGE checkpoint and trained on all 114,699 job-skill pairs (both essential and optional) using Multiple Negatives Ranking Loss (MNRL) [12]. MNRL treats all other items in a batch as implicit negatives for each anchor-positive pair, which is well suited to datasets that consist entirely of positive pairs without explicit negatives. Training uses batch size 32, mixed-precision (fp16), up to 3 epochs with early stopping (patience 5 evaluation steps), monitored by MAP@100 on the validation set.

Stage 2 (Sharpening). Starting from the best Stage 1 checkpoint, the model is further fine-tuned on the 62,480 essential-only pairs at a reduced learning rate of 1×10^{-5}. The intent is to sharpen the embedding space around the most critical job-skill associations, biasing the model towards retrieving skills that are genuinely essential to a role rather than merely optional.

Stage 3 (Hard Negatives). Using the Stage 2 model, we mine hard negatives from the full training skill pool of 13,939 skills. For each of the 3,011 training jobs, all skills are ranked by cosine similarity to the job embedding. To avoid selecting near-positive skills as negatives, the top-20 retrieved skills are skipped, and the next 5 are taken as hard negatives, yielding 312,400 hard negative triplets. A random subsample of 50,000 triplets is used for one epoch of MNRL training at a very low learning rate of 2×10^{-6}, intended to refine the decision boundary without destabilising earlier training.

4.1.3. Hybrid Retrieval with RRF

At inference, dense retrieval using normalised BGE dot-product scores and sparse BM25 retrieval over tokenised lowercase skill text are run independently over the 9,052-skill corpus. The top-$2K$ candidates from each are merged using Reciprocal Rank Fusion (RRF) [11]. Given ranked lists from dense and sparse retrieval, the RRF score for each skill d is computed as:

$$\text{RRF}(d) = \sum_{r \in \{\text{dense},\, \text{bm25}\}} \frac{1}{K + \text{rank}_r(d)}$$

where $\text{rank}_r(d)$ is the rank of skill d in ranked list r and $K = 60$ is a smoothing constant that reduces the impact of very high-ranked documents. Skills are then re-ranked in descending order of their RRF score, and the top-100 results are returned as the final ranked list.

4.2. System 2: BGE with Two-Stage Curriculum and Deeper Retrieval

After evaluating System 1, we observed that Stage 3 hard negative training consistently degraded both MAP and MRR on the validation set, despite improving training loss. We attribute this to the semantic density of the ESCO skill corpus: many skills share closely related textual descriptions and cover overlapping occupational domains. As a result, the top-ranked non-positive skills for a given job are often semantically adjacent to the correct positives — making them poor hard negatives. Training the model to push these apart distorts the overall geometry of the embedding space and reduces recall. This observation was confirmed empirically: when Stage 3 training was applied, both MAP and MRR on the validation set declined relative to the Stage 2 checkpoint, leading us to treat the Stage 2 model as the best checkpoint for System 2.

System 2 therefore retains the BGE backbone and the two-stage curriculum (Stages 1 and 2) from System 1, with Stage 3 removed entirely. The remaining design changes listed below were not derived from a formal hyperparameter sweep or ablation study; they were set a priori based on general best practices in dense retrieval and motivated by the observed failure modes of System 1. Specifically, the larger batch size was motivated by the desire for more diverse in-batch negatives, the switch to `CachedMNRL` by GPU memory constraints at the larger batch size, and the deeper retrieval pool by the hypothesis that System 1 was missing relevant skills ranked beyond position 100. Additionally:

- The training batch size is doubled to 64, providing more diverse in-batch negatives per step.
- `CachedMultipleNegativesRankingLoss` with `mini_batch_size=32` is used instead of standard MNRL, effectively increasing the number of negatives considered without proportionally increasing GPU memory.
- The retrieval depth is extended from top-100 to top-500, increasing the recall ceiling for jobs with many associated skills.
- The validation evaluator is expanded to additionally track MAP@500, MRR@10, MRR@100, NDCG@100, and Precision/Recall at cutoffs 5, 10, and 100 for richer diagnostics.
- The monitored metric is switched to MAP@500 to better align with full-ranking evaluation behaviour.

