A Retrieval-Based Approach to Medical Procedure Matching in Romanian

Andrei Niculae[1], Adrian Cosma[1,2], Emilian Radoi[1]
[1]National University of Science and Technology POLITEHNICA Bucharest
[2]Dalle Molle Institute for Artificial Intelligence Research (IDSIA)
andrei.niculae1004@stud.acs.upb.ro, adrian.cosma@supsi.ch, emilian.radoi@upb.ro

Abstract

Accurately mapping medical procedure names from healthcare providers to standardized terminology used by insurance companies is a crucial yet complex task. Inconsistencies in naming conventions lead to missclasified procedures, causing administrative inefficiencies and insurance claim problems in private healthcare settings. Many companies still use human resources for manual mapping, while there is a clear opportunity for automation. This paper proposes a retrieval-based architecture leveraging sentence embeddings for medical name matching in the Romanian healthcare system. This challenge is significantly more difficult in underrepresented languages such as Romanian, where existing pretrained language models lack domain-specific adaptation to medical text. We evaluate multiple embedding models, including Romanian, multilingual, and medical-domain-specific representations, to identify the most effective solution for this task. Our findings contribute to the broader field of medical NLP for low-resource languages such as Romanian.

1 Introduction

Ensuring accurate mapping between medical procedure names used by different healthcare providers and a standardized terminology set maintained by health insurance companies is a challenging task, with real-world applications. Discrepancies in naming conventions can lead to administrative inefficiencies, misclassification of procedures, and potential barriers for patients seeking insurance coverage. These mismatches can result in denied claims, increased processing times, and overall inefficiencies in the healthcare reimbursement process. For example, "The State of Claims: 2024" report [1] reveals that 46% of denied claims are due to missing or innacurate data and coding errors.

Matching procedure names is similar to the well-known problems of entity resolution and text matching, yet it presents unique challenges in the medical domain. The complexity stems from several factors: *(i)* medical terminology is highly domain-specific and varies across institutions, *(ii)* data distributions are often imbalanced due to the frequency of common procedures overshadowing rare ones, *(iii)* nomenclatures evolve over time, necessitating adaptive matching techniques, and *(iv)* the presence of noise in text data, including typographical errors and abbreviations further complicates standardization efforts. Figure 1 illustrates this problem. While previous studies have addressed similar challenges (Tavabi et al., 2024; Levy et al., 2022; Zaidat et al., 2024), most focus on healthcare systems in the United States or other widely studied regions (Alexander et al., 2003). International standards are typically adapted by each country, and private insurance companies may develop their own coding schemes, making a universal solution impractical.

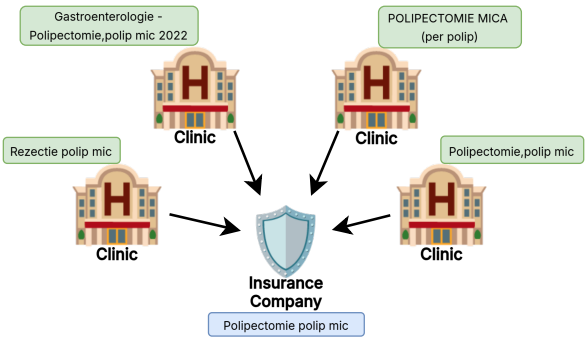

Figure 1: Diagram of the medical procedure matching problem. Clinics often have their own local names for medical procedures that are changed annually, for which a central insurance agency must match to a standardized list of procedures for reimbursement.

This issue is particularly pressing for underrepresented languages such as Romanian. Despite growing interest in NLP for low-resource languages (Nigatu et al., 2024), Romanian remains significantly underrepresented in medical NLP research. Ex-

[1]The State of Claims: 2024, Accessed 19.03.2025

Proceedings of the 24th Workshop on Biomedical Language Processing (BioNLP 2025), pages 167–175

isting language models such as RoBERT (Masala et al., 2020) and RoLLaMA (Masala et al., 2024) provide general-purpose Romanian embeddings, but they lack the necessary specialization for medical text processing. Often, for real-world scenarios, multilingual models are ubiquitously used (Wang et al., 2024, 2020), even if they might fail to capture language-specific nuances.

In this paper, we propose a retrieval-based architecture for medical procedure matching. By leveraging metric learning and dense vector representations of procedure names (Ramesh Kashyap et al., 2024), our method can handle a variable number of input-output mappings, can be expanded without retraining the entire model, and integrate efficiently with scalable vector search frameworks such as Milvus (Wang et al., 2021). This makes retrieval an attractive paradigm for medical name matching, as it enables continuous updates and adaptation to changing medical taxonomies without extensive human intervention. We empirically evaluate three sentence embedding models (Wang et al., 2024; Masala et al., 2020; Alsentzer et al., 2019), comparing their effectiveness in Romanian medical name matching.

