Fine-Tuned Sentence Transformer for Multilingual Job Title Matching

Chinmay Satish Bhangale[1,†], Prajwal Anil Gabhane[1,†] and Anand Kumar Madasamy[1]

[1]Department of Information Technology, National Institute of Technology Karnataka Surathkal, Mangalore 575025, India

Abstract

Matching job titles is critical task in various fields, such as resume evaluation, and job recommendation platforms. Many companies use different job title for similar roles which creates ambiguity. This research tackles the issue by developing a machine learning-based strategy that makes use of a Sentence Transformer model paraphrase-multilingual-mpnet-base v2, finetuned for the job title matching task. The training dataset consists of job titles paired with their corresponding similar job titles across three languages—English, Spanish, and German—while the validation data includes a query file and a corpus file, each containing job titles in the same languages. To ensure data consistency, preprocessing steps are applied, like handling missing values, normalizing text and removing special characters. Cached Multiple Negatives Ranking Loss is used to improve retrieval accuracy, which helps the model to distinguish between similar and dissimilar job titles. After training, the embeddings are generated for each job title in query and corpus file. Cosine similarity is used to compute similarity scores between the query and corpus job title embeddings. Finally, for each query job title, corpus job titles are ranked based on their similarity scores. The model's performance evaluated using standard retrieval metrics, including Mean Average Precision (MAP), Mean Reciprocal Rank (MRR), and Precision@K. The fine-tuned model achieved an average MAP score of 0.49 across English, Spanish, and German languages on the validation data, and 0.45 on the test data.

Keywords

Job title, Sentence Transformer, Cached Multiple Negatives Ranking Loss

1. Introduction

With the tight job market today, organizations are increasingly competing among themselves to find and retain the best qualified candidates [1]. The idea of Information and Communication Technologies (ICTs) in Human Resources (HR) processes has lessened the burden of time-related hiring, especially with the introduction of artificial intelligence (AI) [2]. The evolution of the job application space has been marked with many changes, a key one being the development of automated job recommendations. For job seekers, this also means getting better matched job recommendations from the millions of jobs posted online at any given moment, and applying for those roles according to their skills and ambitions.

Job-matching is at the center of human resource management. Enterprise post-management focuses on defining requirements of whether the talent can meet the job (position) qualifications and whether they match the talent that the enterprise needs for growth in a systematic and scientific way [3, 4].

Accurately matching job titles is still difficult because of ambiguity in job naming conventions, linguistic variances, and domain-specific terminologies, even with the integration of ICTs and improvements in job recommendation systems. Conventional techniques like keyword matching and TF-IDF frequently fall short in capturing the semantic similarity between related but distinct job titles. This causes difficulties for job seekers searching for specific roles and also reduces the effectiveness of recruitment systems.

There is a need for systems that understand the semantic meaning of job titles and return the most relevant matches, especially in cross-domain and multilingual environments. Thus, in order to improve

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

[†]These authors contributed equally.

chinmaybhangale.242it006@nitk.edu.in (C. S. Bhangale); gabhaneprajwal.242it011@nitk.edu.in (P. A. Gabhane); m_anandkumar@nitk.edu.in (A. K. Madasamy)

0009-0008-5713-1673 (C. S. Bhangale); 0009-0006-5202-0996 (P. A. Gabhane); 0000-0003-0310-4510 (A. K. Madasamy)

precision and speed of job title retrieval and matching procedures, a more intelligent and scalable approach is required.

Transformer-based models have changed the landscape of NLP with their performance boost over the previous state-of-the-art approaches in information retrieval tasks. Transformers, like Bidirectional Encoder Representations from Transformers (BERT) [5] and its fellow descendants, learn contextual meaning and capture semantic similarities much better than bag-of-words or Term Frequency-Inverse Document Frequency (TF-IDF) based methods. Research has shown that models like Sentence-BERT improve retrieval tasks by generating high quality sentence embeddings that can easily be compared using cosine similarity. The latest developments in this domain such as the multilingual BERT (m-BERT) [6] and Cross Lingual Language Model with RoBERTa (XLM-R) [7], they have made cross-lingual retrieval easier and most suited to be employed for performing job title matching.