4.3. System 3: Multilingual Bi-Encoder with Single-Stage MNRL (Submitted)

Having observed that both BGE-based systems underperformed expectations, we reconsidered the backbone model choice and training complexity. Our third system simplifies substantially: a single-stage fine-tuned multilingual bi-encoder trained on all job-skill pairs, with straightforward dense retrieval at inference.

4.3.1. Model

We use `paraphrase-multilingual-mpnet-base-v2` [4] as the backbone encoder. This model produces 768-dimensional sentence embeddings pre-trained on over 50 languages using multilingual paraphrase pairs, making it naturally suited to the multilingual nature of ESCO. Unlike BGE, this model was not specifically designed for retrieval but for general-purpose semantic similarity. As our results show, this turns out to be a better fit for this task.

4.3.2. Training Data Preparation

All 114,699 job-skill pairs (both essential and optional) are used. For each pair, the job text is formed by joining all aliases for that job ID, and the skill text by joining all aliases for that skill ID. Both are truncated to a maximum of 80 tokens. Pairs where either the job or skill text is empty after alias resolution are discarded. Each valid pair forms an `(anchor, positive)` example for MNRL training. The choice to concatenate all available aliases rather than using a single canonical name was motivated by two considerations. First, ESCO entries frequently have multiple aliases that describe the same concept using different vocabulary — for example, the skill *"organise rehearsals"* also appears as *"organize rehearsals"*, *"plan rehearsals"*, and *"schedule rehearsals"*. Concatenating all aliases exposes the encoder to this lexical variation during training, encouraging it to produce representations that are invariant to surface-level phrasing differences. Second, since validation and test job titles are selected by domain experts and may not match any single ESCO canonical name exactly, a model trained on single canonical names risks failing to generalise to unseen phrasings, whereas one trained on all aliases is exposed to a broader distribution of how a role can be described. We acknowledge that this choice was not empirically ablated against using single canonical names due to time constraints. It is possible that concatenating many aliases introduces noise if aliases vary substantially in specificity or domain framing, and that a single carefully chosen canonical name per entry would yield a cleaner training signal. We leave a systematic comparison of alias concatenation strategies to future work.

4.3.3. Training

Fine-tuning uses MNRL with a batch size of 32 and a warmup ratio of 0.1. Training runs for up to 8 epochs, with early stopping (patience of 90 evaluation steps) monitored by MAP@100 on the validation set using an `InformationRetrievalEvaluator`. The number of evaluation steps per epoch is computed dynamically as $\max(1, \lfloor |\text{train set}|/(\text{batch size} \times 30) \rfloor)$. The checkpoint achieving the highest MAP@100 on the validation set is retained. The training was run on Google Colab with a total training runtime of 2,627 seconds, processing 349.4 samples per second. All hyperparameters were set a priori rather than through a formal grid search or ablation study. The batch size of 32 and warmup ratio of 0.1 follow common defaults for fine-tuning sentence transformer models with MNRL as recommended in the Sentence Transformers documentation [4]. The maximum token length of 80 was chosen to cover the typical length of concatenated job or skill alias strings observed in the training data, while remaining within the model's context window. The ceiling of 8 epochs and early stopping patience of 90 evaluation steps were set conservatively to allow sufficient training time while preventing overfitting, with the validation MAP@100 acting as the selection criterion for the best checkpoint. To ensure reproducibility, a fixed random seed of 42 was used for all data shuffling operations in Systems 1 and 2; System 3 relies on the deterministic behaviour of the `SentenceTransformerTrainer` under the same hardware and library versions.