By focusing on the Romanian healthcare system, our study highlights the broader challenges of medical terminology standardization and provides insights that can inform similar efforts in other low-resource languages. We aim to contribute to the development of robust, scalable, and language-aware retrieval methods for healthcare applications, ultimately improving the efficiency and accessibility of medical insurance systems.

Our contributions are as follows:

1. We propose a retrieval-based architecture for matching medical procedure names across different healthcare providers and insurance companies, addressing a pressing real-world problem in the Romanian healthcare system.

2. We conduct an extensive evaluation of various sentence embedding models, both Romanian (Masala et al., 2020), multilingual (Wang et al., 2024) and specialized in the medical domain (Alsentzer et al., 2019), highlighting their performance in the context of Romanian medical text matching.

2 Related Work

Sentence embedding models. Semantic text embedding models (Ramesh Kashyap et al., 2024) are a significant component of many NLP applications, most notably text retrieval and question answering. Text embeddings are used to capture semantic representations of text that go beyond surface level word and character matching methods such as TF-IDF. Currently, practitioners are using pretrained transformer models such as BERT (Reimers and Gurevych, 2019), either by aggregating word-level representations with a pooling operation, or by using specialized training for text similarity (Khattab and Zaharia, 2020). Currently, the best performing models are aggregated in the MTEB leaderboard (Muennighoff et al., 2023a), a benchmark of several text embedding tasks, including several non-English datasets. For the medical and scientific domain (Lewis et al., 2020), several models have been developed. Models such as SciBERT (Beltagy et al., 2019), BioBERT (Alsentzer et al., 2019), ClinicalBERT (Alsentzer et al., 2019) and MedBERT (Rasmy et al., 2021) offer domain-specific embeddings by training on either specialized biomedical corpora or task-specific datasets.

However, most contextualized text representation models for the medical domain are focused on the English language, with under-represented languages severely lacking in resources such as specialized models or training datasets. In our setup, medical procedure names are written in Romanian, a low resource language, with only a few pretrained language models (Masala et al., 2024, 2020). Currently, for Romanian, only a pretrained RoBERT model (Masala et al., 2020) is available for direct contextualized text representations, but no such variant exists for the medical domain. Currently, multilingual models such as E5 (Wang et al., 2024) and MiniLM (Wang et al., 2020) are ubiquitously used for non-English tasks.

Medical Procedure Matching. The task of medical procedure matching has been performed in the context of assigning medical notes or pathology reports to a predefined set of medical procedures (Tavabi et al., 2024; Levy et al., 2022; Zaidat et al., 2024), with a focus on the US medical system.

Tavabi et al. (2024) investigated the problem of mapping unstructured operative notes to Current Procedural Terminology (CPT) codes. The CPT code set is a system used to describe medical, surgical and diagnostic services, that are used for billing and insurance reimbursement processes in healthcare. The authors apply common NLP techniques to assign 44,002 notes to 100 most prevalent CPT codes, treating this problem as a classification task.

Masterlist Entry	Associated Clinic Procedures Names
Polipectomie polip mic *(Small polyp polypectomy)*	Polipectomie,polip mic *(Polypectomy, small polyp)*
	Gastroenterologie - Polipectomie,polip mic 2022 *(Gastroenterology - Polypectomy, small polyp 2022)*
	Rezectie polip mic *(Small polyp resection)*
Radiografie omoplat 1 incidenta *(Scapula X-ray 1 view)*	Radiografie omoplat (fata sau profil) *(Scapula X-ray (frontal or lateral))*
	Omoplat profil *(Lateral scapula view)*
	RX omoplat profil *(Lateral scapula X-ray)*
Vitamina B12 *(Vitamin B12)*	Vitamina B12 serica (5 zile) *(Serum Vitamin B12 (5 days))*
	Vitamina B12 (Cianocobalamina) *(Vitamin B12 (Cyanocobalamin))*
	ANALIZA SANGE - Vitamina B12 *(BLOOD ANALYSIS - Vitamin B12)*

Table 1: Selected examples of entries in the masterlist and associated procedure names from clinics. There is significant variation in procedure names, which makes simple text matching inappropriate. We provide English translations for convenience.

Using TF-IDF, Doc2Vec (Le and Mikolov, 2014) and Clinical Bio-BERT (Alsentzer et al., 2019) embeddings as input they train a support vector machine classifier, for each embedding type. In their experiments, TF-IDF outperformed both BERT and Doc2Vec.

