Embedding Models: A Comprehensive Review with Task-Oriented Assessment

Lahbib Ajallouda[1], Meriem Hassani Saissi[2], Ahmed Zellou[3]
Poly-Disciplinary Faculty of Es-Semara, Ibn Zohr University in Agadir, Morocco[1]
SPM-ENSIAS, Mohammed V University, Rabat, Morocco[2, 3]

***Abstract*—Sentence embedding is a very important technique in most natural language processing (NLP) tasks, such as answer generation, semantic similarity detection, text classification and information retrieval. This technique aims to transform the semantic meaning of a sentence into a fixed-dimensional vector, allowing machines to understand human language. Sentence embedding has moved in recent years from simple word vector averaging methods to the development of more sophisticated models, particularly those based on transformer structures such as the BERT model and its variants. However, systematic reviews that critical, analyze and compare the performance of these models are still limited, particularly the selection of the appropriate embedding model for a specific NLP task. This study aims to address this gap by a comprehensive review for sentence embedding models and a systematic evaluation of their performance on NLP tasks, such as semantic similarity, clustering, and retrieval. The study enabled us to identify the appropriate embedding model for each task, identify the main challenges faced by embedding models, and propose effective solutions to improve the performance and efficiency of sentence embedding.**

***Keywords*—Natural language processing; sentence embedding models; transformer models; embedding models challenges**

I. Introduction

Embedding is one of the most techniques used in natural language processing applications [1] [2]. The performance of these applications is affected by the quality of the fixed-dimensional vectors generated [3]. Several studies have underscored the importance of developing sentence embedding models [4], [5], especially with the capabilities offered by deep neural networks and transformers. NLP tasks require a deeper understanding of texts, especially understanding long texts [6], [7]. Therefore, choosing an appropriate embedding model is essential for a task. Recent frameworks such as DSPy [8] underscore the need for more sophisticated and context-aware sentence representations that are better suited to specific NLP tasks. Therefore, there is a need to study and analyze sentence embedding techniques, from statistical models based on word vector averages to modern models based on deep structures such as BERT [9] and its variants. In this context, we conducted a comprehensive literature review of the most sentence embedding models, from simple aggregation techniques to transformer-based models.

The main objective of our paper is to identify the most appropriate embedding model for a specific task. To achieve this, our paper first presents the most important embedding models, their characteristics, the challenges that affect their effectiveness, and solutions that can help overcome them. Second, it evaluates the performance of these models on three natural language processing tasks [10], including semantic similarity, clustering, and retrieval, to identify the most effective models for each task. The methodology used in our paper is based on answering three main research questions:

- RQ 1: What are the main categories of sentence embedding models proposed in the literature?
- RQ 2: What are the strengths, limitations, and challenges of each model?
- RQ 3: Which model is appropriate for each NLP task?

We reviewed and analyzed scientific papers published in peer-reviewed sources (ArXiv, Scopus, Dblp, IEEE) between 2015 and 2025. The results of this study represent an opportunity for researchers to build an overview of the techniques used in sentence embedding, to advance this area.

The rest of the paper will be presented as follows. Section II presents related work on sentence embedding. Section III discusses categories of sentence embedding models, highlighting their advantages and challenges. Section IV is devoted to an empirical study that aims to evaluate these models across NLP tasks. Section V presents and discusses the results of the study. Section VI concludes the paper with future research directions.

II. Related Work

In this section, we first highlight the importance of embedding models for natural language processing tasks and applications. We then trace the evolution of sentence embedding models. Finally, we summarize the scope and limitations of previous reviews to underline the relevance of the present study.

A. Sentence Embeddings in NLP Tasks

Embedding models have attracted growing interest in NLP field, as they demonstrate strong effectiveness in a variety of tasks, such as retrieval, clustering, summarization, and semantic similarity [11]. The development of embedding techniques has positively impacted this area [12]. Recent research has demonstrated that their use significantly improves the performance of NLP applications. For example, the performance of text classification and clustering has improved by pre-trained embeddings [13]. Performance in dense retrieval across multiple datasets has improved by employing contrastive learning models [14]. Similarly, extractive summarization models have improved by enabling the selection of more semantically rich sentences compared to standard baselines

[15]. Clustering studies also confirm that embeddings derived from pre-trained large language models capture accurate semantic relationships [16]. These results confirm that exploiting sentence embeddings for a specific task is essential for improving the performance of NLP tasks and applications.

B. Sentence Embedding Evolution

A text embedding is a vector representation of text (word, sentence, or paragraph), where similar texts appear close together in the embedding space [5]. These representations encode the syntactic and semantic properties of linguistic elements. Classic representations have relied on the bag-of-words model or TF-IDF weighting [17], where each word is represented by a high-dimensional vector reflecting its frequency in the document (Formula 1). For a word w in a document d, the TF-IDF vector is defined as:

$$\vec{W} = (TFIDF(w,d_1), TFIDF(w,d_2), \dots, TFIDF(w,d_N)) \quad (1)$$

Where $TF-IDF(w,d)$ is calculated by Formula 2.

$$TF-IDF(w,d) = \frac{f_{w,d}}{\sum_{w' \epsilon d} f_{w',d}} \times log\left(\frac{N}{1+|\{d \epsilon D: w \epsilon d\}|}\right) \quad (2)$$

Where:

- $f_{w,d}$: Frequency of the word w in document d.
- N: Number of documents in corpus D.
- $|\{d \epsilon D: w \epsilon d\}|$: Number of documents containing the word w.

However, these models suffer from several limitations, including their high dimensionality and their inability to capture semantic similarity between terms.

Word embedding, through models such as Word2Vec [18] and GloVe [19], has enabled the representation of words using low-dimensional vectors while considering relevant semantic contexts. On the other hand, the same vector is assigned to a word regardless of its context. Therefore, these models ignore the context in which the word appears, making them unsuitable for modeling sentences or paragraphs. Fig. 1 provides an example of this challenge.

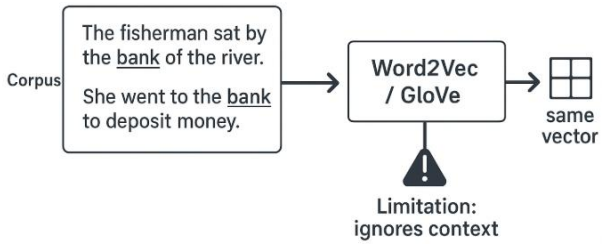

Fig. 1. Contextual limitation of Word2Vec and GloVe.

To address the need to model text units containing more than one word, sentence embeddings were developed that aim to represent each sentence as a fixed-dimensional dense vector.