This work leverages transformer-based models for Job Title Matching, for retrieving and ranking similar job titles, as described in TalentCLEF 2025 overview paper [8]. The training dataset contains job titles and their corresponding similar job titles, whereas test dataset consists of query file and corpus file, each consists of job titles. The Siamese Sentence Transformer model is then fine-tuned to learn meaningful representation of job titles. The embeddings are generated for each job title in query and corpus file. Cosine similarity is then used to compute similarity scores between the query and corpus job title embeddings. Finally, for each query job title, corpus job titles are ranked based on their similarity scores.

The rest of this report is structured as follows. Section II presents a review of literature pertaining to the use of NLP techniques in the field of human resource management. Section III covers a detailed methodology of the work that includes dataset, data preprocessing and transformer model information. Later, similarity score calculation is described. Section IV contains the experimental results obtained after fine-tuning the model. Section V concludes the report by summarizing the entire work, followed by suggestions for future work.

2. Literature Review

Work related to the usage of deep learning techniques, especially transformers, in human resource management are discussed below.

Modified BERT architecture using a Siamese or triplet network [9] has been implemented to generate sentence embeddings efficiently. The model does not simply concatenate each pair of sentences. Instead, it processes each independently and applies a pooling mechanism (mean, or max pooling) to obtain fixed-sized embeddings. An order of semantic similarity is optimized with the help of contrastive, triplet, and regression losses. Unlike other tasks, Semantic Textual Similarity (STS) and Natural Language Inference (NLI) greatly accelerate similarity evaluations. However, there are limitations, including the tradeoff between efficiency and accuracy, since cross-encoders are better for some use cases. Regarding limited generalizability, as Sentence-BERT suggests, it must be fine-tuned on a narrower dataset. Thereby, it is not as generalizable to different NLP tasks. Its performance drops for deeply interactive problems (e.g., question answering) and requires large amounts of high quality labeled data, severely restricting its use in low-resource settings. These disadvantages aside, Sentence-BERT works well for semantic similarity and information retrieval tasks.

Unsupervised learning and contrastive learning have been combined to propose a two-stage multilingual job recommendation system [10] to improve semantic similarity across 11 languages. An encoder (such as XLM-RoBERTa) is pretrained on multilingual data through Doc2Vec to align job titles with skill-embeddings. Then, contrastive fine-tuning uses positive and negative job pairs from European Skills, Competences, Qualifications and Occupations (ESCO) taxonomy dataset to optimize the embeddings, thereby improving cross-lingual alignment. The approach attains 4.3% mAP above monolingual BERT on English and excellent cross-lingual accuracy (e.g., Chinese-to-English mAP rises from 0.04 to 0.72). Nevertheless, it comes with certain drawbacks such as reliance on the noisy skill data, exclusion of Asian languages in the ESCO, absence of contextual job descriptions, and very high

computational cost of the transformer models.

VacancySBERT [11], a Siamese network leveraging BERT has been used to normalize job titles by matching them with skills from descriptions, using distant supervision on 50M title-description pairs and evaluated on 33K manually annotated triplets. It uses a custom [SKILL] token to sum skill embeddings and Multiple Negative Ranking Loss to align titles with their associated skills, resulting in 21.5% improved recall by including skills. Despite being effective, it has several limitations, including that the test data is biased towards specific niches ("Professional"), it works exclusively on English-language applications, comes with a proprietary skill-extraction algorithm, it uses BERT-base instead of larger and better models to optimize processing, and that poor negative sampling would leave out existing some relevant candidates.

3. Methodology

Figure 1 shows the overview of the multilingual job title matching system. On a broad level, it consists of 4 components - input, data preprocessing, model architecture and output. These are discussed in detail in subsequent subsections.