Levy et al. (2022) used machine-learning models for predicting CPT codes from pathology reports. Their study analyzed 93,039 pathology reports from the Dartmouth-Hitchcock Department of Pathology and Laboratory Medicine, classifying 42 CPT codes. They evaluated the performance of XGBoost and BERT—using both diagnostic text alone and all report subfields. Their findings indicated that while BERT outperformed XGBoost when trained only on diagnostic text, but using all report subfields resulted in XGBoost achieving the best performance.

Zaidat et al. (2024) have also explored assigning CPT codes to spine surgery operative notes, using XLNet (Yang et al., 2019), a bidirectional LSTM (Hochreiter and Schmidhuber, 1997) model. They fine-tune the model to their operative note dataset, containing 922 entries.

Previous studies have evaluated the performance of statistical, machine learning and deep learning models on classification of a large number of samples to a relatively small subset of CPT codes. In contrast, we formulate our problem as a retrieval problem, since our dataset is severely imbalanced, and contains two orders of magnitude more CPT codes (38,814 entries). Furthermore, another advantage of this formulation is that by avoiding a fixed set of classes, the addition of more procedures does not require modifying the architecture or retraining the model. Unique to our work, we are the first to tackle this problem in Romanian, a severely low-resource language in terms of specialized models for the medical domain.

3 Method

In this section, we provide an overview of the problem description, our dataset of medical procedure names and we describe the architecture for performing mapping between clinic descriptions and a set of standardized procedure names.

3.1 Problem Description

The problem of matching medical procedure names to a standardized masterlist is non-trivial. Simple text matching is insufficient, as we will demonstrate in Section 4. Our dataset is comprised of medical procedures and tests from 528 Romanian private clinics, containing 145,298 unique procedure names mapped to their corresponding masterlist entries. Through manual filtering of incorrect mappings, we reduced the dataset to 139,210 entries. Healthcare providers frequently use varying terms, abbreviations, and phrasing for the same procedure, which creates inconsistencies. To illustrate the difficulty, Table 1 shows some relevant examples of mappings. Healthcare providers may omit obvious terms, such as "polyp resection" being synonymous with "polypectomy". Similarly, entries such as "frontal or lateral X-ray" must be mapped to "1 view X-ray", as they represent a sin-

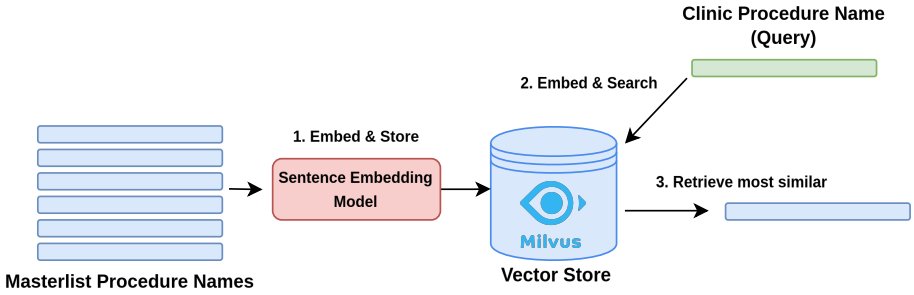

Figure 2: Overall diagram of our method. We formulate medical procedure matching as a retrieval problem: entries in the masterlist are embedded and stored in a vector store and the most similar entry is retrieved based on the similarity with a procedure name from a clinic.

gle test being performed. Terminology variations, like vitamin B12 also being called Cyanocobalamin, can add complexity, especially when descriptions include irrelevant details that mislead text-based matching. Several other relevant examples are presented in Table 2: matching a single medical procedure, when the description actually describes two procedures, not recognizing the semantic meaning of descriptions, ignoring important numerical thresholds, retrieving specific procedures instead of general ones (or vice versa), and prioritizing less important terms.

We chose to model our problem as a retrieval problem, and not as a classification problem, since 50% of elements from the masterlist have only 1 unique procedure assigned. Figure 3 shows the distribution of clinic descriptions assigned to masterlist entries. If we frame our task as a classification problem, we have 39,097 distinct classes, with 19,493 containing only a single sample. Given the severe class imbalance per procedure, a classification model would be inappropriate and would generalize poorly. However, a retrieval-based method can be effectively used by leveraging semantic text embeddings and metric-learning approaches to capture the similarity between clinic descriptions and masterlist entries.

3.2 Procedure Matching as Retrieval

In our retrieval setup, we used the provided masterlist procedure names to build a retrieval index (Wang et al., 2021) and clinic descriptions as the references for the queries. We embed the descriptions using dense (Masala et al., 2020; Wang et al., 2024; Alsentzer et al., 2019) and sparse models (Robertson and Zaragoza, 2009). At inference time, we embed the query clinic descriptions that require a masterlist description and perform a similarity search. The vector DB returns the top-k most similar results for each of our clinic description. Figure 2 showcases this approach.