Sentence embedding techniques are classified into two main categories [20]. The first includes statistical techniques, such as Doc2Vec [21] and SIF [22]. These techniques are computationally inexpensive. However, they remain limited because they only capture a portion of the semantic context. The second set consists of deep neural techniques, which rely on advanced encoding models. These include InferSent [23] and Universal Sentence Encoder [24], Sentence-BERT [25] and SimCSE [26]. These models use transformers to contextually model complex relationships between words and sentences. They therefore provide more expressive vector representations suitable for advanced NLP tasks.

C. Comprehensive Review of Sentence Embedding

Several research papers have studied and reviewed proposed sentence embedding models, offering different perspectives on their classification, evaluation, and application. The work presented in [27] is one of the first general reviews of sentence representations. This study was limited to unsupervised neural approaches. Other reviews such as [23], [28] have analyzed and compared sentence representation learning, proposed unified evaluation metrics such as SentEval [29], and focused on classical RNN and CNN encoders. In contrast, studies, such as [24], [25], have highlighted the effectiveness of transformer learning using pre-trained models such as BERT and its variants. Finally, recent published studies [26], [30] have focused on the importance of contrastive learning and pre-trained models. Despite the comprehensiveness of these reviews, the development of embedding models across NLP tasks, and the difficulty of identifying the appropriate model for each task, is a challenge that motivates a comprehensive and updated review that considers the importance of models to the nature of NLP tasks and their applications.

III. Sentence Embedding Models

This section introduces two main categories of sentence embedding models. Statistical sentence embedding models represent sentences using aggregated word-level statistics, while transformer-based sentence embedding models represent sentences using pre-trained contextual language models. This classification highlights the shift from shallow frequency-based representations to neural structures that better encode semantic and syntactic relationships.

A. Statistical Sentence Embedding Techniques

Statistical methods for sentence embedding rely on simple aggregation techniques, such as averaging or weighted summation of word embeddings, or distributional models. Although computationally efficient and easy to implement, these methods provide an approximate semantic representation of a sentence, ignoring word order and syntactic structures.

1) Unweighted averaging: Unweighted averaging is one of the most techniques widely used to generate sentence embeddings. The method consists of representing a sentence by computing the arithmetic mean of the vector representations of its constituent words. According to study [18], a sentence embedding is defined by Formula 3.

$$v_{sentence} = \frac{1}{n}\sum_{i=1}^{n} v_{w_i} \quad (3)$$

Where:

- n: Number of words in the sentence.
- v_{w_i} : The embedding representation of word w_i.

Despite its simplicity, it has been shown to capture some of the semantic meaning of vectors. However, its limitations include ignoring the significance of words in a sentence. It treats all words equally, including frequent terms.

2) TF-IDF weighted averaging: The TF-IDF weighted averaging incorporates word importance by exploiting the TF-IDF coefficient. This allows us to overcome the challenge of treating all words equally and reduce the influence of common words. Formally, according to [22], the sentence embeddings are obtained by calculating a weighted average of word vectors, where the weights are determined by TF-IDF coefficient (Formula 4).

$$V_{sentence} = \frac{\sum_{i=1}^{n} TF\text{-}IDF(w_i) \cdot v_{w_i}}{\sum_{i=1}^{n} TF\text{-}IDF(w_i)} \quad (4)$$

Where:

- n: Number of words in the sentence.
- v_{w_i} : The embedding representation of word w_i.
- $TF - IDF(w_i)$: Weight of word w_i, is calculated by Formula 2.

This model generates better semantic representations than unweighted averaging. However, it still ignores the syntactic information and sequential order of words, which may limit its effectiveness in more complex NLP tasks.

3) Max and min pooling models: Max and min pooling is a statistical model for generating sentence embeddings by selecting the maximum (or minimum) value of all word embeddings in a sentence. This approach captures the most prominent features present in any word vector, making it robust to noise. According to [23], the sentence embedding can be generated using either max-pooling or min-pooling as defined by Formulas (5) and (6).

$$V_{sentence}^{max} = \left[\max_{i=1..n}(v_{w_i}^{(1)}), \max_{i=1..n}(v_{w_i}^{(2)}), \dots \max_{i=1..n}(v_{w_i}^{(d)})\right] \quad (5)$$

$$V_{sentence}^{min} = \left[\min_{i=1..n}(v_{w_i}^{(1)}), \min_{i=1..n}(v_{w_i}^{(2)}), \dots \min_{i=1..n}(v_{w_i}^{(d)})\right] \quad (6)$$

Where:

- n: Number of words in the sentence.
- $v_{w_i}^{(k)}$: the value of the k-th dimension of word w_i (each of dimension d).

However, it still ignores syntactic information and may oversimplify the semantic representation. The semantic structure of sentences may be overly simplified.

4) Smooth inverse frequency model: The Smoothed Inverse Frequency (SIF) [22] model reduces the influence of common words on sentence embeddings by weighting word embeddings inversely with their frequency in a corpus. To generate sentence embeddings, SIF calculates a weighted average (Formula 7) and removes the first principal component that defines common directions in the embedding space to produce more discriminative sentence representations (Formula 8).

$$V_{sentence} = \frac{1}{n}\sum_{i=1}^{n} \frac{\alpha}{\alpha + p(w_i)} \cdot v_{w_i} \quad (7)$$

Where:

- n: Number of words in the sentence.
- v_{w_i} : Embedding representation of word w_i.
- $p(w_i)$: Frequency of w_i in the corpus.
- α: Smoothing parameter (default 10^{-3})

To remove the first principal component F that defines the common directions in the embedding space, we need to calculate $FF^T V_{sentence}$ the Projection of $V_{sentence}$ onto F ($F^T V_{sentence}$ is the scalar product between F and $V_{sentence}$).

$$V_{Sentence}^{SIF} = V_{sentence} - FF^T V_{sentence} \quad (8)$$

Although SIF is efficient, and robust, but its reliance on frequency statistics, principal component analysis calculations, and inability to capture syntax or context limits its effectiveness for more complex NLP tasks.

B. Transformer-Based Sentence Embedding Models

Sentence embedding using transformers has contributed to generating contextual representations that capture the semantic meaning of sentences and reducing the challenges of statistical models. Bidirectional Encoder from Transformers (BERT) [9] is the first transformer used for sentence embedding. Other transformers have followed, improving the performance of the sentence embedding task. In this part, we provide a comprehensive overview of the transformer-based sentence embedding models, based on the classification shown in Fig. 2.

