MUSTS: MUltilingual Semantic Textual Similarity Benchmark

Tharindu Ranasinghe[♢], **Hansi Hettiarachchi**[♢], **Constantin Orăsan**[♡] **and Ruslan Mitkov**[♢]
[♢]School of Computing and Communications, Lancaster University, UK
[♡]Centre for Translation Studies, University of Surrey, UK
{t.ranasinghe, h.hettiarachchi, r.mitkov}@lancaster.ac.uk
c.orasan@surrey.ac.uk

Abstract

Predicting semantic textual similarity (STS) is a complex and ongoing challenge in natural language processing (NLP). Over the years, researchers have developed a variety of supervised and unsupervised approaches to calculate STS automatically. Additionally, various benchmarks, which include STS datasets, have been established to consistently evaluate and compare these STS methods. However, they largely focus on high-resource languages, mixed with datasets annotated focusing on relatedness instead of similarity and contain automatically translated instances. Therefore, no dedicated benchmark for multilingual STS exists. To solve this gap, we introduce the Multilingual Semantic Textual Similarity Benchmark (MUSTS), which spans 13 languages, including low-resource languages. By evaluating more than 25 models on MUSTS, we establish the most comprehensive benchmark of multilingual STS methods. Our findings confirm that STS remains a challenging task, particularly for low-resource languages.

1 Introduction

Semantic textual similarity (STS) measures the extent to which two sentences convey the same meaning (Cer et al., 2017). Automatically measuring STS is a foundational natural language understanding (NLU) problem relevant to numerous applications (Mu and Lim, 2024), including machine translation (Wieting et al., 2019; Ranasinghe et al., 2020), text summarisation (Majumder et al., 2024), question answering (Mitkov et al., 2023), and information retrieval (Iida and Okazaki, 2021).

Over the years, researchers have developed various supervised and unsupervised approaches to calculate STS, ranging from training recurrent neural networks (RNNs) (Mueller and Thyagarajan, 2016; Ranasinghe et al., 2019b), fine-tuning pre-trained language models (Chandrasekaran and Mago, 2021) and prompting large language models (LLMs) (Chen et al., 2023). The organisation of SemEval shared tasks from 2012 to 2017 has fuelled the development of these neural network architectures (Cer et al., 2017; Agirre et al., 2016, 2015, 2014, 2013, 2012).

Natural language understanding (NLU) benchmarks such as GLUE (Wang et al., 2018) have been introduced to enable the systematic evaluation and comparison of STS methods. The multilingual adaptation of GLUE in languages such as Korean (Park et al., 2021), Japanese (Kurihara et al., 2022), and Sinhala (Ranasinghe et al., 2025) also included STS tasks. From the multilingual benchmarks, the Massive Text Embedding Benchmark (MTEB) (Muennighoff et al., 2023) includes 12 STS datasets spanning multiple languages, making it a valuable resource for multilingual text similarity evaluation (Su et al., 2023). However, despite its widespread adoption, several notable weaknesses are associated with the STS datasets included in MTEB.

(*i*) **Language Coverage** - None of the languages in the STS datasets of MTEB benchmark are low-resource (Muennighoff et al., 2023). According to Joshi et al. (2020), they all fall into the resource-rich *'Winners'* or *'Underdogs'* categories. Consequently, it remains uncertain how the top models in MTEB would generalise into low-resource languages as unsupervised STS methods.

(*ii*) **Text Similarity *vs.* Relatedness** - Several STS datasets in MTEB, including Polish (Dadas et al., 2020) and Finnish (Kanerva et al., 2021) have been annotated with a focus on relatedness rather than similarity. While similarity involves paraphrasal or entailment relations, relatedness accounts for all of the commonalities that can exist between two sentences (Abdalla et al., 2023; Morris and Hirst, 1991). As a result, datasets focused on relatedness cannot be classified as STS datasets.

