Applied Computer Systems
ISSN 2255-8691 (online)
ISSN 2255-8683 (print)
2025, vol. 30, no. 1, pp. 34–39
https://doi.org/10.2478/acss-2025-0004
https://content.sciendo.com

Comparison of Language Models for English-Latvian Semantic Search

Artem Kucheravy[1], Gints Jēkabsons[2*]
[1,2]*Institute of Applied Computer Systems, Riga Technical University, Riga, Latvia*

***Abstract* – In this study, ten language models are explored and compared in an English-Latvian semantic information retrieval setting, where the indexed collection of documents is written in English while the query documents are written in Latvian. Currently, no similar research has been done regarding the Latvian language. A dataset of 77736 pairs of articles from Latvian and English Wikipedia was created, transformed into embedding vectors, and used for retrieval experiments with brute force search, Hierarchical Navigable Small World method, and Inverted File Indexing method. The LaBSE language model achieved the best performance for short texts and a version of Sentence-BERT and E5-large for long texts.**

***Keywords* – Embeddings, language models, semantic search, sentence-transformers.**

I. Introduction

Semantic search is the next step in information retrieval, which abstracts from keywords and focuses on text meaning instead. Text similarity is found based on text semantics.

Nowadays, to capture text semantics, transformer models are primarily used. This currently dominant architecture was initially proposed in [1]. The self-attention mechanism is the core of transformers; it allows models to understand contextual importance in relationships between words in a text.

Multilingual semantic search is an information retrieval process where the query is written in one language, and document collection is written in another language. Multilingual semantic search is relevant today as most of the available information on the web is in English, while at the same time, most people's native language is not English. This unequal knowledge distribution is an unnecessary complication.

Research by various authors, e.g., [2], [3], [4], [5], shows strong performance of transformers, in general outperforming other types of models.

This study is motivated by the small amount of work that would focus on multilingual semantic search involving low-resource languages, especially English-Latvian language pair.

This study aims to evaluate and compare language models in English-Latvian semantic search.

The models chosen for evaluation were trained to capture the meaning of sentences and paragraphs in a multilingual setting. Models that do not support both the English and Latvian languages were omitted.

For the experiments in this study, a new dataset was created out of publicly available Wikipedia dumps. The dataset consists of Latvian Wikipedia pages paired with their corresponding English analogues. Secondly, each article was embedded into vector space using the language models. After that, the search was performed where the English articles served as the collection to search in while the Latvian articles served as queries. Performance was measured using the top-k accuracy score.

The rest of the paper is organised as follows: Section II overviews similar research and results that have already been achieved. Section III overviews language models chosen for this evaluation. Section IV describes the experimental setup. Section V presents and analyses the results. Finally, Section VI concludes the paper.

II. Related Work

Some research with similar goals has already been conducted in previous studies. For example, the authors in [5] trained two models on the same dataset. The first model implemented transformer architecture, while the second implemented Convolution Neural Network architecture. Evaluation of these models in Bitext Retrieval showed that the first model performed better except in the English–Chinese language pair. The highest achieved result with the Precision@1 score was 88.9 in the English-Russian language pair. Other results were 86.1 in the English-Spanish pair, 83.3 in the English-French pair, and 78.8 in English-Chinese.

The author in [6] compared BERT, LASER, and the classical TF-IDF method in English semantic search. BERT outperformed both other methods, reaching a recall of 0.55, while LASER and TF-IDF obtained 0.44 and 0.37, respectively.

In [7], authors compared transformer-based models and similarity-specialised sentence encoders between five languages on document-level and sentence-level information retrieval. Results showed that the best model for document-level was DistilmBERT, with a Mean Average Precision (MAP) score of 0.302. However, in sentence-level information

* Corresponding author's e-mail: gints.jekabsons@rtu.lv
Article received 2024-09-30; accepted 2025-01-02

retrieval, LASER outperformed everything else, reaching a MAP score of 0.974.

Research [8] evaluated multiple models on document and sentence-level cross-lingual information retrieval. DistilmBERT model achieved the best result in document-level retrieval with a MAP score of 0.313 in the English-Russian language pair. In contrast, LASER showed the most robust performance in sentence-level retrieval, achieving a MAP score of 0.976 in the English-Italian language pair. LaBSE performed very well as well, achieving a score of 0.972.