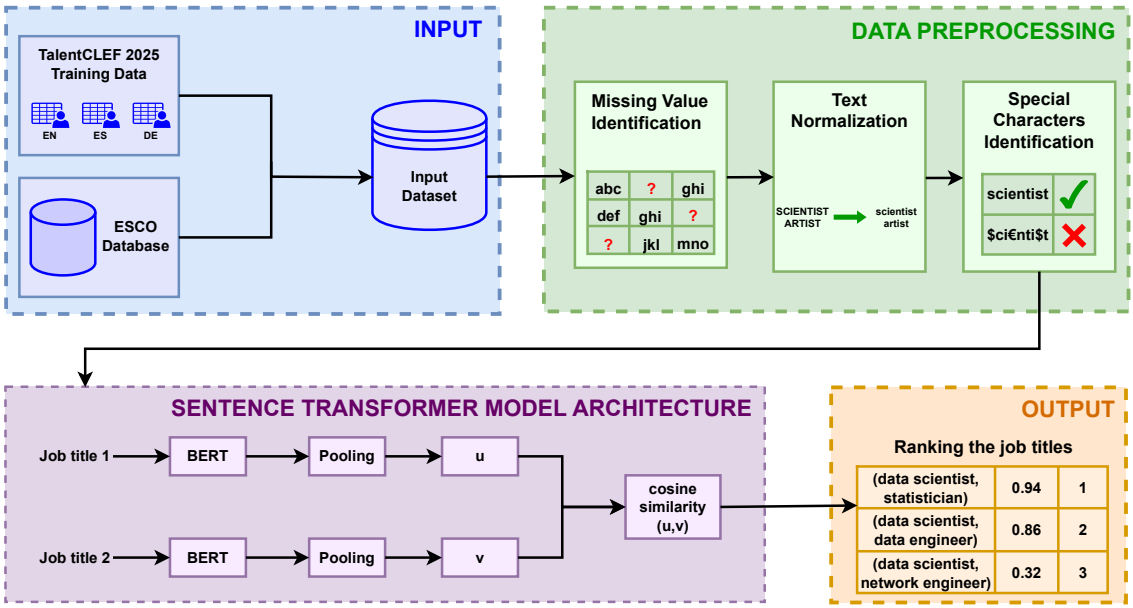

Figure 1: Overview of the Multilingual Job Title Matching System development

3.1. Dataset

We have used the training data and validation data from the TalentCLEF 2025 Task A [12] for this work. The data is provided in three languages: English, Spanish, and German. For each language the training data file, is a tab-delimited CSV file with four columns: family_id, id, jobtitle_1, jobtitle_2. The family_id column represents the ISCO family ID, which groups job titles under standardized occupational categories. The id column denotes the ESCO identifier, which traces the origin of each job title pair. A semantically similar or related occupation to jobtitle_1 is represented by jobtitle_2, while pairs of related job titles are stored in the jobtitle_1 and jobtitle_2 columns. The training data with English language contains 28880 rows, Spanish language contains 20724 rows, and German language contains 23023 rows.

The validation data for each language is structured into three separate files: queries, corpus_elements, and q_rels. The queries file contains 2 columns: q_id, and jobtitle, where q_id is a unique identifier

for each query and jobtitle specifies the job title used as the query. The corpus elements file contains 2 columns: c_id, and jobtitle, where c_id is a unique identifier for each corpus element and jobtitle represents the job title in the corpus. Finally, the q_rels file defines the relationship between the queries and the corpus_elements files. It contains 4 columns: q_id, iter, c_id, relevance, where q_id is query identifier, iter is a reserved field which is always set to 0, c_id is corresponding corpus element identifier, and relevance is a binary score which indicates the relevance of the corpus element to the query, where 1 means relevant and 0 means non-relevant. The q_rels file serves as the ground truth values for evaluating the model's performance, providing the expected relevance labels for each query-corpus jobtitle pair. For English language queries file contains 105 rows, corpus_elements file contains 2619 rows, and q_rels file contains 2419 rows. For Spanish language queries file contains 185 rows, corpus_elements file contains 4661 rows, and q_rels file contains 7578 rows. For German language queries file contains 203 rows, corpus_elements file contains 4729 rows, and q_rels file contains 8416 rows.