4.3.4. Inference

All 9,052 corpus skill entries are encoded in batches of 128 using the fine-tuned model. For each query job title, its embedding is computed and cosine similarity is evaluated against all corpus skill embeddings. Skills are ranked in descending order of similarity, and a TREC-format run file is produced. The resulting run file is submitted to the official evaluation platform.

Table 2 summarises the configuration of all three systems side by side.

Table 2
Configuration comparison across the three systems.

Configuration	System 1	System 2	System 3
Backbone	bge-base-en	bge-base-en	mpnet-multilingual
Max token length	120	120	80
Loss function	MNRL	CachedMNRL	MNRL
Training stages	3	2	1
Batch size	32	64	32
Warmup ratio	0.1	0.1	0.1
Hard negatives	Yes (Stage 3)	No	No
Retrieval method	Dense + BM25	Dense + BM25	Dense only
Retrieval depth	Top-100	Top-500	Full corpus
GPU used	Tesla T4	Tesla T4	Tesla T4

5. Results and Discussion

5.1. Evaluation Metrics

We report results using two evaluation approaches. For validation-set comparison across systems, we use `pytrec_eval` [13] directly against the provided qrels to compute NDCG@10. We also report results using the official TalentCLEF 2026 evaluation script [2], which applies a relevance mapping (labels 1 or 2 are treated as relevant for binary metrics; label 2 as core skill and label 1 as contextual skill for graded metrics) and reports NDCG, MAP, MRR, and Precision at multiple cutoffs. For the final submitted system, we also report official test-set scores across four metrics:

- **$\text{NDCG}_{\text{binary}}$:** NDCG under binary relevance.
- **$\text{MAP}_{\text{binary}}$:** MAP under binary relevance.
- **$\text{NDCG}_{\text{graded}}$:** NDCG under graded relevance (essential/core skills weighted higher).
- **$\text{MAP}_{\text{graded}}$:** MAP under graded relevance.

5.2. Validation Results

Table 3 reports NDCG@10 computed with `pytrec_eval` on the validation set for all three systems. System 3 achieves the highest score of 0.3135, confirming it as the best system for submission.

Table 3
Validation set NDCG@10 (pytrec_eval) for each system.

System	NDCG@10
System 1 (BGE, 3-stage + hard negatives, hybrid top-100)	0.3083
System 2 (BGE, 2-stage, hybrid top-500)	0.2851
System 3 (mpnet, single-stage, dense) — submitted	**0.3135**

Table 4
Validation set results using the official TalentCLEF 2026 evaluation script.

System	NDCG	MAP	MRR	P@5
System 1 (BGE, 3-stage + hard neg., hybrid top-100)	0.1861	0.0594	0.6433	0.3974
System 2 (BGE, 2-stage, hybrid top-500)	0.3633	0.1108	0.6285	0.3895
System 3 (mpnet, single-stage, dense)	**0.7050**	**0.2025**	**0.6461**	**0.4224**

Table 4 reports validation results using the official TalentCLEF 2026 evaluation script. Note that the NDCG values here differ from Table 3 because the official script applies a different relevance mapping and evaluates over all ranks rather than just the top 10.

System 3 achieves substantially higher NDCG (0.7050) and MAP (0.2025) on the official script compared to Systems 1 and 2, confirming its superiority across all metrics and motivating its selection as the submission. The very low NDCG and MAP for System 1 (0.1861 and 0.0594) under the official script, despite its reasonable pytrec NDCG@10 of 0.3083, is explained by the retrieval depth limit of top-100: the official script evaluates over all ranks, so a system returning only 100 results is heavily penalised for the many relevant skills it fails to retrieve beyond position 100.

5.3. Official Test Set Results

Table 5 reports the official evaluation results for our submitted system (System 3, Submission ID: 693022) against the TalentCLEF 2026 baseline, as returned by the official evaluation platform.

Table 5
Official Task B test set results for Team Olive (Submission ID: 693022), as reported by the TalentCLEF 2026 evaluation platform.