III. The Evaluated Models

Ten models were selected for the experiments. All of the models are multilingual and support Latvian and English languages and can create meaningful sentence vectors instead of just performing average pooling. The LASER model is based on the long short-term memory architecture, while the rest are based on transformer architecture.

Language Agnostic Sentence Representation (LASER2). The first version was presented in 2019 [9], and an improved version in 2022 [10]. In this study, the second version is used. It is based on Bidirectional Long Short-Term Memory encoder with a shared Byte-Pair Encoding vocabulary for all languages [9]. Model is available through [11], has 250 million parameters [10], yields 1024 dimensional vectors [9], and does not have an input token limit. The model supports over 200 languages.

Sentence-level multimodal and language-Agnostic Representations (SONAR). It was introduced in 2023 [12]. SONAR is based on a transformer encoder-decoder architecture, initialized with pre-trained machine translation model weights [12]. The model is available through [13]. It yields 1024 dimensional vectors and has an input token limit equal to 512. The model supports 200 languages.

Language-agnostic BERT Sentence Embedding (LaBSE). This sentence-transformer model was introduced in 2020 [14]. It is available through [15]. It has 471 million parameters [16], yields 768-dimensional vectors, and has an input limit of 256 tokens [15]. The model supports 110 languages.

The following seven models are sentence-transformers. The first four were introduced as a part of [17] in 2019, the authors used multilingual knowledge distillation (teacher-student approach) by making smaller model imitate bigger model embeddings. The last four models were introduced in [18] in 2024; the authors initialised them from MiniLM and did further training.

Sentence-BERT paraphrase-multilingual-MiniLM-L12-v2 (henceforth referred to as paraphrase-MiniLM-L12). It is available through [19]. It has 118 million parameters [16], yields 384-dimensional vectors, and has an input limit of 128 tokens [19]. The model supports 50 languages.

Sentence-BERT paraphrase-multilingual-mpnet-base-v2 (henceforth referred to as paraphrase-mpnet). It is available through [20]. It has 278 million parameters [16], yields 768-dimensional vectors, and has an input limit of 128 tokens [20]. The model supports 50 languages.

Sentence-BERT distiluse-base-multilingual-cased (henceforth referred to as distiluse-base). It is available through [21]. It has 135 million parameters, yields 512-dimensional vectors, and has an input limit equal to 128 tokens [21]. The model supports 50 languages.

Sentence-BERT distiluse-base-multilingual-cased-v2 (henceforth referred to as distiluse-base-v2). It is available through [22]. It has 135 million parameters [16], yields 512-dimensional vectors, and has an input limit equal to 128 tokens [22]. The model supports 50 languages.

E5 multilingual-e5-small (henceforth referred to as E5-small). It is available through [23]. It has 118 million parameters [16], yields 384-dimensional vectors, and has an input limit equal to 512 [23]. The model supports 94 languages.

E5 multilingual-e5-base (henceforth referred to as E5-base). It is available through [24]. It has 278 million parameters [16], yields 768-dimensional vectors, and has an input token limit equal to 512 [24]. The model supports 94 languages.

E5 multilingual-e5-large (henceforth referred to as E5-large). It is available through [25]. It has 560 million parameters [16], yields 1024-dimensional vectors, and has an input token limit equal to 512 [25]. The model supports 94 languages.

Sentence-BERT-based models have been evaluated in semantic similarity using Spearman rank correlation between cosine similarity of sentence representations and gold standard [17]. Paraphrase models achieved an average score of 83.7 in seven language pairs and distiluse models achieved 76.0.

E5 models have been evaluated in multilingual retrieval [18]. The used metric was Recall@100. The average results per 16 languages for E5-small, E5-base, and E5-large were 92.4, 93.1, and 94.3, respectively.

IV. Experimental Setup

A. Description of the Datasets

In order to conduct experiments evaluating the language models in semantic search, a new dataset was created from Wikipedia CirrusSearch dumps [26] for English and Latvian Wikipedia (version 20240401). The dumps had already been cleaned from tags and metadata, so no additional text cleaning was necessary. All articles with missing titles, introductions, or full texts were discarded.