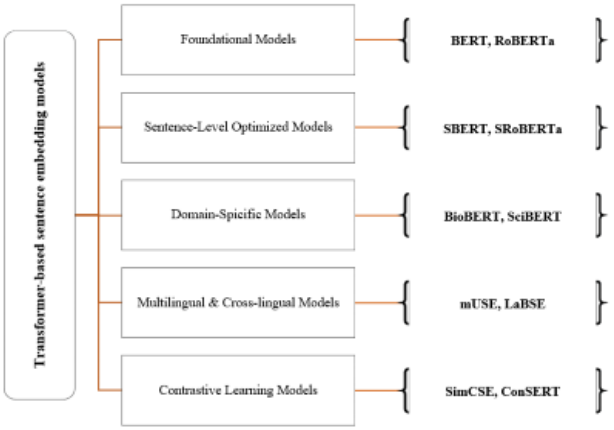

Fig. 2. Taxonomy of transformer-based sentence embedding models.

1) Foundational models: BERT is the first transformer model for sentence embedding. Unlike statistical approaches, BERT relies on self-attention mechanisms rather than recurrent or convolutional networks. It has bidirectional encoding, which allows it to read sequences of tokens in both directions. It uses multiple transformer layers (12 in BERT-base and 24 in BERT-large), each consisting of multi-head self-attention networks and feed-forward networks. Its pre-training combines masked language modeling (MLM), where random tokens are predicted from context, and next-sentence prediction (NSP), which

captures sentence-level relationships. Fig. 3 illustrates the process by which BERT produces these embeddings. These features make BERT more effective to learn deep contextual representations.

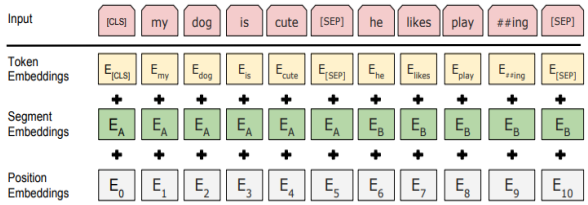

Fig. 3. The BERT sentence embedding process, reproduced from study [9].

However, the computational cost of pretraining, biases in pretraining data, and sequence length constraints limit its performance to represent long documents. This challenge has prompted the development of BERT variants such as RoBERT [31] that uses the same transformer encoder architecture as BERT, but with improvements in the pre-training phase. It Eliminates the NSP task, employs dynamic masking in the MLM objective, and trains on larger datasets (over 160 GB). This allows it to benefit from longer training schedules and higher learning, enhancing the quality of its contextual representations.

Despite their effective performance, Foundational models face several challenges. The large training dataset makes them computationally expensive, and they are also affected by biases present in pre-training datasets. Furthermore, the input length limit of 512 tokens limits their ability to represent longer sequences.

2) Sentence-level optimized models: To overcome the challenges of foundational models, sentence-level optimized models, such as SBERT and SRoBERTa [25], rely on Siamese and triplet network architectures, which improve the accuracy of transformer encoders to generate semantically meaningful sentence representations. They also reduce computational cost on semantic text similarity (STS) tasks. The time to find the most similar pair of sentences in a set of 10,000 sentences is reduced from 65 hours using BERT to about 5 seconds using SBERT. Using a Siamese and triplet network also embeds sentences into higher-dimensional semantic spaces to better capture hierarchical information.

TABLE I. KEY DISTINCTIONS BETWEEN SBERT AND SROBERTA

Criterion	SBERT	SRoBERTa
Encoder	BERT or RoBERTa	RoBERTa (with enhanced pretraining)
Training Data	STS/NLI datasets	Larger and more diverse datasets, often domain-adapted
Specialized Tasks	Semantic similarity	Semantic similarity, clustering
Performance	Strong for semantic tasks	Higher performance, especially in specific domain

Table I shows the key differences between SBERT and SRoBERTa. In SBERT, the input sentence is encoded using BERT or RoBERTa, followed by a pooling layer that produces a fixed-size embedding, which is then applied to semantic similarity tasks. In contrast, SRoBERTa uses the RoBERTa encoder and incorporates structured training strategies, such as variance and classification losses, to produce more accurate embeddings. These embeddings are particularly effective for semantic search and clustering, and therefore SRoBERTa outperforms SBERT.

The performance of SBERT and SRoBERTa on Cross-lingual data is low, compared to their results on English data. Therefore, their performance depends on the language of their training data (English) which makes their generalization to complex languages difficult. Therefore, multilingual and cross-linguistic sentence embedding models are important to achieve robust performance on multilingual applications.

3) Multilingual and cross-linguistic models: Semantic similarity Detection between sentences across different languages has been a challenge for monolingual sentence embedding models. In contrast, multilingual and cross-linguistic models, such as the Multilingual Universal Sentence Encoder (mUSE) [32] and Language-Independent BERT Sentence Embeddings (LaBSE) [33], achieve minimal linguistic bias, leading to better performance.

The mUSE model uses a universal sentence encoder and is trained on a set of translation pairs in multilingual datasets. mUSE supports more than 16 universal languages, and achieves a semantic similarity and effective cross-language retrieval. However, its performance deteriorates on untrained languages, this limit its application in global contexts. Unlike mUSE, LaBSE is a language-agnostic model trained on over 110 languages using a dual-encoder architecture and translation ranking objectives. LaBSE achieved state-of-the-art results on benchmarks such as Tatoeba [34] and cross-linguistic STS [35]. It demonstrated its ability to encode semantically equivalent sentences from different languages, making it one of the most robust models for multilingual applications.

Overall, these models have achieved the transition from English-centric sentence embeddings to multilingual sentence embeddings. Despite this contribution, these models still face significant challenges. Their reliance on massive parallel datasets and complex training pipelines limits their adaptability to resource-limited languages. They also rely on translation-based alignment objectives, which can introduce biases and limit the generalization of sentence representations in monolingual or domain-specific. To overcome these challenges, contrastive learning models have emerged, which aim to learn sentence embeddings by exploiting self-supervised objectives.

4) Contrastive learning models: To overcome the challenges faced by multilingual sentence embedding models, contrastive learning models has been adopted as a paradigm for learning universal sentence representations. these models aim to approximate semantically similar sentences in the embedding space while separating unrelated sentences, without the need for massive parallel datasets. SimCSE [26] and ConSERT [36] are the most widely used contrastive learning models for sentence embedding.