Since the training data contains ESCO URLs in the family_id column for each language, we have used an ESCO dataset (version 1.2.0) [13] as additional dataset, to extract more information about each job title, which will help to train the model on more data. The Occupations.csv file from ESCO dataset for each language that is English, Spanish, and German are used as new training dataset. For each language Occupations.csv contains 3039 rows and 14 columns. We have used two columns named "preferredLabel" (contains unique job titles) and "altLabels" (contains similar job titles) from each language Occupations.csv file for training the model.

3.2. Data Preprocessing

First, the training dataset (from ESCO) is checked for missing values for each language, resulting in missing similar job titles for 28 unique job titles in English, 143 in Spanish and 28 in German. Hence, all these missing rows are deleted from the training dataset for each language. After that, the unique job titles are extracted. There are a total of 3011 unique job titles in English, 2896 in Spanish and 3011 in German.

For each unique job title, a set of similar job titles is created, which also includes the primary job title in that set. This ensured that the model could learn the relationship between the primary job title and its closely related titles.

After generating set of similar job title, each job title in the set is converted into lowercase text. Next, leading and trailing whitespace are removed. Along with that presence of punctuation and special characters (like !,@,*, etc.) are checked, if found that are also removed.

For each language, all possible job title pairs are generated within each set of similar job titles. For example, in English dataset, if a set contained Software Engineer, Developer, Programmer, the possible pairs are created as: ('Software Engineer', 'Developer'), ('Software Engineer', 'Programmer'), ('Developer', 'Programmer'). Once all pairs are generated, they are randomly shuffled. Shuffling is performed to eliminate any order bias, ensuring that the model should be trained on a diverse and randomized set of pairs. Hence, 221889 unique job title pairs are generated in English language, 45647 in Spanish language, and 67818 in German language. After this, all unique job title pairs from each language are combined and randomly shuffled twice, resulting in total 335354 job title pairs.

Finally, 335354 job title pairs are formatted into the InputExample, which is usable by the Sentence-Transformer model. For each pair, We have created an InputExample object with a list of two texts (job titles). The format is essential and is used in the SentenceTransformer framework to generate embeddings and train the model for similarity learning. Figure 2 shows the creation of multilingual training dataset.

3.3. Sentence Transformer Model

This particular model is based on BERT (Bidirectional Encoder Representations from Transformers), which was created specifically to provide dense representations of words or small pieces text. BERT is mainly for getting token-level and contextual word representations, but Sentence Transformers

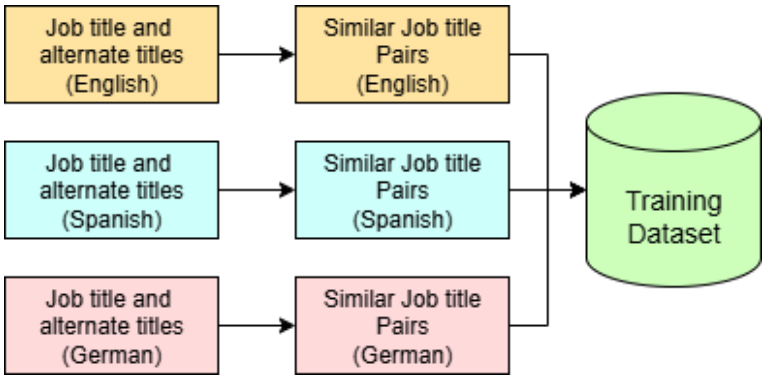

Figure 2: Creation of Training Dataset

provides meaningful sentence embeddings. These can be used downstream in applications including semantic similarity tasks and clustering or information retrieval. Reimers and Gurevych (2019) proposed an approach targeting DIF between BERT and other methodology such as BERT for the scoring of near identical sentence pair. BERT requires pairwise input comparisons, making it computationally expensive. Sentence Transformer can encode each sentence independently, allowing for efficient similarity calculations using cosine similarity or Euclidean distance.