System	NDCG_{bin}	MAP_{bin}	NDCG_{grd}	MAP_{grd}
Olive (Sub. ID: 693022)	**0.6899**	**0.1853**	**0.6457**	**0.1853**
TalentCLEF Baseline	0.6634	0.1471	0.6204	0.1471

Our submitted system surpasses the official baseline across all four metrics: $\text{NDCG}_{\text{binary}}$ improves by +0.0265, $\text{NDCG}_{\text{graded}}$ by +0.0253, and both MAP scores by +0.0382. The identical MAP values under binary and graded settings (0.1853 for Olive, 0.1471 for the baseline) are expected: MAP is computed from the rank positions of relevant documents and does not incorporate graded relevance weights, so the binary and graded MAP scores coincide when the same set of documents is judged relevant under both settings.

5.4. Discussion

5.4.1. Why the Simplest System Won

The most important finding from our experiments is that the simplest approach — a single-stage fine-tuned multilingual bi-encoder with straightforward dense retrieval — outperformed both more elaborate systems across all evaluation frameworks. We offer two explanations. First, the multilingual `mpnet-base-v2` backbone was pre-trained on a large and diverse multilingual paraphrase corpus, giving it broad coverage of the semantic relationships present in occupational language, including cross-lingual associations within ESCO. Second, single-stage MNRL training on the full set of job-skill pairs, covering both essential and optional associations, exposes the model to the widest possible range of job-skill relationships in each batch, providing rich and diverse in-batch negatives without any of the distortion introduced by hard negatives or curriculum sharpening.

5.4.2. The Cost of Hard Negative Training

System 1 demonstrated that hard negative mining, while theoretically beneficial for dense retrieval in general domains, can be actively harmful in semantically dense corpora like ESCO. To make this concrete, consider the following real example from our hard negative cache. The job *"technical director"* (also aliased as *"head of technical department"*, *"technical supervisor"*) has annotated positive skills including *"organise rehearsals"*, *"theatre techniques"*, and *"promote health and safety"*. When the Stage 2 BGE model ranks all 13,939 training skills for this job, the top non-positive results selected as hard negatives include *"manage creative department"*, *"coordinate artistic production"*, *"supervise sound production"*, and *"collaborate with a technical staff in artistic productions"*. These are not genuinely irrelevant: a technical director in a performing arts context plausibly requires all of them. They are absent from the ESCO annotations for this job ID not because they are irrelevant, but because ESCO annotations are incomplete — no job in the taxonomy is exhaustively labelled with every skill it might require. Training the model to push these skills away from the *"technical director"* embedding introduces conflicting gradients that damage the coherence of the entire embedding space, reducing recall for semantically adjacent skills across all jobs.

One might ask whether simply increasing the skip depth — skipping more than the top-20 retrieved skills before selecting hard negatives — would avoid this problem. In a sparse retrieval corpus with a clear lexical boundary between relevant and irrelevant documents, this would help. In a dense occupational taxonomy like ESCO, however, the problem is structural rather than positional: skills are organised into tightly clustered competency groups, and any job title will have dozens of semantically adjacent unannotated skills at varying ranks throughout the ranked list. As shown in the example above, all five of the top-ranked non-positive skills for *"technical director"* are arguably relevant. Increasing the skip depth merely pushes the selection window further down the ranking, where skills are less similar on average but still potentially relevant and unannotated. The only reliable way to avoid selecting near-positives as hard negatives would be to have exhaustive relevance annotations for every job-skill pair, which the ESCO training labels do not provide. The low official-script NDCG of 0.1861 for System 1 — versus 0.7050 for System 3 — quantifies the severity of this effect. System 2 confirmed the finding: removing Stage 3 improved the official-script NDCG from 0.1861 to 0.3633, but the two-stage BGE curriculum still fell well short of System 3.