SimCSE relies on two different representations of the same sentence by applying contrastive dropout masks within a pre-

trained model such as BERT or RoBERTa. These representations are considered as positive pairs, while the other sentences are treated as negative samples. By the contrastive loss optimization (InfoNCE) [37], SimCSE is learn to align semantically consistent representations without the need for external supervision. The ConSERT sentence embedding model relies on contrastive learning to efficiently fine-tune BERT, leveraging positive pair construction through strategies including word reordering, token masking, and embedding dimension masking. Sentence representations are obtained by aggregating mean pooling instead of CLS. Which makes it able to adapt in low-resource contexts.

However, some challenges are remained. These models require fine-tuning of hyperparameters (dropout rates, and augmentation strategies), making their performance unstable across domains. Their performance is also affected in resources lacking sufficient negative samples due to their reliance on aggregated negatives. Furthermore, these models are largely trained on English corpora, offering limited cross-language generalization without extensive retraining.

IV. Empirical Study

This section presents the evaluation process adopted in our study. Fig. 4 shows the most important stages of this process. We introduce the benchmark datasets commonly used to evaluate the performance of sentence embedding models. Next, we present the evaluation metrics used to measure the performance of NLP tasks such as semantic similarity, clustering, retrieval information, and classification. We present empirical results on the performance of these models. Finally, we discuss the results, identifying the appropriate models for each NLP task.

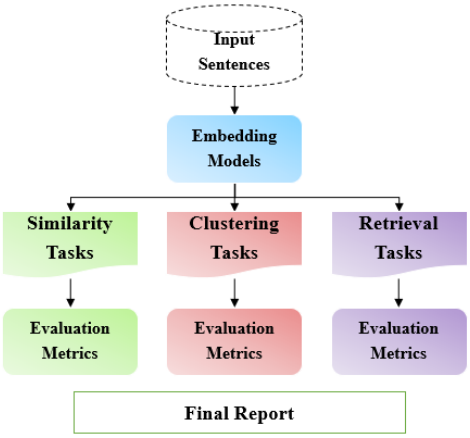

Fig. 4. Process of the evaluation sentence embedding models.

In the first stage, sentences are received by the embedding model to be evaluated, which converts them into numerical vectors in the second stage. These vectors are tested on three main tasks in the third stage. Similarity tasks to measure the degree of meanings similarity, clustering tasks to test the ability to cluster similar sentences, and retrieval tasks to evaluate the effectiveness of sentence retrieval. For each task, appropriate evaluation metrics are adopted to reflect the model's accuracy and performance. Finally, the results are analyzed in a comprehensive report that allows for model comparison.

A. Evaluation Datasets

The evaluation of sentence embedding models relies on datasets that capture task requirements. Semantic text similarity (STS) datasets, such as STS-Benchmark [35] and SICK-R [38], are used because they allow for easy interpretation of the semantic quality scores of embeddings. However, its small size and focus on a specific domain limit its generalizability. Therefore, Natural Language Inference (NLI) datasets, including SNLI [39] and MultiNLI [40], are also commonly used to evaluate embedding techniques. Despite its large size and diversity of sentence pairs, it can lead to embedding without a real understanding of the meaning. Datasets such as MRPC [41] and QQP [42] are used to measure the ability of embeddings to capture paraphrasing and lexical diversity. For retrieval and clustering tasks, datasets such as MS MARCO [43], BEIR [44], and StackExchange [45] are used. Their challenges lie in high variability and data noisy.

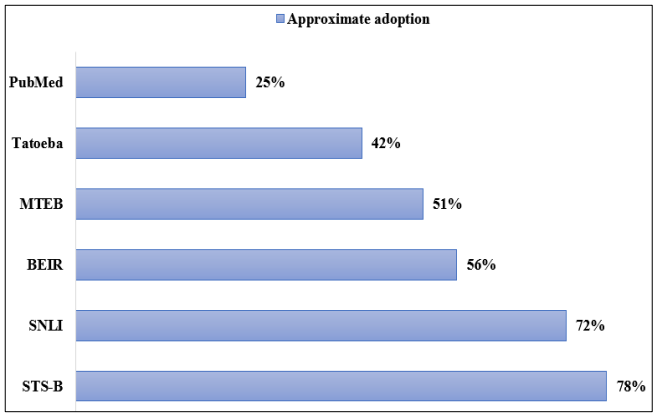

Fig. 5. Approximate adoption rates of training and evaluation datasets in sentence embedding research.

The multilingual datasets, including XNLI [46], Tatoeba [47], and the MTEB [48] multilingual path, are essential for evaluating models across different languages. However, they remain limited in low-resource languages. Fig. 5 shows the approximate adoption rates of datasets in sentence embedding research. Table II compares the most important datasets used for evaluation, in terms of task, size, language, and challenges.

Domain-specific models such as BioBERT, SciBERT instead rely on specialized datasets such as the PubMed biomedical dataset or scientific papers from Semantic Scholar.

B. Evaluation Metrics

To measure the performance of sentence embedding models, the NLP tasks identify the appropriate metric. For semantic textual similarity (STS) tasks, correlation metrics such as Pearson correlation coefficient and Spearman correlation coefficient [49] are used. Cosine similarity [50] is used to measure how closely embeddings converge in vector space, making it suitable for tasks such as clustering and paraphrase detection.

TABLE II. COMPARISON OF DATASETS USED TO TRAIN AND EVALUATE SENTENCE EMBEDDING MODELS

Dataset	Task	Size	Language	Challenges
STS-B	Semantic Textual Similarity	8,600 pairs	English	Small size, domain-limited
SICK-R	Semantic Textual Similarity + Relatedness	10,000 pairs	English	Domain-limited, relatively small
SNLI	Natural Language Inference	570,000 pairs	English	Annotation in English-only
MultiNLI	Natural Language Inference	430,000 pairs	English (multiple genres)	Still English-only, annotation bias
MRPC	Paraphrase Identification	5,800 pairs	English	Small dataset, domain bias (news)
QQP	Paraphrase Identification	400,000 pairs	English	Noise, duplicates, biased toward question style
MS MARCO	Passage Retrieval / Ranking	1M passages, 100k queries	English	Noisy labels, domain-specific (web queries)
StackExchange	Clustering & Retrieval	100k+ QA pairs	English	Annotation ambiguity , Duplicate questions
XNLI	Cross-lingual NLI	750k (15 languages)	Multilingual	Limited low-resource language support
Tatoeba	Cross-lingual Sentence Similarity	1,000+ pairs per language	100+ languages	Uneven quality across languages
MTEB	Multi-task, multi-lingual (STS, retrieval, classification, clustering)	8M examples across 50 datasets	Multilingual (100+ languages)	Complex to run, requires large compute

Also, for clustering tasks, Normalized Mutual Information (NMI), Silhouette score, and Adjusted Rand Index (ARI) [51], can be used to evaluate how well embeddings cluster semantically similar sentences. For classification and natural language inference (NLI) tasks, Accuracy and F1-Score [52] are the most appropriate metrics. For information retrieval and classification tasks, metrics such as mean average precision (MAP) and mean reciprocal rank (MRR) [53] are preferred. Table III presents the most important metrics for evaluating sentence embedding models for each NLP task.