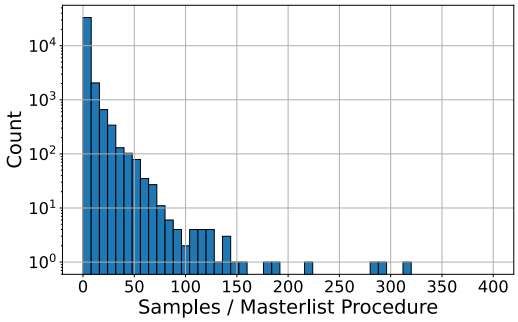

Figure 3: Distribution of number of unique clinic descriptions per masterlist procedure. There is a severe data imbalance: 19,493 (50%) out of 39,097 entries contain only a single example.

The vector index includes two types of entries: masterlist entries and clinic description $\leftrightarrow$ masterlist pairs. In the first scenario, the similarity score is calculated between the query and the masterlist entries, with the index returning the most similar masterlist entries. In the second scenario, the similarity score is computed between the query and the clinic descriptions stored in the index, and the masterlist entry associated with the most similar clinic description is returned. We build our search and evaluation architecture over Milvus (Wang et al., 2021), a high-performance vector database.

For our setup, we used three types of text embeddings: *(i)* sparse text embeddings using BM25 (Robertson and Zaragoza, 2009), *(ii)* dense semantic embeddings with several pretrained transformer models (Masala et al., 2020; Wang et al., 2024; Alsentzer et al., 2019), both zero-shot and fine-tuned with metric learning, and *(iii)* a hybrid ranking approach using RRF (Cormack et al., 2009).

3.3 Sparse Embeddings with BM25

For computing the sparse embeddings, we use BM25 (Robertson and Zaragoza, 2009) to identify the most relevant word-level features from the training set descriptions and masterlist entries. Text is preprocessed by removing diacritics, punctuation, and Romanian stopwords, followed by stemming the remaining words. At inference time, we compute the inner product between the query descriptions and the masterlist descriptions, as well as the clinic description pairs.

3.4 Dense Embeddings with Pretrained Transformer Models

Recent studies have highlighted the challenges of selecting optimal sentence embedding models for domain-specific retrieval tasks (Wornow et al., 2023). Generic benchmarks do not always align with real-world performance, necessitating task-specific evaluations. The MTEB Leaderboard (Muennighoff et al., 2023b) ranks top-performing embedding models based on retrieval performance across various datasets.

We experiment with three models for dense embeddings: mE5-large (Wang et al., 2024), RoBERT-large (Masala et al., 2020), and BioClinicalBERT (Alsentzer et al., 2019). We select mE5 due to its strong performance on multilingual retrieval tasks, RoBERT as a strong language-specific baseline model pre-trained using only Romanian text, and BioBERT as a domain-specific model, pretrained on biomedical text, which may capture medical terminology better than general-purpose models.

Fine-tuning with Metric Learning. We fine-tune the pretrained text embedding models using the MultipleNegativesRankingLoss objective (Henderson et al., 2017), as shown in Figure 4. We consider the clinic descriptions as anchors (a_i) and the corresponding masterlist descriptions (p_i) as positive pairs - (a_i, p_i). The negative pair consists of every combination (a_i, p_j), where $p_j, j \neq i$ are all other masterlist descriptions. In this way, our embedding model learns to increase the cosine similarity between the clinic descriptions and their mapped masterlist description, while decreasing the similarity between the clinic description and all other masterlist items. The model is fine-tuned on 80,911 pairs for 20 epochs, using a batch size of 4096. We use a learning rate of 2e-5, with a cosine scheduler and a warmup ratio of 0.1. All experiments are run on an NVIDIA A100 80GB GPU.

3.5 Hybrid search

Cormack et al. (2009) proposed Reciprocal Rank Fusion (RRF) as a method of aggregating the ranking results of multiple information retrieval systems. It is calculated using the formula:

$$\text{RRFscore}(d \in D) = \sum_{r \in R} \frac{1}{k + r(d)} \quad (1)$$

where D is the set of results to be ranked, R represents the multiple returned rankings of these results, k is a constant, and $r(d)$ is the rank of a result d. We combine the results of dense and sparse embeddings using RRF and analyze its effect on retrieval accuracy.