5.4.3. Retrieval Depth and Evaluation Sensitivity

An important methodological observation concerns the interaction between retrieval depth and evaluation score. For System 1, pytrec_eval NDCG@10 is 0.3083, yet the official-script NDCG is only 0.1861. For System 2, pytrec NDCG@10 is 0.2851 while the official-script NDCG is 0.3633. These discrepancies arise because NDCG@10 measures quality only at the top 10 positions, whereas the official script evaluates over all retrieved results. Systems 1 and 2 cap their output at top-100 and top-500 respectively, so any relevant skill ranked beyond those cutoffs is treated as not retrieved — a severe penalty under full-ranking evaluation.

The top-K limits in Systems 1 and 2 were chosen for practical reasons. In a two-stage retrieve-and-rerank pipeline (the intended use case for a hybrid dense+BM25 system), returning a smaller candidate set is standard practice: top-100 or top-500 candidates would be passed to a cross-encoder for reranking, making full-corpus ranking unnecessary and computationally wasteful. However, since we did not implement a reranking stage in the final pipeline, these truncation limits became a confounding factor in the evaluation.

We acknowledge this as a limitation of our comparison: the official-script score gap between Systems 1 and 2 versus System 3 is driven in part by retrieval depth rather than purely by model quality. A fair comparison would re-evaluate Systems 1 and 2 by extending their retrieval to the full 9,052-skill corpus. Due to time and resource constraints during the shared task, this re-evaluation was not performed. Based on the pytrec NDCG@10 scores — which are not confounded by retrieval depth — System 3 (0.3135) still outperforms Systems 1 (0.3083) and 2 (0.2851), suggesting the model quality conclusion

holds independently of retrieval depth. Addressing this with full-corpus re-evaluation is left as future work.

5.4.4. NDCG vs. MAP

Across all systems, NDCG scores are consistently higher than MAP scores. This reflects a fundamental difference in what these metrics measure: NDCG rewards placing any relevant skill high in the ranking and applies a logarithmic discount to lower positions, so a system can achieve a strong NDCG by reliably surfacing a subset of relevant skills near the top. MAP, by contrast, computes the average precision at every rank position where a relevant skill appears, penalising the system each time a relevant skill is ranked below an irrelevant one. Since each job title is associated with many relevant skills spread throughout the 9,052-skill corpus, achieving consistently high precision at all recall levels is inherently demanding, and the gap between NDCG and MAP is expected in this setting.

5.4.5. Directions for Improvement

There remains a gap between our submitted system and the top-performing submissions on the leaderboard. Based on our experimental findings and the related literature, we identify the following as the most promising directions for future work. Augmenting job title queries with LLM-generated descriptions that enumerate key responsibilities and typical skills, as demonstrated in [7], would enrich the query representation and likely improve recall for skills that are not lexically similar to the bare job title. Experimenting with larger backbone models or models specifically domain-adapted to occupational language may also yield stronger baseline embeddings. A two-stage retrieve-and-rerank pipeline, where a bi-encoder retrieves candidates and a cross-encoder reranks them, could improve precision at the top of the ranking. Finally, incorporating an explicit skill type classifier to predict whether a skill is essential or optional for a given job could directly improve graded evaluation performance.

6. Conclusion

We presented Team Olive's system for TalentCLEF 2026 Task B. Through three successive iterations, starting from a complex multi-stage curriculum with hard negative mining and hybrid retrieval, and progressively simplifying our design, we found that a single-stage fine-tuned multilingual bi-encoder with full-corpus dense retrieval performed best across all evaluation frameworks. Our submitted system achieved an NDCG of 0.7050 and MAP of 0.2025 on the validation set, and $\text{NDCG}_{\text{binary}} = 0.6899$, $\text{MAP}_{\text{binary}} = 0.1853$, $\text{NDCG}_{\text{graded}} = 0.6457$, and $\text{MAP}_{\text{graded}} = 0.1853$ on the official test set, all exceeding the TalentCLEF 2026 baseline.