Next, the MediaWiki API was used to find direct correspondence between the English and Latvian articles and discard any articles available in only one of the languages. The result is a dataset of 77 736 article pairs.

TABLE I
Average Length in Words of Articles by Text Type

Language	Text type	Mean length	Median length	Standard deviation
Latvian	Title	2	2	1
Latvian	Introduction	74	60	59
Latvian	Full text	296	159	546
English	Title	2	2	1
English	Introduction	154	111	134
English	Full text	3243	1547	4348

Finally, two more datasets were created from the first dataset. While each article in the first dataset contains a title, introduction text, and full text, in the second dataset, full text is removed, and in the third dataset, introduction text is removed as well, leaving only titles.

TABLE II

TOP-*K* ACCURACY SCORES (PERCENTAGES) FOR *FLATL2* INDEX WITH DIFFERENT LANGUAGE MODELS, TEXT TYPES, AND *K* VALUES

NUMBERS IN BOLD ARE THE BEST SCORES IN A COLUMN

Text type / Language model	Title				Introduction				Full Article			
	***k* = 1**	***k* = 5**	***k* = 10**	***k* = 20**	***k* = 1**	***k* = 5**	***k* = 10**	***k* = 20**	***k* = 1**	***k* = 5**	***k* = 10**	***k* = 20**
distiluse-base	45.44	55.91	59.62	63.25	71.46	85.41	88.83	91.54	70.67	84.57	88.14	90.93
distiluse-base-v2	45.44	55.91	59.62	63.25	71.46	85.41	88.83	91.54	70.67	84.57	88.14	90.93
E5-small	53.25	65.33	69.07	72.37	80.21	90.71	93.00	94.73	61.49	75.78	79.86	83.60
E5-base	55.32	66.54	69.98	73.27	81.75	92.36	94.48	96.05	63.40	77.82	81.92	85.70
E5-large	58.25	69.49	72.75	76.01	**88.75**	**96.61**	**97.83**	**98.62**	83.30	**93.39**	**95.32**	**96.74**
LaBSE	**69.53**	**79.32**	**82.24**	**84.72**	73.22	86.25	89.61	92.20	73.30	86.45	89.69	92.29
LASER2	52.38	61.03	63.75	66.25	35.59	45.54	49.65	53.60	13.31	17.81	19.83	22.07
paraphrase-MiniLM-L12	37.55	46.37	49.77	52.97	74.70	87.19	90.12	92.45	73.66	85.97	89.03	91.50
paraphrase-mpnet	47.73	57.68	61.34	64.55	84.27	93.59	95.45	96.80	**83.35**	92.96	94.98	96.44
SONAR	69.04	76.85	78.99	80.83	59.26	72.39	76.76	80.34	59.47	74.28	78.98	82.98

TABLE III

TOP-*K* ACCURACY SCORES (PERCENTAGES) FOR *HNSWFLAT* INDEX WITH DIFFERENT LANGUAGE MODELS, TEXT TYPES, AND *K* VALUES

NUMBERS IN BOLD ARE THE BEST SCORES IN A COLUMN

Text type / Language model	Title				Introduction				Full Article			
	***k* = 1**	***k* = 5**	***k* = 10**	***k* = 20**	***k* = 1**	***k* = 5**	***k* = 10**	***k* = 20**	***k* = 1**	***k* = 5**	***k* = 10**	***k* = 20**
distiluse-base	44.23	53.87	57.15	60.30	69.61	82.84	86.00	88.42	68.98	82.26	85.58	88.14
distiluse-base-v2	44.23	53.86	57.15	60.30	69.58	82.84	85.98	88.39	68.88	82.20	85.51	88.05
E5-small	46.15	54.54	56.79	58.47	65.15	73.21	74.82	75.98	36.39	42.73	44.28	45.48
E5-base	48.52	56.37	58.42	60.14	67.22	75.25	76.74	77.77	37.87	44.88	46.51	47.83
E5-large	48.14	54.82	56.43	57.73	81.06	87.93	88.94	89.56	66.13	73.53	74.88	75.75
LaBSE	**68.10**	**76.91**	**79.29**	**81.22**	69.34	80.80	83.49	85.44	71.41	83.76	86.62	88.88
LASER2	50.05	57.04	58.94	60.55	30.17	37.80	40.89	43.71	11.38	14.94	16.45	18.02
paraphrase-MiniLM-L12	36.52	44.58	47.60	50.33	73.72	85.77	88.54	90.70	72.90	84.82	87.72	90.02
paraphrase-mpnet	46.58	55.62	58.81	61.50	**83.52**	**92.57**	**94.35**	**95.60**	**82.65**	**91.96**	**93.91**	**95.26**
SONAR	64.96	71.06	72.56	73.81	50.22	59.95	63.03	65.27	52.24	64.25	67.80	70.59