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 331–353

Language	Family	|Train|	|Test|	Reference	Source	MTEB
Arabic	Afro-Asiatic (Semitic)	1080	250	Cer et al. (2017)	SNLI	✖
Brazilian Portuguese	Indo-European (Romance)	2500	2000	Fonseca et al. (2016)	SICK	✖
Czech	Indo-European (Slavic)	925	500	Sido et al. (2024)	News	✖
English	Indo-European (Germanic)	5750	1380	Cer et al. (2017)	SNLI	✔
French	Indo-European (Romance)	600	410	Cardon and Grabar (2020)	Wikipedia & Vikidia	✖
Japanese	Japonic	12500	1460	Kurihara et al. (2022)	YJ Caption	✔
Korean	Koreanic	5750	1380	Ham et al. (2020)	SNLI	✖
Portuguese	Indo-European (Romance)	2500	2000	Fonseca et al. (2016)	SICK	✖
Romanian	Indo-European (Romance)	5750	1380	Dumitrescu et al. (2021)	SNLI	✔
Serbian	Indo-European (Slavic)	953	239	Batanović et al. (2018)	paraphrase.sr	✖
Sinhala	Indo-European (Indo-Aryan)	5000	100	Kadupitiya et al. (2016)	SICK	✖
Spanish	Indo-European (Romance)	1620	250	Cer et al. (2017)	SNLI	✔
Tamil	Dravidian	2500	100	Nilaxan and Ranathunga (2021)	SICK	✖

Table 1: Overview of the languages included in MUSTS, listing the language, language family, dataset size (train and test), reference, source, and whether the dataset is part of MTEB. We group the language into high-resource, mid-resource, and low-resource based on Joshi et al. (2020).

(*iii*) **Machine Translations** - Several STS datasets in MTEB, such as German and Russian, were generated by machine translations of English STS datasets without post-editing. These datasets risk propagating translation errors and introducing stylistic biases that affect model training and evaluation (Mahfuz et al., 2025; Hettiarachchi et al., 2025). As a result, the validity of using these datasets for model evaluation remains questionable (Mager et al., 2018).

Although some of these limitations are acceptable for an embedding evaluation benchmark like MTEB (Muennighoff et al., 2023), they prevent the datasets within MTEB from being suitable as an STS benchmark. Therefore, in this research, we compile MUSTS: MUltilingual Semantic Textual Similarity Benchmark addressing the limitations mentioned above. Furthermore, we evaluate several popular unsupervised and supervised STS methods on MUSTS and report the results.

The **main findings** of the paper are;

- LLM prompting and LLM-based encoders provide the best results as unsupervised STS methods for high and mid-resource languages. However, for low-resource languages, simple STS methods like smooth inverse frequency (Arora et al., 2017) and earlier transformer-based sentence encoders like LaBSE (Feng et al., 2022) outperform LLMs.
- MTEB ranking does not reflect in MUSTS, encouraging more low-resource language inclusion in multilingual benchmarks.
- Training transformers provide the best results for STS even for low-resource languages, aligning with the other regression-based NLP tasks (Qian et al., 2024; Ma et al., 2024).

2 MUSTS: Multilingual Semantic Textual Similarity Benchmark

Table 1 shows datasets in 13 languages that we identified fulfilling our requirements to be included in MUSTS. More statistics are available in Appendix D. We assign each language into one of the three categories following Joshi et al. (2020), where *'Winners'* are high-resource, *'Underdogs'* are mid-resource and the rest are low-resource. Compared to MTEB, MUSTS provides a better coverage in mid and low-resource languages as it includes languages like Sinhala and Tamil (De Silva, 2019).

Unlike MTEB, MUSTS only contains datasets that followed STS annotation guidelines from Agirre et al. (2013) (Appendix A) without mixing with other concepts such as relatedness and paraphrasing. While some languages, such as Portuguese (Fonseca et al., 2016) and Sinhala (Kadupitiya et al., 2016), include sentences from the SICK dataset (Marelli et al., 2014), which focus on semantic relatedness, they were reannotated for the STS labels. From the STS datasets that were machine-translated from an English source, such as Sinhala (Kadupitiya et al., 2016) and Tamil (Nilaxan and Ranathunga, 2021), we only include those that have undergone a post-editing process. Finally, we restrict our selection to STS datasets published in peer-reviewed papers.