TABLE III. EVALUATION METRICS FOR SENTENCE EMBEDDING MODELS ON NLP TASKS

Metric	STS	Clustering	Paraphrase	Retrieval	NLI	Classification
Cosine Similarity	X	X	X			
Pearson Correlation	X					
Spearman Correlation	X					
Accuracy					X	X
F1-Score			X		X	X
Mean Average Precision (MAP)				X		
Mean Reciprocal Rank (MRR)				X		
Normalized Mutual Information (NMI)		X				
Adjusted Rand Index (ARI)		X				
Silhouette Score		X				

Analysis of sentence embedding research shows that most sentence embedding models prefer evaluation based on semantic textual similarity tasks, while evaluation tasks such as information retrieval and classification are neglected. Fig. 6 shows the approximate adoption rates of evaluation metrics in sentence embedding research.

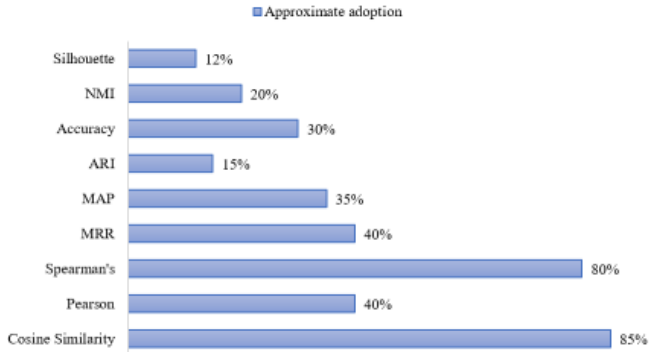

Fig. 6. Approximate adoption rates of evaluation metrics in sentence embedding research.

C. Evaluation protocol

1) Statistical sentence embedding techniques: To evaluate statistical sentence embedding techniques, popular datasets and evaluation metrics were used. The STS-Benchmark dataset and Spearman and Pearson correlations were used to measure the effectiveness of embedding on semantic textual similarity (STS). For sentiment classification tasks, the Stanford Sentiment Treebank (SST-2) dataset [54] and the classification accuracy metric were used. Additionally, the clustering quality was measured on sentence embeddings derived from the STS-B corpus using the silhouette score.

According to the evaluation results presented in Fig. 7, the performance of unweighted averaging was affected by uniform treatment of words, especially in semantic similarity tasks. The

results of TF-IDF weighted averaging also make it more suitable for classification tasks. While the performance of max/min pooling, which oversimplifies the semantic of a sentence, remains inconsistent.

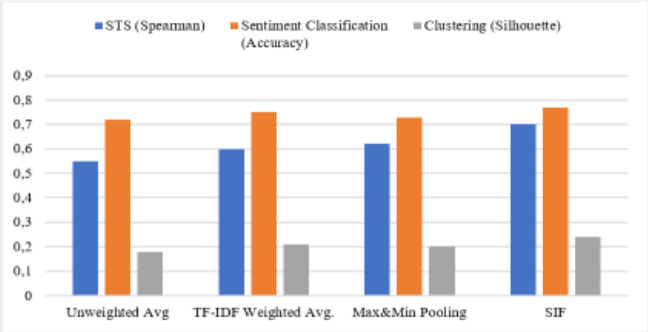

Fig. 7. Performance of statistical sentence embedding techniques across STS, sentiment classification, and clustering.

Conversely, the results demonstrate that SIF outperforms other methods by removing the most common directions in the embedding space, especially in semantic similarity and clustering tasks.

2) Transformer-based models: To ensure a comprehensive evaluation of sentence embedding models, three NLP tasks were adopted. Semantic similarity, clustering, and retrieval. This evaluation requires specific datasets and evaluation metrics for each task. For semantic similarity task, STS-B and Tatoeba were used. STS-B is a widely used for semantic similarity in English, while Tatoeba extends the evaluation to include multilingual sentence pairs. The performance was measured using Pearson and Spearman coefficients. In contrast, for the clustering task, MTEB and STS Clustering were adopted, which test the ability of embeddings to cluster semantically similar texts. Cosine similarity and silhouette score are used to measure the performance. Retrieval task was evaluated using BEIR and MIRACL datasets. BEIR provides a unified framework for information retrieval, while MIRACL focuses on cross-linguistic retrieval performance. MAP and MRR, which reflect the importance of retrieval, were used to measure the performance. Table IV presents the performance of the sentence embedding models on the semantic similarity task evaluation in STS-B and Tatoeba datasets.

The results showed modest performance of BERT and RoberTa, compared to the effectiveness of SBERT and SRoBERTa, on semantic similarity tasks. The multilingual models (mUSE and LabSE) also demonstrated effective performance, particularly on Tatoeba, confirming their performance on cross-linguistic representations. In addition to their computational efficiency, the contrastive learning models (SimCSE and ConSERT) achieved competitive results, making them attractive for light applications. Overall, these results suggest that domain, resource constraints, and language requirements determine the appropriate model.

The performance of sentence embedding models on the clustering task was evaluated using the MTEB Clustering and STS Clustering datasets as a practical benchmark for document clustering. Performance is measured using cosine similarity, which measures the convergence of embeddings within clusters, and the Silhouette score, which reflects cohesion and separation within a cluster. Table V presents the performance results of these models for each dataset.