TABLE IV

TOP-*K* ACCURACY SCORES (PERCENTAGES) FOR *IVFFLAT* INDEX WITH DIFFERENT LANGUAGE MODELS, TEXT TYPES, AND *K* VALUES

NUMBERS IN BOLD ARE THE BEST SCORES IN A COLUMN

Text type / Language model	Title				Introduction				Full Article			
	***k* = 1**	***k* = 5**	***k* = 10**	***k* = 20**	***k* = 1**	***k* = 5**	***k* = 10**	***k* = 20**	***k* = 1**	***k* = 5**	***k* = 10**	***k* = 20**
distiluse-base	44.47	53.91	57.11	60.19	69.80	83.07	86.28	88.80	69.25	82.53	85.88	88.51
distiluse-base-v2	44.47	53.91	57.11	60.19	69.80	83.07	86.28	88.80	69.25	82.53	85.88	88.51
E5-small	45.92	53.39	55.35	57.04	75.34	84.42	86.26	87.68	53.34	64.57	67.62	70.30
E5-base	49.50	56.90	59.00	60.90	77.29	86.50	88.26	89.57	58.17	70.67	74.11	77.17
E5-large	50.89	57.83	59.52	61.19	**86.12**	**93.18**	**94.23**	94.91	80.04	89.16	90.87	92.13
LaBSE	**67.28**	**74.90**	**76.85**	**78.29**	68.63	79.50	82.04	83.92	72.08	84.67	87.72	90.19
LASER2	48.92	55.57	57.42	58.96	34.11	42.97	46.51	49.77	11.99	15.52	17.06	18.71
paraphrase-MiniLM-L12	36.69	44.60	47.43	50.09	73.53	85.47	88.23	90.38	72.77	84.70	87.60	89.96
paraphrase-mpnet	46.54	55.22	58.12	60.60	83.35	92.31	94.08	**95.37**	**82.57**	**91.82**	**93.75**	**95.16**
SONAR	64.41	70.27	71.76	72.94	54.86	65.48	68.86	71.42	56.84	70.57	74.86	78.48

As can be seen in Table I, introductions and full texts of Latvian articles, on average, are much shorter than their English counterparts. This might introduce an additional challenge for the search as with such differences in text length the language models might produce embeddings that are too dissimilar.

It should be noted that in the experiments, when a model with an input token limit received a text that exceeded this limit, the input text was truncated to the limit.

The created dataset is available for download at https://github.com/TheHappyLemon/comparison-of-LMs-in-semantic-IR

B. Experiments and Evaluation Criteria

As a vector database, we used the Facebook AI Similarity Search (FAISS) [27], [28] library. Since FAISS provides several vector index types, experiments were done with FlatL2, IVFFlat, and HNSWFlat. These indices were chosen because they represent the "Flat" family, which stores vectors without compression.

The FlatL2 index uses a simple brute force approach. When searching for nearest neighbours, it calculates Euclidean distance from a query vector to every vector in the index and returns top-*k* results. Retrieval of nearest neighbours is guaranteed, but the process is slow.

HNSWFlat index implements Hierarchical Navigable Small World [29] architecture. The index is a graph where each vector stores links to a limited number of neighbours. During the search, the graph is explored towards the most similar vectors. This index type is able to increase the search speed even by an order of magnitude while sacrificing some recall [30].

IVFFlat is an inverted file index. This index groups vectors into partitions (Voronoi cells). Changing the number of explored cells affects query time and accuracy. This study uses HNSWFlat and IVFFlat indices with their default parameters.