4 Experiments and Results

To evaluate our approach, we split the dataset into a training and evaluation split, containing 80,911 and 58,299 clinic description $\leftrightarrow$ masterlist pairs, respectively. For fine-tuning, we used only the training split. For evaluation, we split the evaluation set into gallery and probe sets in a 4:1 ratio, in a setup similar to 5-fold cross-validation, where gallery entries form the vector store data. Each fold is stratified based on the masterlist entries, such that each fold contains approximately the same distribution of masterlist entries. Specifically, for each masterlist entry, we distribute its associated clinic descriptions evenly across all folds – for example if 5 clinic descriptions map to the same masterlist entry, each fold will contain exactly 1 such mapping.

Evaluation Metrics. Our primary evaluation metric is Accuracy@k, which measures whether a ground-truth masterlist description is in the first k returned results for a query clinic description. Our target is to optimize for Acc@1, but we also include the results for Acc@3, Acc@5 and Acc@100. In a real-life use of such an system will involve suggesting top-3 or top-5 most similar masterlist entries, and Acc@3 and Acc@5 provides insight into the usefulness of our system. We also include Acc@100, as a low value indicates a problem with the chosen search technique, but usually it indicates the presence of incorrect annotations. In all our results, we show the mean and standard deviation across 5 folds.

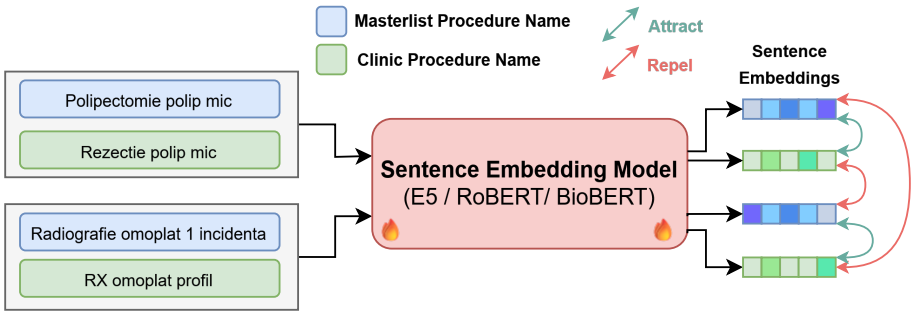

Figure 4: Fine-tuning approach for dense sentence embeddings. A pretrained text embedding model is trained to minimize the distance between representations of masterlist entries and associated clinic procedure names while maximising the distance between every other entry.

Clinic Description	BM25 Miss	mE5 Hit	Observations
Aplicare sterilet + EEV control (*IUD application + EEV control*)	Aplicare sterilet (*IUD application*)	Montare sterilet (DIU) + ecografie control (*IUD insertion (DIU) + ultrasound control*)	Did not account for the additional ultrasound control term
EXOSTOZA (*Exostosis*)	Sonoterapie in exostoze calcaneene (*Sonotherapy in calcaneal exostoses*)	Excizia exostozei (*Excision of exostosis*)	Focused on "exostoses" but did not recognize "excision" as a relevant treatment
Chiuretare molluscum contagiosm > 10 leziuni (*Curettage of molluscum contagiosum > 10 lesions*)	Chiuretare < 10 leziuni (*Curettage of < 10 lesions*)	Chiuretare molluscum contagiosum peste 10 leziuni (*Curettage of molluscum contagiosum over 10 lesions*)	Matched on "curettage" but ignored the numerical threshold
Radiofrecventa ablatie tumori (*Radiofrequency ablation of tumors*)	Ablatie laser / radiofrecventa tumora ureche dificultate redusa (*Laser/radiofrequency ablation of ear tumor - low difficulty*)	Excizie leziune cu radiofrecventa (*Excision of lesion with radiofrequency*)	Retrieved a more specific procedure (ear tumor) instead of a general one
RM articulatii sacro iliace cu subst. de contrast (*MRI of sacroiliac joints with contrast*)	Artrodeza articulatiei sacro iliace percutanata cu implant I Fuse (*Percutaneous sacroiliac joint arthrodesis with I Fuse implant*)	RMN articulatii sacroiliace cu SC, 1.5T (*MRI of sacroiliac joints with SC, 1.5T*)	Retrieved a surgical procedure instead of an imaging scan

Table 2: Selected examples of clinic Descriptions with BM25 Misses, mE5 Dense Embedding Hits. Sparse indexes are not appropriate for this task, which require high level semantic understanding of descriptions.

4.1 Comparison between different types of search indexes

In Table 3, we show a comparison between dense, sparse, and hybrid approaches. For dense embeddings, we used a fine-tuned mE5 (Wang et al., 2024) model. The results show that the fine-tuned dense model consistently outperforms both sparse and hybrid search methods. When searching only masterlist entries, the dense approach achieves 26.2% higher Acc@1 than the sparse approach. When using both masterlist and associated mappings, the dense approach obtains a 17.2% Acc@1 margin. The sparse approach also shows poor performance for Acc@100, indicating that a bag-of-word approach is not appropriate for this task, and semantic understanding is needed. Hybrid search fails to outperform dense search as it is limited by the poor performance of sparse search. In Table 2 we show selected examples of clinic descriptions where sparse embeddings fail to capture variations in text descriptions.