The architecture of Sentence Transformer is based on pretrained BERT, but it includes an additional pooling layer to derive fixed-size embeddings from token level representations. In standard transformer models, the output consists of contextualized word embeddings for each token in the input sequence. A pooling operation is applied for sentence level tasks to aggregate information into a single vector. The most commonly used pooling strategies include mean-pooling (averaging all token embeddings), [CLS] token representation, and max-pooling. Among these, mean-pooling is frequently used as it effectively captures global sentence semantics.

Compared to BERT, Sentence Transformer is optimized for similarity-based tasks through the use of contrastive learning objectives. Instead of fine-tuning BERT using next sentence prediction (NSP) or masked language modeling (MLM), Sentence Transformer is trained using ranking losses, such as Multiple Negatives Ranking Loss and Multiple Negatives Symmetric Ranking Loss, which improve the model's ability to differentiate between similar and dissimilar text pairs.

The paraphrase-multilingual-mpnet-base-v2 model is a pretrained Sentence Transformer based on Microsoft's MPNet (Masked and Permuted Pre-training) architecture, which is fine-tuned for multilingual sentence similarity tasks. It generates 768-dimensional embeddings and it is trained on paraphrase pairs across more than 50 languages, which makes it exceptionally powerful for multilingual semantic understanding. This model captures very deep semantic relationships between languages, allowing for precise comparison of sentence meaning despite language disparity. It is particularly beneficial for multilingual use cases like multilingual information retrieval and semantic search. The paraphrase multilingual-mpnet-basev2 is selected for its best performance for multilingual sentence embeddings, a key requirement whenever job titles or job descriptions contain different languages yet similar meaning.

For the loss function, Cached Multiple Negatives Ranking Loss (Cached MNRL) is employed. This is a sophisticated version of the basic Multiple Negatives Ranking Loss (MNRL), which is intended to enhance training efficiency and representation quality in Sentence Transformer models for text similarity tasks. Like MNRL, it considers all other positive pairs within a batch as implicit negatives, avoiding the need for explicitly labeled negative samples. But Cached MNRL adds a cache of past batch embeddings to the memory, enabling the model to match current positive pairs with a more extensive and varied set of negatives. This greatly boosts the number of hard negatives, and this facilitates the model to better detect fine-grained semantic differences. Cached MNRL is particularly useful in large-scale or domain-specific applications—like job title matching—where it's important to detect fine-grained semantic differences.

3.4. Hyperparameter Setting

In the training phase, the Sentence Transformer model is trained using the below set of hyperparameters to optimize performance for the job title matching task.

The training was performed using the paraphrase-multilingual-mpnet-base-v2 model with the Cached Multiple Negatives Ranking Loss. The model was fine-tuned on a GPU-accelerated environment in order to improve training speed. Important hyperparameters were properly chosen to maximize performance. The number of training epochs was 1, and the batch size was 128 per device to utilize GPU memory efficiently without sacrificing negative diversity. A learning rate of 2e-5 was utilized as per standard fine-tuning practices for transformer-based models. A warmup ratio of 0.1 was also used to slowly increase the learning rate, minimizing the chances of instability at the beginning of training. Mixed precision training was also activated to save memory and accelerate computations. Training occurred in epoch strategy mode for saving model checkpoints and logging. In addition, we employed a no-duplicates batch sampling strategy to ensure that each batch contains unique samples, which is beneficial for contrastive learning methods that depend on in-batch negatives. Figure 3 illustrates the fine-tuning of the paraphrase-multilingual-mpnet-base-v2 Sentence Transformer model using the previously created multilingual training dataset.