To evaluate a model, first, it is used to turn all 77 736 English articles of a dataset into multidimensional vectors stored in a vector index. Then, the same model is used to embed Latvian articles one by one, and *k* most similar (according to the index) English articles are found in the index. *k* values used were 1, 5, 10, and 20. If among the found English articles, there is the one article directly corresponding to the Latvian article, the search is considered successful.

In total, each of the ten models mentioned in Section III was evaluated in 48 such scenarios. Each scenario uses one of the three datasets (see Section IV.A), one of the four index types, and one of the four *k* values. The evaluation uses the top-*k* accuracy score, which measures the percentage of times the desired item was in the top-*k* results.

V. Results

The evaluation results are presented in Table II for the FlatL2 index, Table III for the HNSWFlat index, and Table IV for the IVFFlat index.

A. Text Type Impact on the Results

First, it should be noted that about 11 % of the articles share the same title across the two languages, and these articles were perfectly found by all models.

The best performance with titles was achieved by LaBSE. However, SONAR fell short only by a few percent. The most probable reason for the best performance of these two models is that both were trained on sentence-level translated parallel text corpora [12], [14], and many of the titles in the dataset are either direct translations or the same exact words.

All models, except LASER2 and SONAR, performed better when the data contained introduction texts. While for LaBSE, the improvement is just by a few percent, for the rest of the models, it is dramatic – about 30 and even 40 percent. Generally, the improvement was expected as introductions of Wikipedia articles tend to capture the essence of the whole article. The best results were achieved by E5-large (best model with FlatL2 and IVFFlat index) and paraphrase-mpnet (best model with HNSWFlat index). LASER2's drop of performance (which keeps dropping for the full-text dataset as well) shows that it is either not able to compute good embeddings for longer texts or yields very different embeddings for texts of different lengths (recall that the texts in Latvian are much shorter than texts in English). At the same time, it is the only model without an input sequence limit – it always embeds the whole available text instead of a fixed amount of first tokens, as is done by other models.

Full text of Wikipedia articles is usually much longer than introductions and the difference between full article lengths in Latvian and English is even greater than with introductions. This may create additional noise for language models. As a result, for all models, except LaBSE and SONAR, the performance got worse. It should be mentioned that Sentence-BERT-based models distiluse-base, distiluse-base-v2, paraphrase-MiniLM-L12, and paraphrase-mpnet results were worse by only less than one percent, which is explainable by their input limit equal to 128 tokens, so for them, the input text did not change much when moving from introductions to full texts. The best results were achieved by E5-large (with FlatL2 index) and paraphrase-mpnet (with HNSWFlat and IVFFlat indices). The lighter versions of the E5 lost a lot of accuracy, while the most notable loss was for LASER2.

B. Index Type Impact on the Results

The brute force algorithm in the FlatL2 index type helped secure the best performance in all cases, which was expected. With HNSWFlat and IVFFlat indices, the performances of all models naturally decreased. Overall, the models that gave the best performance were LaBSE for titles and paraphrase-mpnet (with the two faster indices) and E5-large (with brute force), both for introductions and full articles.

An obvious drawback of brute force search with FlatL2 is search efficiency; however, for all index types, efficiency also depends on embeddings dimensionality. Table V shows performed searches per second for each index type and embeddings dimensionality. The measurements were done using one core on a PC with an Intel i7-12700 CPU. In Table V, it can be seen that increasing embeddings dimensionality, the efficiency of IVFFlat drops much faster than that of HNSWFlat. Acknowledging the fact that the accuracies of best models were very similar between the two indices, one can conclude that

HNSWFlat has a clear advantage. However, it should be noted that the developers of FAISS have concluded that, due to higher memory requirements of HNSW, for datasets with more than 1 million vectors the IVF-type indices become the only option [27].