4.2 Fine-tuning with metric learning

In Table 4 we compare the performance of three dense embedding models: mE5-large (Wang et al., 2024), RoBERT-large (Masala et al., 2020), and BioClinicalBERT (Alsentzer et al., 2019). We obtained that mE5 has higher off-the-shelf retrieval accuracy compared to RoBERT and BioClinicalBERT. This advantage stems from mE5's design as a sentence-transformer model specifically trained to evaluate similarity between sentences or descriptions, whereas RoBERT and BioClinicalBERT is adapted for sentence embedding through a pooling operation over token embeddings.

Sparse search initially outperforms both RoBERT and BioClinicalBERT. However, after fine-tuning, all dense embeddings surpass sparse embeddings in performance metrics, with mE5 maintaining its position as the highest-performing model.

Table 5 illustrates the impact of incorporating both masterlist and associated mappings in search processes. The inclusion of reduces the performance difference between E5 and the other two models. While the relative ranking of models remains consistent, E5 achieves the highest perfor-

Vector Store Data	Index Type	Acc@1	Acc@3	Acc@5	Acc@100
Masterlist Entries Only	sparse (BM25)	52.6 ± 0.002	64.5 ± 0.002	68.5 ± 0.002	86.3 ± 0.001
	dense (mE5)	78.8 ± 0.002	92.2 ± 0.002	95.0 ± 0.002	99.5 ± 0.001
	hybrid (RRF)	63.9 ± 0.003	77.7 ± 0.003	82.1 ± 0.003	99.5 ± 0.001
Masterlist Entries + Mappings	sparse (BM25)	68.0 ± 0.003	82.3 ± 0.001	86.1 ± 0.001	94.7 ± 0.001
	dense (mE5)	**85.2 ± 0.003**	**95.8 ± 0.001**	**97.5 ± 0.001**	**99.5 ± 0.001**
	hybrid (RRF)	81.0 ± 0.002	92.3 ± 0.001	94.9 ± 0.001	99.5 ± 0.000

Table 3: Comparison between sparse embeddings from BM25, dense embeddings from mE5 (Wang et al., 2024), and hybrid search, having only masterlist entries in the vector store and having both masterlist and associated clinical mappings. Using dense embeddings from mE5 provides the best results in both cases. Results are averaged across 5 folds.

Model Name	Type	Acc@1	Acc@3	Acc@5	Acc@100
RoBERT (Masala et al., 2020)	off-the-shelf	44.7 ± 0.003	53.4 ± 0.003	56.9 ± 0.004	75.3 ± 0.003
BioClinicalBERT (Alsentzer et al., 2019)	off-the-shelf	47.7 ± 0.003	56.7 ± 0.003	60.2 ± 0.002	74.9 ± 0.003
mE5 (Wang et al., 2024)	off-the-shelf	56.8 ± 0.003	69.4 ± 0.002	74.3 ± 0.002	91.3 ± 0.002
RoBERT (Masala et al., 2020)	fine-tuned	75.9 ± 0.001	89.9 ± 0.002	93.2 ± 0.000	98.9 ± 0.001
BioClinicalBERT (Alsentzer et al., 2019)	fine-tuned	75.7 ± 0.002	89.2 ± 0.002	92.7 ± 0.002	98.9 ± 0.000
mE5 (Wang et al., 2024)	fine-tuned	**78.8 ± 0.002**	**92.2 ± 0.002**	**95.0 ± 0.002**	**99.5 ± 0.001**

Table 4: Comparison between different types of text embedding models, having entries in the vector store only from the masterlist entries. We obtained the best results using a fine-tuned version of mE5, a general-purpose multi-lingual model. Results are averaged across 5 folds.