TABLE IV. PERFORMANCE OF THE TRANSFORMER-BASED MODELS ON THE SEMANTIC SIMILARITY TASK

Model	STS-B		Tatoeba	
	Pearson	Spearman	Pearson	Spearman
BERT	0.63	0.60	0.42	0.40
RoBERTa	0.66	0.65	0.45	0.42
SBERT	**0.85**	**0.81**	0.72	0.70
SRoBERTa	**0.86**	**0.82**	0.71	0.68
BioBERT	0.60	0.55	0.40	0.43
SciBERT	0.70	0.62	0.50	0.46
mUSE	0.76	0.72	**0.85**	**0.82**
LaBSE	0.78	0.75	**0.88**	**0.86**
SimCSE	0.82	0.79	0.78	0.75
ConSERT	0.80	0.76	0.76	0.73

TABLE V. PERFORMANCE OF THE TRANSFORMER-BASED MODELS ON THE CLUSTERING TASK

Model	MTEB Clustering		STS Clustering	
	Cosine	Silhouette	Cosine	Silhouette
BERT	0.38	0.14	0.42	0.15
RoBERTa	0.40	0.16	0.44	0.20
SBERT	0.58	0.27	0.60	0.30
SRoBERTa	0.62	0.31	0.63	0.30
BioBERT	0.45	0.20	0.50	0.20
SciBERT	0.48	0.23	0.54	0.23
mUSE	0.58	0.25	0.60	0.25
LaBSE	**0.62**	**0.29**	0.62	0.28
SimCSE	**0.64**	**0.32**	**0.67**	**0.31**
ConSERT	0.61	0.28	**0.65**	**0.29**

The clustering results demonstrate the superior performance of the SimCSE and ConSERT models, confirming the effectiveness of contrastive learning to produce embeddings that enable the formation of coherent clusters. However, the performance of domain-specific models, such as BioBERT and SciBERT, is affected by the specialization of their training data and thus unsuitable for clustering tasks. In contrast, the multilingual models (mUSE and LabSE) provide competitive results. LaBSE achieves strong performance in multilingual clustering, making it suitable for clustering multilingual documents.

The performance of sentence embedding models on the retrieval task was evaluated using the BEIR and MIRACL benchmarks, which serve as widely adopted frameworks for testing retrieval effectiveness across both monolingual and multilingual settings. Evaluation was carried out using Mean

Average Precision (MAP), which measures the overall ranking quality, and Mean Reciprocal Rank (MRR), which reflects the ability of models to return relevant results at top ranks. Table VI presents the results of all models across both datasets.

TABLE VI. PERFORMANCE OF THE TRANSFORMER-BASED MODELS ON THE RETRIEVAL TASK

Model	BEIR		MIRACL	
	MAP	MRR	MAP	MRR
BERT	0.27	0.30	0.21	0.25
RoBERTa	0.30	0.32	0.23	0.26
SBERT	0.41	0.45	0.35	0.38
SRoBERTa	0.44	0.46	0.40	0.42
BioBERT	0.33	0.37	0.20	0.23
SciBERT	0.35	0.40	0.22	0.24
mUSE	0.32	0.35	**0.42**	**0.46**
LaBSE	0.34	0.39	**0.45**	**0.48**
SimCSE	**0.48**	**0.52**	0.41	0.45
ConSERT	**0.45**	**0.50**	0.38	0.41

Evaluation results showed that baseline models, such as BERT and RoberTa, underperformed on the retrieval task. In contrast, fine-tuning the semantic similarity tasks in SBERT and SRoBERTa improved this performance. Domain-specific models, such as BioBERT and SciBERT, faced challenges in the cross-domain. Contrast learning models (SimCSE and ConSERT) produced embeddings that improved the retrieval task. In contrast, cross-linguistic models (mUSE and LabSE) outperformed only on the MIRACL dataset, where cross-linguistic coverage is essential, but their performance lagged behind the English models in BEIR.

V. DISCUSSION

The evaluation of sentence embedding models focused on quantitative results for natural language processing tasks, including semantic similarity, clustering, and retrieval. While these results provide a numerical comparison of performance, a deeper analysis requires discussion and interpretation that considers practical aspects, such as training costs, model size, generalizability, and cross-linguistic applicability. This section interprets the results and links them to the research questions. This provides a consistent understanding of how each result contributes to answering these questions. This will enable researchers to use embedding models effectively in NLP tasks.

- RQ1: What families of sentence embeddings exist?

Regardless of whether the models are supervised or unsupervised, our results identify two main categories:

Statistical models such as Bag-of-Words, TF-IDF, and SIF, which are fast, interpretable, and easy to implement but fail to capture word order and deeper semantic relations. Statistical models are still inexpensive, but their performance on semantic tasks is low.

Transformer-based models such as BERT, SBERT, SimCSE and LaBSE, which produce contextualized and semantically rich representations. Transformer-based models, on the other hand, achieve significantly higher scores on NLP tasks (semantic similarity, clustering, and retrieval).

- RQ2: What are their strengths and limitations?

Statistical models are simple and easy to interpret, making them suitable for fields with limited resources. However, their representation neglects word order and context, limiting their semantic performance. A comparative analysis showed differences between the four statistical models in simplicity, interpretation, and representation. The unweighted average model provides a straightforward baseline with minimal computational cost, but it often fails to capture semantic relationships. The weighted average TF-IDF model focuses on meaningful words in a sentence, making it more effective for classification tasks. Max-min pooling provides richer embeddings by considering extreme feature values, which can better reflect sentence variance. Finally, the SIF model emerges as a more accurate approach, balancing simplicity and semantic accuracy by reducing the dominance of frequent words and processing meaning at the sentence level.

Transformer-based sentence embedding models also have strengths and weaknesses. Foundational models, such as BERT and RoBERTa, have formed the basis of most models, but they are not optimized sentence level and lack cross-linguistic support. Optimized models, such as SBERT and SRoBERTa, have improved performance on NLP tasks, but they require fine-tuning specific to the task and remain mostly English-focused. While specialized models, such as BioBERT and SciBERT, are effective at embedding biomedical and scientific terms, their performance declines outside of specialized domains. Contrast learning models, such as SimCSE and ConSERT, produce embeddings that achieve high performance on similarity and clustering through simple and efficient contrastive objectives, although they are highly sensitive to training strategies and require large datasets. The main strength of multilingual models, such as mUSE and LabSE, is their ability to exploit common embedding spaces across more than 100 languages, particularly for cross-linguistic retrieval. However, their performance declines in low-resource languages and requires intensive training resources.

Overall, the analysis results indicate that statistical embedding models are more suitable for simple applications, while transformer-based models have become the preferred choice for most NLP tasks, especially those requiring precise semantic analysis. Future research could explore hybrid strategies that combine the efficiency of statistical models with the semantic depth of transformers.

- RQ3: What model is appropriate for each NLP task?

The evaluation results, presented in Tables IV, V, and VI, demonstrate the appropriate sentence embedding model for each NLP task. For semantic similarity tasks, SRoBERTa and SBERT achieved the highest correlation rates on the STS dataset, thus capturing subtle semantic nuances. While LaBSE achieved the highest correlation rates on the Tatoeba dataset, making it more suitable for multilingual scenarios. For clustering tasks, SimCSE outperformed the other models, achieving sentence clustering without being affected by

multilingualism. For retrieval tasks, SimCSE and ConSERT performed best, followed by SRoBERTa, making them most effective for monolingual retrieval tasks. Conversely, LaBSE outperformed on the MIRACL dataset, making it more suitable for multilingual retrieval tasks. Table VII presents the classification of NLP tasks according to appropriate sentence embedding model.