TABLE V
AVERAGE SEARCHES PER SECOND FOR EACH INDEX TYPE AND EMBEDDINGS DIMENSIONALITY

Index	Embeddings dimensionality			
	384	**512**	**768**	**1024**
FlatL2	125.2	96.0	65.9	52.8
HNSWFlat	669.9	659.9	634.7	583.6
IVFFlat	677.7	587.2	473.4	189.5

VI. CONCLUSION

Ten different language models were evaluated and compared. For short texts (article titles), the best performance was achieved by LaBSE (with SONAR being the second best). In contrast, for long texts (article introductions or full texts), the best performance was achieved by E5-large and paraphrase-mpnet, the first being more effective with the FlatL2 index and the second being more effective with the faster indices. At the same time, paraphrase-mpnet has two times fewer parameters than E5-large (278 million vs. 560 million) making it more practical. E5-base, having the same number of parameters, could not demonstrate results anywhere near the ones by paraphrase_mpnet. The overall recommendation for English-Latvian semantic search would be to use LaBSE for short texts and paraphrase-mpnet for long texts, together with HNSW indexing.

As a possible future research direction, it might be worthwhile to study the dependence of model effectiveness on the length of the given text.

REFERENCES

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, pp. 5998–6008, 2017. https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[2] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Jul. 2020, pp. 8440–8451. https://doi.org/10.18653/v1/2020.acl-main.747

[3] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, and P. Barham, "PaLM: Scaling language modeling with pathways," *Journal of Machine Learning Research*, vol. 24, pp. 1–113, 2023. https://www.jmlr.org/papers/volume24/22-1144/22-1144.pdf

[4] N. Muennighoff, "SGPT: GPT sentence embeddings for semantic search," *arXiv preprint 2202.08904*, Aug. 2022. https://doi.org/10.48550/arXiv.2202.08904

[5] Y. Yang, D. Cer, A. Ahmad, M. Guo, J. Law, N. Constant, G.H. Abrego, S. Yuan, C. Tar, Y.-H. Sung, B. Strope, and R. Kurzweil, "Multilingual universal sentence encoder for semantic retrieval," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, Jul. 2020, pp. 87–94. https://doi.org/10.18653/v1/2020.acl-demos.12

[6] B. Bijin, "A local and intelligent web information retrieval system," M.S. thesis, University of Alberta, Alberta, AB, Canada, 2021. https://doi.org/10.7939/r3-eb5m-q238

[7] R. Litschko, I. Vulić, S.P. Ponzetto, and G. Glavaš, "Evaluating multilingual text encoders for unsupervised cross-lingual retrieval," in *Advances in Information Retrieval: 43rd European Conference on IR Research, ECIR 2021*, *Lecture Notes in Computer Science*, vol. 12656, Springer Verlag, Berlin, Heidelberg, Mar. 2021, pp. 342–358. https://doi.org/10.1007/978-3-030-72113-8_23

[8] R. Litschko, I. Vulić, S.P. Ponzetto, and G. Glavaš, "On cross-lingual retrieval with multilingual text encoders," *Information Retrieval Journal*, vol. 25, pp. 149–183, Mar. 2022. https://doi.org/10.1007/s10791-022-09406-x

[9] 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, Sep. 2019. https://doi.org/10.1162/tacl_a_00288

[10] K. Heffernan, O. Çelebi, and H. Schwenk, "Bitext mining using distilled sentence representations for low-resource languages," in *Findings of the Association for Computational Linguistics: EMNLP 2022*, Abu Dhabi, United Arab Emirates, Dec. 2022, pp. 2101–2112. https://doi.org/10.18653/v1/2022.findings-emnlp.154

[11] Meta Research, "Language-agnostic sentence representations," 2018. [Online]. Available: https://github.com/facebookresearch/LASER. Accessed on: Jul. 18, 2024.

[12] P.-A. Duquenne, H. Schwenk, and B. Sagot, "SONAR: Sentence-level multimodal and language-agnostic representations," *arXiv preprint 2308.11466*, Aug. 2023. https://doi.org/10.48550/arXiv.2308.1146

[13] AI at Meta, "SONAR", 2023. [Online]. Available: https://huggingface.co/facebook/SONAR. Accessed on: Jul. 18, 2024.

[14] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, and W. Wang, "Language-agnostic BERT sentence embedding," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, Dublin, Ireland, May 2022, pp. 878–891. https://doi.org/10.18653/v1/2022.acl-long.62

[15] Sentence Transformers, "LaBSE", 2019. [Online]. Available: https://huggingface.co/sentence-transformers/LaBSE. Accessed on: Jul. 24, 2024.