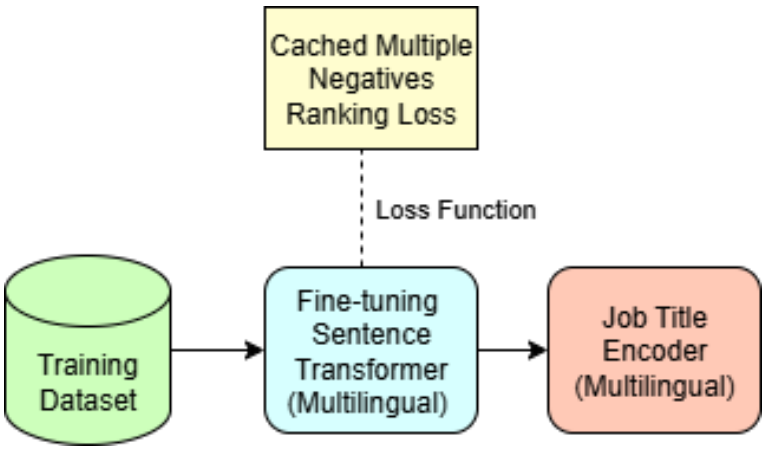

Figure 3: Model Training

3.5. Similarity Score Computation

Once the Sentence Transformer model is trained on all job title pairs, for each language, it generates embeddings for each job title in queries and corpus element files. Embeddings are dense vector representations that capture the semantic meaning of the job titles in a high-dimensional space. To measure the similarity between the queries and the corpus job titles, cosine similarity is used.

Cosine similarity computes similarity by measuring the angle between two vectors in a multi-dimensional space. The formula to calculate cosine similarity between two vectors A and B is given in Equation (1) below:

$$\text{cosine\_similarity}(A, B) = \frac{A \cdot B}{\|A\| \times \|B\|} \tag{1}$$

For each language in validation data, each job title in the query file is compared against all job titles in the corpus element file, and the corresponding cosine similarity scores are calculated and stored. Job title pairs with higher cosine similarity scores indicate stronger semantic similarity, while lower similarity scores reflect dissimilar job title pairs.

After calculating similarity for all query-corpus job title pairs, for each query job title, corpus element job titles are ranked based on their similarity scores in descending order, that is job title with highest

similarity score ranked as 1. Figure 4 illustrates the Similarity computation of job titles using fine-tuned model and ranking the job titles.

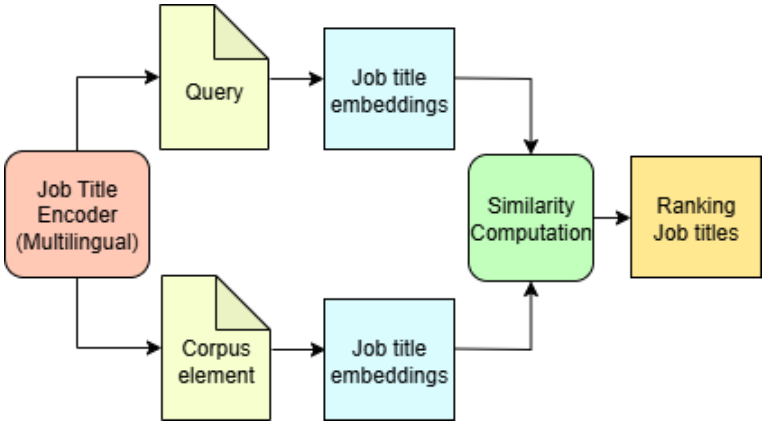

Figure 4: Similarity Computation and Ranking

4. Experimental Results

This section discusses the results obtained after evaluating the fine-tuned model on the validation data.