TABLE VII. CLASSIFICATION OF NLP TASKS ACCORDING TO BEST SENTENCE EMBEDDING MODELS

Task	Best Models	Description
Retrieval	SimCSE, ConSERT, SBERT	Contrastive models (SimCSE, ConSERT) are strong baselines, while SBERT remain very competitive.
Clustering	SimCSE, ConSERT	Contrastive models generate well-separated embeddings for clustering.
Semantic Similarity	SBERT, SRoBERTa	SBERT is traditionally strongest for similarity tasks
Multilingual tasks	LaBSE	LaBSE is outperformed for Multilingual tasks

In general, SBERT and SRoBERTa are suitable for semantic similarity, SimCSE and ConSERT are effective for retrieval and clustering, and LaBSE is best suited for multilingual tasks.

VI. CONCLUSION

The study presented in our paper combines a review and empirical assessment of sentence embedding models to address the challenge of selecting the most appropriate model for each NLP task. The study answers three main research questions. First, (RQ1), there are two main categories of embeddings: statistical models such as unweighted average, weighted average TF-IDF, Max-min pooling and SIF, which remain simple and interpretable, and transformer-based models such as BERT, RoBERTa, SBERT, BioBERT, LabSE and SimCSE, which exploit contextual representations. Second, (RQ2), we analyzed their strengths and weaknesses, while lightweight statistical models appear to be incapable of semantic representations. Transformer-based models capture rich contextual information but require significant computational resources and may face challenges in domain transfer. Finally, (RQ4), we determined the appropriate model for three specific NLP tasks. Our empirical study showed that SBERT and SRoBERTa produce embeddings that are well-suited for semantic similarity tasks. In contrast, the embedding generated by the SimCSE model performs best for clustering tasks, while the embeddings generated by the SimCSE, ConSERT, and SRoBERTa models are well-suited for retrieval tasks. In general, the selection of an appropriate model depends on the task requirements, available resources, and language context. For low-resource or interpretable applications, statistical embeddings remain appropriate. For semantically rich applications, despite their computational cost, transformer models are highly efficient. In contrast, multilingual applications can benefit from models such as LaBSE. Future work should explore hybrid strategies that combine efficiency and semantic depth, and expand the scope of evaluations to include multimodal embeddings to meet the growing needs of NLP systems.

REFERENCES

[1] A. Ramesh Kashyap, T.-T. Nguyen, V. Schlegel, S. Winkler, S.-K. Ng, and S. Poria, "A Comprehensive Survey of Sentence Representations: From the BERT Epoch to the CHATGPT Era and Beyond,", 2024, pp. 1738–1751. doi: 10.18653/v1/2024.eacl-long.104.

[2] T. Schopf, D. N. Schneider, F. Matthes, "Efficient Domain Adaptation of Sentence Embeddings Using Adapters," 2023, pp. 1046–1053. doi: 10.26615/978-954-452-092-2_112.

[3] B. Zhang, K. Chang, and C. Li, "Simple Techniques for Enhancing Sentence Embeddings in Generative Language Models," 2024, arXiv. doi: 10.48550/ARXIV.2404.03921.

[4] H. Cao, "Recent advances in text embedding: A Comprehensive Review of Top-Performing Methods on the MTEB Benchmark," 2024, arXiv. doi: 10.48550/ARXIV.2406.01607.

[5] Z. Nie et al., "When Text Embedding Meets Large Language Model: A Comprehensive Survey," 2025, arXiv. doi: 10.48550/ARXIV.2412.09165.

[6] O. Weller et al., "FollowIR: Evaluating and Teaching Information Retrieval Models to Follow Instructions," 2024, arXiv. doi: 10.48550/ARXIV.2403.15246.

[7] H. Su et al., "BRIGHT: A Realistic and Challenging Benchmark for Reasoning-Intensive Retrieval," 2024, arXiv. doi: 10.48550/ARXIV.2407.12883.

[8] O. Khattab et al., "DSPy: Compiling Declarative Language Model Calls into Self-Improving Pipelines," 2023, arXiv. doi: 10.48550/ARXIV.2310.03714.

[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," 2018, arXiv. doi: 10.48550/ARXIV.1810.04805.

[10] A. Joshi et al., "Natural Language Processing for Dialects of a Language: A Survey," 2024, arXiv. doi: 10.48550/ARXIV.2401.05632.

[11] M. Zhang, X. Zhang, X. Zhao, S. Huang, B. Hu, and M. Zhang, "On The Role of Pretrained Language Models in General-Purpose Text Embeddings: A Survey," 2025, arXiv. doi: 10.48550/ARXIV.2507.20783.

[12] E. Oro, F. M. Granata, and M. Ruffolo, "A Comprehensive Evaluation of Embedding Models and LLMs for IR and QA Across English and Italian," BDCC, vol. 9, no. 5, p. 141, May 2025, doi: 10.3390/bdcc9050141.

[13] C. Fettal, L. Labiod, and M. Nadif, "More Discriminative Sentence Embeddings via Semantic Graph Smoothing," 2024, arXiv. doi: 10.48550/ARXIV.2402.12890.

[14] S. Shrestha, N. Reddy, and Z. Li, "ESPN: Memory-Efficient Multi-Vector Information Retrieval," 2023, arXiv. doi: 10.48550/ARXIV.2312.05417.

[15] Y. Wang, J. Zhang, Z. Yang, B. Wang, J. Jin, and Y. Liu, "Improving extractive summarization with semantic enhancement through topic-injection based BERT model," Information Processing & Management, vol. 61, no. 3, p. 103677, May 2024, doi: 10.1016/j.ipm.2024.103677.

[16] M. A. Mersha, M. G. yigezu, and J. Kalita, "Semantic-Driven Topic Modeling Using Transformer-Based Embeddings and Clustering Algorithms," 2024, doi: 10.48550/ARXIV.2410.00134.

[17] C. D. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval, 1st ed. Cambridge University Press, 2008. doi: 10.1017/CBO9780511809071.

[18] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," 2013, arXiv. doi: 10.48550/ARXIV.1301.3781.

[19] J. Pennington, R. Socher, and C. Manning, "Glove: Global Vectors for Word Representation,", 2014, pp. 1532–1543. doi: 10.3115/v1/D14-1162.

[20] Q. Chen, Z.-H. Ling, and X. Zhu, "Enhancing Sentence Embedding with Generalized Pooling," 2018, arXiv. doi: 10.48550/ARXIV.1806.09828.