Model Name	Type	Acc@1	Acc@3	Acc@5	Acc@100
RoBERT (Masala et al., 2020)	off-the-shelf	62.5 ± 0.005	76.8 ± 0.004	81.1 ± 0.005	92.0 ± 0.004
BioClinicalBERT (Alsentzer et al., 2019)	off-the-shelf	66.7 ± 0.005	80.8 ± 0.003	84.6 ± 0.002	93.4 ± 0.002
mE5 (Wang et al., 2024)	off-the-shelf	67.9 ± 0.004	85.2 ± 0.002	89.6 ± 0.002	98.1 ± 0.001
RoBERT (Masala et al., 2020)	finetuned	84.4 ± 0.002	94.8 ± 0.002	96.6 ± 0.001	99.0 ± 0.001
BioClinicalBERT (Alsentzer et al., 2019)	finetuned	83.8 ± 0.003	94.3 ± 0.001	96.4 ± 0.001	99.0 ± 0.001
mE5 (Wang et al., 2024)	finetuned	**85.2 ± 0.003**	**95.8 ± 0.001**	**97.5 ± 0.001**	**99.5 ± 0.001**

Table 5: Comparison between different types of text embedding models, having entries in the vector store from both the masterlist entries and associated clinical mappings. We obtained the best results using a fine-tuned version of mE5, a general-purpose multi-lingual model. Results are averaged across 5 folds.

mance with an Acc@1 of 85.2% and an Acc@5 of 95%. Notably, Acc@1 metric may under-represent actual performance. Manual inspection of misclassified results reveals many plausible matches. This discrepancy occurs due to the presence of duplicate entries within the masterlist itself—entries with slightly different formulations that reference identical medical procedures. The markedly higher Acc@3 metric, which captures whether the ground-truth result appears within the first three recommendations, supports this observation. Although duplicated masterlist results present a methodological challenge for evaluation, they do not compromise practical application. The real-world accuracy exceeds the reported metrics, as demonstrated in the next section.

4.3 Doctor evaluation

Our medical procedure mapping system was used to map new unmapped procedures. We evaluated on new procedure descriptions from 10 clinics, comprising 12,836 unique descriptions. After mapping the procedures using a fine-tuned mE5 models trained on all available data, doctors validated each pair to determine if the masterlist assignment was correct. As shown in Table 6, the model achieves a real-world Acc@1 of 94.7%. The 98.5% Acc@2 indicates that doctors considered either the first or second recommendation correct, while for only 1% of entries, doctors assigned a different description than the ones recommended.

Model Name	Acc@1	Acc@2	Acc@3
mE5 - All Data	94.7	98.5	99.0

Table 6: Real-world evaluation of our system. Doctors manually evaluated 12,836 new entries after mapping them with a fine-tuned version of mE5 on all data.

Another notable aspect is the speed of the map-

pings process. While manually mapping the 12,836 descriptions would take more than 60 hours, using our retrieval system reduces this to only 3 minutes, resulting in an 1200× speedup.

5 Conclusion

This paper presents a retrieval-based approach for medical procedure matching in the Romanian healthcare system, addressing the challenges posed by inconsistent naming conventions across clinics and insurance providers. We demonstrate that dense sentence embeddings, particularly fine-tuned multilingual models, significantly outperform traditional sparse methods such as BM25. Our experiments show that a fine-tuned mE5 model achieves the highest retrieval accuracy, with an Acc@1 of 85.2% when using both masterlist entries and clinical mappings. The real-world evaluation further confirms the efficacy of our approach, achieving a validated accuracy of 94.7% in a doctor-reviewed dataset. Furthermore, our systems enables significant labor efficiency: using our automated matching systems results in 1200× speedup compared to manual matching. Our findings contribute to the broader domain of medical NLP for low-resource languages and offer a viable solution for improving the Romanian healthcare system.

Limitations

Our approach has several limitations. Firstly, errors in historical mappings may propagate into future predictions, potentially reinforcing inaccuracies over time. This challenge necessitates periodic human review and correction to prevent systematic errors. Secondly, cosine similarity between embeddings may not always provide a reliable confidence estimate, due to the considerable overlap between the score distributions of hits and misses. This makes it difficult to differentiate between correct and incorrect matches. Incorporating additional uncertainty modeling or ranking refinements could improve result interpretability. Thirdly, while our retrieval model significantly improves over rule-based methods, its performance is still constrained by the lack of a specialized Romanian medical language model. A dedicated medical NLP model trained on domain-specific Romanian corpora could further enhance accuracy.

Acknowledgement

This research was supported by the project "Search Techniques for Matching Medical Information - MATCHMED", ID 420246344, and by the project "Romanian Hub for Artificial Intelligence - HRIA", Smart Growth, Digitization and Financial Instruments Program, MySMIS no. 334906.

References

Sherri Alexander, Therese Conner, and Teresa Slaughter. 2003. Overview of inpatient coding. *American journal of health-system pharmacy: AJHP: official journal of the American Society of Health-System Pharmacists*, 60(21 Suppl 6):S11–14.

Emily Alsentzer, John R. Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew B. A. McDermott. 2019. Publicly available clinical BERT embeddings. *CoRR*, abs/1904.03323.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Gordon V. Cormack, Charles L A Clarke, and Stefan Buettcher. 2009. 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*, pages 758–759, Boston MA USA. ACM.