4.1. System Configuration

We have used the Google Colab environment to train the model and to evaluate its performance. This online environment provided us with an Nvidia Tesla T4 GPU having 15GB of Video RAM. This GPU is better suited for performing deep learning tasks and running inference models. In addition, the CPU provided was Intel Xeon processor clocked at 2.20 GHz and a cache size of 54.96 MB. This CPU supports multi-threading, which enabled faster data processing and computation. Further, the system had 15GB of on-board RAM which was sufficient for us to load and work with the dataset.

4.2. Information retrieval metrics

The following information retrieval metrics are used to observe the retrieval performance of the model.

1. **Mean Average Precision (MAP):** MAP is the mean of the average precision (AP) scores across all queries. It tells how well the model ranks relevant items across multiple queries, rewarding relevant results appearing higher in the ranking. MAP is computed using Equation (2) as given below:

$$\text{MAP} = \frac{1}{Q} \sum_{q=1}^{Q} \text{AP}_q \tag{2}$$

Where:

- Q is the total number of queries.
- AP_q is the average precision for query q.

Average Precision for a query is computed using Equation (3) as follows:

$$\text{AP}_q = \frac{1}{|R_q|} \sum_{k=1}^{N} P(k) \cdot \text{rel}(k) \tag{3}$$

Where:

- $|R_q|$ is the number of relevant items for query q.

- $P(k)$ is the precision at rank k.
- $rel(k) = 1$ it item at rank k is relevant else 0.

Higher values mean relevant results are being ranked closer to the top — and consistently so across queries.

2. **Mean Reciprocal Rank (MRR)**: MRR measures how early the first relevant result appears in the ranking for each query, then averages that over all queries. MRR is computed using Equation (4) given below:

$$\text{MRR} = \frac{1}{Q}\sum_{q=1}^{Q}\frac{1}{\text{rank}_q} \tag{4}$$

Where:

- $rank_q$ is the rank position of the first relevant item for query q.
- Q is the total number of queries.

If the system often gets correct answer right at the top (rank 1), MRR will be high. It strongly penalizes late-ranked relevant results.

3. **Precision@K**: Precision@K is the proportion of relevant items in the top K retrieved results for a query. It is calculated as shown in Equation (5):

$$\text{Precision@K} = \frac{\text{Number of relevant items in top K}}{K} \tag{5}$$

Table 1
Performance Metrics of different models on validation data

Models	Language	MAP	MRR	Precision@5	Precision@10	Precision@100
paraphrase-multilingual-MiniLM-L12-v2	English	0.5434	0.8111	0.6857	0.6229	0.1758
	Spanish	0.4250	0.5513	0.6454	0.6049	0.2262
	German	0.3462	0.5136	0.5468	0.5438	0.2073
paraphrase-multilingual-mpnet-base-v2	English	0.5815	0.8004	0.7162	0.6295	0.1871
	Spanish	0.4737	0.5529	0.6714	0.6486	0.2485
	German	0.4183	0.5181	0.5803	0.6025	0.2448

To compare the performance of paraphrase-multilingual-mpnet-base-v2, the paraphrase-multilingual-MiniLM-L12-v2 model was also fine-tuned using the same approach. Table 1 presents the performance metrics for both models across three languages: English, Spanish, and German. The fine-tuned paraphrase-multilingual-mpnet-base-v2 model outperforms the paraphrase-multilingual-MiniLM-L12-v2 in all evaluated metrics. On the validation data, the average MAP across all three languages is 0.49 for the paraphrase-multilingual-mpnet-base-v2 model, compared to 0.44 for the paraphrase-multilingual-MiniLM-L12-v2 model, demonstrating the superior ranking ability of the model. On the test data, the fine-tuned paraphrase-multilingual-mpnet-base-v2 model achieves an average MAP of 0.45 across all three languages.