The central lesson from our experiments is twofold. First, hard negative training can actively hurt performance in semantically dense skill corpora where the boundary between true positives and the highest-ranked non-positives is blurry. Second, limiting retrieval depth at inference can significantly suppress NDCG and MAP under full-ranking evaluation, even when top-10 precision looks reasonable. Both findings are practically important for teams building dense retrieval systems over domain-specific taxonomies like ESCO. Future work will explore LLM-generated job descriptions for richer query representations, larger retrieval-optimised backbone models, and cross-encoder reranking to improve precision at the top of the ranked list.

Declaration on Generative AI

During the preparation of this work, the author(s) used Claude (Anthropic) in order to: Paraphrase and reword, Grammar and spelling check. After using this tool, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

Acknowledgments

The authors thank the TalentCLEF 2026 organisers for providing the dataset, evaluation infrastructure, and the shared task platform.

References

[1] L. Gasco, H. Fabregat, L. García-Sardiña, P. Estrella, W. Veys, C. P. Carrino, M. De Lange, D. Deniz Cerpa, A. Rodrigo, J.-J. Decorte, R. Zbib, Overview of the talentclef 2026: Skill and job title intelligence for human capital management, in: International Conference of the Cross-Language Evaluation Forum for European Languages, 2026.

[2] W. Veys, L. Gasco, M. De Lange, J.-J. Decorte, Overview of talentclef 2026: Task B — job-skill matching with skill type classification, in: CLEF (Working Notes), 2026.

[3] M. le Vrang, A. Papantoniou, E. Pauwels, P. Fannes, D. Vandensteen, J. De Smedt, ESCO: Boosting job matching in europe with semantic interoperability, Computer 47 (2014) 57–64.

[4] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2019, pp. 3982–3992.

[5] S. Xiao, Z. Liu, P. Zhang, N. Muennighoff, C-Pack: Packaged resources to advance general chinese embedding, arXiv preprint arXiv:2309.07597 (2023).

[6] L. Gasco, H. Fabregat, L. García-Sardiña, P. Estrella, D. Deniz, A. Rodrigo, R. Zbib, Overview of the talentclef 2025: Skill and job title intelligence for human capital management, in: International Conference of the Cross-Language Evaluation Forum for European Languages, 2025, pp. 464–485.

[7] M. Ali, Enhancing job-skill matching with LLM-driven data augmentation and fine-tuned bi-encoders, in: CLEF 2025 Working Notes, 2025.

[8] H. Fabregat, L. García-Sardiña, P. Estrella, L. Gasco, C. P. Carrino, D. Deniz Cerpa, A. Rodrigo, R. Zbib, Overview of talentclef 2026: Task a — contextualized job-person matching, in: CLEF (Working Notes), 2026.

[9] L. Xiong, C. Xiong, Y. Li, K.-F. Tang, J. Liu, P. Bennett, J. Ahmed, A. Overwijk, Approximate nearest neighbor negative contrastive estimation for dense text retrieval, in: International Conference on Learning Representations, 2021.

[10] J. Lin, X. Ma, A few brief notes on deepimpact, coil, and a conceptual framework for information retrieval techniques, arXiv preprint arXiv:2106.14807 (2021).

[11] G. V. Cormack, C. L. A. Clarke, S. Buettcher, Reciprocal rank fusion outperforms condorcet and individual rank learning methods, in: Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2009, pp. 758–759.

[12] M. Henderson, R. Al-Rfou, B. Strope, Y.-H. Sung, L. Lukács, R. Guo, S. Kumar, B. Miklos, R. Kurzweil, Efficient natural language response suggestion for smart reply, in: arXiv preprint arXiv:1705.00652, 2017.

[13] C. Van Gysel, M. de Rijke, Pytrec_eval: An extremely fast python interface to trec_eval, in: Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2018, pp. 873–876.