[16] N. Muennighoff, N. Tazi, L. Magne, and N. Reimers, "MTEB: Massive text embedding benchmark," in *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, Dubrovnik, Croatia, May 2023, pp. 2014–2037. https://doi.org/10.18653/v1/2023.eacl-main.148

[17] N. Reimers and I. Gurevych, "Making monolingual sentence embeddings multilingual using knowledge distillation," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, Nov. 2020, pp. 3982–3992. https://doi.org/10.18653/v1/2020.emnlp-main.365

[18] L. Wang, N. Yang, X. Huang, L. Yang, R. Majumder, and F. Wei, "Multilingual E5 text embeddings: A technical report," *arXiv preprint 2402.05672*, 2024. https://arxiv.org/pdf/2406.01607

[19] Sentence Transformers, "paraphrase-multilingual-MiniLM-L12-v2," 2019. [Online]. Available: https://huggingface.co/sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2. Accessed on: Jul. 24, 2024.

[20] Sentence Transformers, "paraphrase-multilingual-mpnet-base-v2," 2019. [Online]. Available: https://huggingface.co/sentence-transformers/paraphrase-multilingual-mpnet-base-v2. Accessed on: Jul. 24, 2024.

[21] Sentence Transformers, "distiluse-base-multilingual-cased," 2019. [Online]. Available: https://huggingface.co/sentence-transformers/distiluse-base-multilingual-cased. Accessed on: Jul. 24, 2024.

[22] Sentence Transformers, "distiluse-base-multilingual-cased-v2," 2019. [Online]. Available: https://huggingface.co/sentence-transformers/distiluse-base-multilingual-cased-v2. Accessed on: Jul. 24, 2024.

[23] L. Wang, "Multilingual-E5-small," 2024. [Online]. Available: https://huggingface.co/intfloat/multilingual-e5-small. Accessed on: Jul. 25, 2024.

[24] L. Wang, "Multilingual-E5-base," 2024. [Online]. Available: https://huggingface.co/intfloat/multilingual-e5-base. Accessed on: Jul. 25, 2024.

[25] L. Wang, "Multilingual-E5-large," 2024. [Online]. Available: https://huggingface.co/intfloat/multilingual-e5-large. Accessed on: Jul. 23, 2024.

[26] Wikimedia, "CirrusSearch dumps," 2024. [Online]. Available: https://dumps.wikimedia.org/other/cirrussearch/. Accessed on: May 18, 2024.
[27] M. Douze, A. Guzhva, C. Deng, J. Johnson, G. Szilvasy, P.-E. Mazaré, M. Lomeli, L. Hosseini, and H. Jégou, "The Faiss library," *arXiv preprint 2401.08281*, 2024. https://arxiv.org/pdf/2401.08281
[28] J. Johnson, M. Douze, and H. Jégou, "Billion-scale similarity search with GPUs," *IEEE Transactions on Big Data*, vol. 7, no. 3, pp. 535–547, June 2019. https://doi.org/10.1109/TBDATA.2019.2921572
[29] Y.A. Malkov and D.A. Yashunin, "Efficient and robust approximate nearest neighbor search using Hierarchical Navigable Small World Graphs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 4, pp. 824–836, Apr. 2020. https://doi.org/10.1109/TPAMI.2018.2889473
[30] J. Briggs, "Faiss: The missing manual," in *Pinecone*, 2023. [Online]. Available: https://www.pinecone.io/learn/series/faiss/. Accessed on: Aug. 18, 2024.

Artem Kucheravy received his Bachelor's degree in 2024 from Riga Technical University. At present, he is a Master's student at the Institute of Applied Computer Systems of Riga Technical University. His current research interests include information retrieval, natural language processing, and machine learning.
E-mail: artjoms.kucerjavijs@edu.rtu.lv

Gints Jēkabsons received his Doctoral degree in 2009 from Riga Technical University. At present, he is an Assistant Professor and Researcher at the Institute of Applied Computer Systems of Riga Technical University. His current research interests include information retrieval, natural language processing, and machine learning.
E-mail: gints.jekabsons@rtu.lv
ORCID iD: https://orcid.org/0000-0002-9575-2488