The MRR values show that the paraphrase-multilingual-mpnet-base-v2 model consistently returns more relevant results at higher ranks, with an MRR of 0.8004 in English compared to 0.8111 for paraphrase-multilingual-MiniLM-L12-v2, and improved results in Spanish and German as well. While paraphrase-multilingual-MiniLM-L12-v2 slightly outperforms paraphrase-multilingual-mpnet-base-v2 in MRR for English, paraphrase-multilingual-mpnet-base-v2 shows stronger performance overall when considering the complete set of metrics.

This trend is consistent across the other evaluation metrics, indicating that the paraphrase-multilingual-mpnet-base-v2 model provides better semantic understanding and retrieval performance.

5. Conclusion and Future Scope

In this work, a Sentence Transformer model named paraphrase-multilingual-mpnet-base-v2 is used to create a multilingual job title matching system. To ensure that the model captures the semantic

relationships between job titles across different languages, it is fine-tuned using a similar job title pairs dataset constructed from the ESCO dataset for English, Spanish and German. With high-quality embeddings produced by the fine-tuned model for job titles, it enables precise similarity computation and effective retrieval in a multilingual context. The performance of the model is measured by standard information retrieval metrics, such as MAP, MRR, and Precision@K. Training was performed using Cached Multiple Negatives Ranking Loss with a batch size of 128, which allowed the model to learn from a larger set of implicit negatives. The overall average MAP over all three languages is 0.49 on the validation data and 0.45 on the test data, reflecting the model's generalization capacity in multilingual settings. Compared to the paraphrase-multilingual-MiniLM-L12-v2 baseline, the paraphrase-multilingual-mpnet-base-v2 model consistently achieved higher scores across most metrics, confirming its robustness in semantic matching tasks.

Future work can focus on adding domain-specific job title data, expanding to more languages, and experimenting with different contrastive learning losses to enhance representation quality. Furthermore, adding external labor market databases could increase retrieval accuracy and increase the model's industry adaptability.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] S. Böhm, O. Linnyk, J. Kohl, T. Weber, I. Teetz, K. Bandurka, M. Kersting, Analysing gender bias in it job postings: A pre-study based on samples from the german job market, in: Proceedings of the 2020 Computers and People Research Conference, 2020, pp. 72–80.

[2] Z. Tasheva, V. Karpovich, Transformation of recruitment process through implementation of ai solutions, Journal of Management and Economics 4 (2024) 12–17.

[3] M. Nützi, U. Schwegler, S. Staubli, R. Ziegler, B. Trezzini, Factors, assessments and interventions related to job matching in the vocational rehabilitation of persons with spinal cord injury, Work 64 (2019) 117–134.

[4] W. Lee, Y. J. Yoon, Structural change in the job matching process in the united states, 1923–1932, European Review of Economic History 26 (2022) 107–123.

[5] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers), 2019, pp. 4171–4186.

[6] T. Pires, E. Schlinger, D. Garrette, How multilingual is multilingual bert?, arXiv preprint arXiv:1906.01502 (2019).

[7] 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, arXiv preprint arXiv:1911.02116 (2019).

[8] 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.

[9] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, arXiv preprint arXiv:1908.10084 (2019).

[10] D. Deniz, F. Retyk, L. García-Sardiña, H. Fabregat, L. Gasco, R. Zbib, Combined unsupervised and contrastive learning for multilingual job recommendation (2024).

[11] M. Bocharova, E. Malakhov, V. Mezhuyev, Vacancysbert: the approach for representation of titles and skills for semantic similarity search in the recruitment domain, arXiv preprint arXiv:2307.16638 (2023).
[12] L. Gascó, F. M. Hermenegildo, G.-S. Laura, D. C. Daniel, P. Estrella, R. Alvaro, Z. Rabih, Talentclef 2025 corpus: Skill and job title intelligence for human capital management, 2025. URL: https://doi.org/10.5281/zenodo.15038364.
[13] Esco– european skills, competences, qualifications, and occupations, european union, 2024. URL: https://esco.ec.europa.eu/en/use-esco/download, (Accessed on 25 February, 2025).