[21] Q. V. Le and T. Mikolov, "Distributed Representations of Sentences and Documents," 2014, arXiv. doi: 10.48550/ARXIV.1405.4053.

[22] S. Arora, Y. Liang, and T. Ma, "A SIMPLE BUT TOUGH-TO-BEAT BASELINE FOR SEN- TENCE EMBEDDINGS," 2017.

[23] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, and A. Bordes, "Supervised Learning of Universal Sentence Representations from Natural Language Inference Data,", 2017, pp. 670–680. doi: 10.18653/v1/D17-1070.

[24] D. Cer et al., "Universal Sentence Encoder," 2018, arXiv. doi: 10.48550/ARXIV.1803.11175.

[25] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," 2019, arXiv. doi: 10.48550/ARXIV.1908.10084.

[26] T. Gao, X. Yao, and D. Chen, "SimCSE: Simple Contrastive Learning of Sentence Embeddings," 2021, arXiv. doi: 10.48550/ARXIV.2104.08821.

[27] R. Kiros et al., "Skip-Thought Vectors," 2015, arXiv. doi: 10.48550/ARXIV.1506.06726.

[28] F. Hill, K. Cho, and A. Korhonen, "Learning Distributed Representations of Sentences from Unlabelled Data,", 2016, pp. 1367–1377. doi: 10.18653/v1/N16-1162.

[29] A. Conneau and D. Kiela, "SentEval: An Evaluation Toolkit for Universal Sentence Representations," 2018, arXiv. doi: 10.48550/ARXIV.1803.05449.

[30] N. Thakur, N. Reimers, A. Rücklé, A. Srivastava, and I. Gurevych, "BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models," 2021, arXiv. doi: 10.48550/ARXIV.2104.08663.

[31] Y. Liu et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," 2019, arXiv. doi: 10.48550/ARXIV.1907.11692.

[32] Y. Yang et al., "Multilingual Universal Sentence Encoder for Semantic Retrieval," 2019, arXiv. doi: 10.48550/ARXIV.1907.04307.

[33] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, and W. Wang, "Language-agnostic BERT Sentence Embedding," 2020, arXiv. doi: 10.48550/ARXIV.2007.01852.

[34] M. Artetxe and H. Schwenk, "Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond," Transactions of the Association for Computational Linguistics, vol. 7, pp. 597–610, Nov. 2019, doi: 10.1162/tacl_a_00288.

[35] D. Cer, M. Diab, E. Agirre, I. Lopez-Gazpio, and L. Specia, "SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Crosslingual Focused Evaluation," i, 2017, pp. 1–14. doi: 10.18653/v1/S17-2001.

[36] Y. Yan, R. Li, S. Wang, F. Zhang, W. Wu, and W. Xu, "ConSERT: A Contrastive Framework for Self-Supervised Sentence Representation Transfer," 2021, arXiv. doi: 10.48550/ARXIV.2105.11741.

[37] A. van den Oord, Y. Li, and O. Vinyals, "Representation Learning with Contrastive Predictive Coding," 2018, arXiv. doi: 10.48550/ARXIV.1807.03748.

[38] M. Marelli, S. Menini, M. Baroni, L. Bentivogli, R. Bernardi, and R. Zamparelli, "A SICK cure for the evaluation of compositional distributional semantic models," vol. Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), pp. 216–223, 2014.

[39] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning, "A large annotated corpus for learning natural language inference," 2015, arXiv. doi: 10.48550/ARXIV.1508.05326.

[40] A. Williams, N. Nangia, and S. Bowman, "A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference,", 2018, pp. 1112–1122. doi: 10.18653/v1/N18-1101.

[41] W. B. Dolan and C. Brockett, "Automatically Constructing a Corpus of Sentential Paraphrases," vol. Proceedings of the Third International Workshop on Paraphrasing (IWP2005), 2005.

[42] L. Sharma, L. Graesser, N. Nangia, and U. Evci, "Natural Language Understanding with the Quora Question Pairs Dataset," 2019, arXiv. doi: 10.48550/ARXIV.1907.01041.

[43] P. Bajaj et al., "MS MARCO: A Human Generated MAchine Reading COmprehension Dataset," 2016, arXiv. doi: 10.48550/ARXIV.1611.09268.

[44] N. Thakur, N. Reimers, A. Rücklé, A. Srivastava, and I. Gurevych, "BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models," 2021, arXiv. doi: 10.48550/ARXIV.2104.08663.

[45] M. Hazoom, V. Malik, and B. Bogin, "Text-to-SQL in the Wild: A Naturally-Occurring Dataset Based on Stack Exchange Data," 2021, arXiv. doi: 10.48550/ARXIV.2106.05006.

[46] A. Conneau et al., "XNLI: Evaluating Cross-lingual Sentence Representations,", 2018, pp. 2475–2485. doi: 10.18653/v1/D18-1269.

[47] J. Tiedemann, "The Tatoeba Translation Challenge -- Realistic Data Sets for Low Resource and Multilingual MT," 2020, arXiv. doi: 10.48550/ARXIV.2010.06354.

[48] K. Enevoldsen et al., "MMTEB: Massive Multilingual Text Embedding Benchmark," 2025, arXiv. doi: 10.48550/ARXIV.2502.13595.

[49] M. Gao, X. Hu, L. Lin, and X. Wan, "Analyzing and Evaluating Correlation Measures in NLG Meta-Evaluation,", 2025, pp. 2199–2222. doi: 10.18653/v1/2025.naacl-long.111.

[50] S. Sahoo and J. Maiti, "Variance-Adjusted Cosine Distance as Similarity Metric," 2025, arXiv. doi: 10.48550/ARXIV.2502.02233.

[51] M. McCrory and S. A. Thomas, "Cluster Metric Sensitivity to Irrelevant Features," 2024, arXiv. doi: 10.48550/ARXIV.2402.12008.

[52] C. Miller, T. Portlock, D. M. Nyaga, and J. M. O'Sullivan, "A review of model evaluation metrics for machine learning in genetics and genomics," Front. Bioinform., vol. 4, p. 1457619, Sept. 2024, doi: 10.3389/fbinf.2024.1457619.

[53] O. Rainio, J. Teuho, and R. Klén, "Evaluation metrics and statistical tests for machine learning," Sci Rep, vol. 14, no. 1, p. 6086, Mar. 2024, doi: 10.1038/s41598-024-56706-x.

[54] R. Socher et al., "Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank,", 2013, pp. 1631–1642. doi: 10.18653/v1/D13-1170.