Matthew Henderson, Rami Al-Rfou, Brian Strope, Yun-Hsuan Sung, László Lukács, Ruiqi Guo, Sanjiv Kumar, Balint Miklos, and Ray Kurzweil. 2017. Efficient natural language response suggestion for smart reply. *arXiv preprint arXiv:1705.00652*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Omar Khattab and Matei Zaharia. 2020. ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 39–48, Virtual Event China. ACM.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents.

Joshua Levy, Nishitha Vattikonda, Christian Haudenschild, Brock Christensen, and Louis Vaickus. 2022. Comparison of Machine-Learning Algorithms for the Prediction of Current Procedural Terminology (CPT)

Codes from Pathology Reports. *Journal of Pathology Informatics*, 13:3.

Patrick Lewis, Myle Ott, Jingfei Du, and Veselin Stoyanov. 2020. Pretrained language models for biomedical and clinical tasks: understanding and extending the state-of-the-art. In *Proceedings of the 3rd clinical natural language processing workshop*, pages 146–157.

Mihai Masala, Denis C. Ilie-Ablachim, Alexandru Dima, Dragos Corlatescu, Miruna Zavelca, Ovio Olaru, Simina Terian, Andrei Terian, Marius Leordeanu, Horia Velicu, Marius Popescu, Mihai Dascalu, and Traian Rebedea. 2024. "vorbeşti româneşte?" a recipe to train powerful romanian llms with english instructions. *Preprint*, arXiv:2406.18266.

Mihai Masala, Stefan Ruseti, and Mihai Dascalu. 2020. RoBERT – a Romanian BERT model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6626–6637, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. 2023a. MTEB: Massive text embedding benchmark. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2014–2037, Dubrovnik, Croatia. Association for Computational Linguistics.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2023b. MTEB: Massive Text Embedding Benchmark. *Preprint*, arXiv:2210.07316.

Hellina Hailu Nigatu, Atnafu Lambebo Tonja, Benjamin Rosman, Thamar Solorio, and Monojit Choudhury. 2024. The zeno's paradox of 'low-resource' languages. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17753–17774, Miami, Florida, USA. Association for Computational Linguistics.

Abhinav Ramesh Kashyap, Thanh-Tung Nguyen, Viktor Schlegel, Stefan Winkler, See-Kiong Ng, and Soujanya Poria. 2024. A comprehensive survey of sentence representations: From the BERT epoch to the CHATGPT era and beyond. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1738–1751, St. Julian's, Malta. Association for Computational Linguistics.

Laila Rasmy, Yang Xiang, Ziqian Xie, Cui Tao, and Degui Zhi. 2021. Med-bert: pretrained contextualized embeddings on large-scale structured electronic health records for disease prediction. *NPJ digital medicine*, 4(1):86.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3980–3990, Hong Kong, China. Association for Computational Linguistics.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389.

Nazgol Tavabi, Mallika Singh, James Pruneski, and Ata M. Kiapour. 2024. Systematic evaluation of common natural language processing techniques to codify clinical notes. *PLOS ONE*, 19(3):e0298892.

Jianguo Wang, Xiaomeng Yi, Rentong Guo, Hai Jin, Peng Xu, Shengjun Li, Xiangyu Wang, Xiangzhou Guo, Chengming Li, Xiaohai Xu, Kun Yu, Yuxing Yuan, Yinghao Zou, Jiquan Long, Yudong Cai, Zhenxiang Li, Zhifeng Zhang, Yihua Mo, Jun Gu, and 3 others. 2021. Milvus: A Purpose-Built Vector Data Management System. In *Proceedings of the 2021 International Conference on Management of Data*, SIGMOD '21, pages 2614–2627, New York, NY, USA. Association for Computing Machinery.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Multilingual E5 Text Embeddings: A Technical Report. *Preprint*, arXiv:2402.05672.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in neural information processing systems*, 33:5776–5788.

Michael Wornow, Yizhe Xu, Rahul Thapa, Birju Patel, Ethan Steinberg, Scott Fleming, Michael A Pfeffer, Jason Fries, and Nigam H Shah. 2023. The shaky foundations of large language models and foundation models for electronic health records. *npj digital medicine*, 6(1):135.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Bashar Zaidat, Justin Tang, Varun Arvind, Eric A. Geng, Brian Cho, Akiro H. Duey, Calista Dominy, Kiehyun D. Riew, Samuel K. Cho, and Jun S. Kim. 2024. Can a Novel Natural Language Processing Model and Artificial Intelligence Automatically Generate Billing Codes From Spine Surgical Operative Notes? *Global Spine Journal*, 14(7):2022–2030.