Eliminated Datasets While Faroese (Snæbjarnarson et al., 2023) and Bengali (Shajalal and Aono, 2018) STS datasets satisfy MUSTS criteria, they contain less than 1000 instances. As MUSTS aims to evaluate supervised models as well, we decided not to include datasets with less than 1000 instances.

3 Methods

We evaluated the following unsupervised and supervised STS methods on MUSTS.

3.1 Unsupervised STS Methods

3.1.1 Vector Averaging

The first unsupervised STS method averages the word embeddings of both sentences and computes their cosine similarity. We used three multilingual transformer models; `XLM-R Large` (Conneau et al., 2020), `RemBERT` (Chung et al., 2021), and `infoXLM Large` (Chi et al., 2021) to obtain embeddings. For the words which are split into sub-tokens by a particular tokeniser, we used the first sub-token embedding as the word's representation.

3.1.2 Smooth Inverse Frequency

Taking the average of word embeddings in a sentence gives equal weights for words such as *but, just*, etc. Smooth Inverse Frequency (SIF) addresses this in two steps (Arora et al., 2017);

(*i*) **Weighting**: SIF computes a weighted average of word embeddings, where each word is weighted by $\frac{a}{a+p(w)}$, with a which we set to 0.001 and $p(w)$ as the word's estimated frequency in a reference corpus. For these experiments, we used HPLT (de Gibert et al., 2024).

(*ii*) **Common Component Removal**: It then computes the principal component of these embeddings across sentences and removes their projections onto the first principal component, reducing noise from frequent words like stop words.

After these two steps, we compute the cosine similarity between the two embeddings (Ranasinghe et al., 2019a). We employed the same multilingual transformer models used in §3.1.1 in SIF too.

3.1.3 Sentence Embeddings

We also utilised multilingual sentence encoders to generate sentence embeddings and calculate the cosine similarity to represent the STS. Specifically, we used `LaBSE` (Feng et al., 2022), `all-MiniLM-L12-v2`, and `LASER` (Heffernan et al., 2022) as the sentence encoders.

3.1.4 LLM Prompting

Following recent advances in NLP, we evaluated LLMs for measuring STS using a prompt-based approach. We conducted experiments with two recently released multilingual LLMs: `Llama-3.1-8B-Instruct` (Touvron et al., 2023) and `Mistral-8B-Instruct` (Jiang et al., 2023), using four prompting strategies.

(*i*) **Zero-shot (ZS)**: We asked the models to predict similarity solely by following instructions without providing any examples.

(*ii*) **Few-shot-English (FS-En)**: Under this variant, we provided five English examples randomly chosen from the training dataset with similarity scores spread across the entire range (0-5).

(*iii*) **Few-shot-Monolingual (FS-Mono)**: Unlike FS-En, we provided the model with five examples in the same language as the given sentence pair for measuring STS. We used the same approach as (*ii*) to subsample training data.

(*iv*) **Few-shot with Chain of thoughts (FS-CoT)**: We provided the model with a small set of examples, each demonstrating a series of intermediate reasoning steps that break down the task for better understanding. We utilised six English examples using the explanations in Table 3 in the Appendix A.

We adapted the original prompt templates from Wang et al. (2024a) and refined them based on our initial experiments. More details on prompt design and our prompts are provided in Appendix B.

3.1.5 LLM-Encoders

We also employed the top six LLM-based embedding models in MTEB leaderboard as of December 2024. Namely, they are `NV-Embed-v2` (Lee et al., 2024), `bge-en-icl` (Li et al., 2024), `stella_en_1.5B_v5`, `SFR-Embedding-2_R`, `gte-Qwen2-7B-instruct` (Li et al., 2023) and `gte-Qwen2-1.5B-instruct` (Li et al., 2023). We compute the cosine similarity between the embeddings corresponding to the two sentences to represent the similarity.

3.2 Supervised STS Methods

3.2.1 Transformers

As the first supervised STS approach, we concatenated training sets from all the languages from MUSTS and trained three multilingual transformer models; `XLM-R Large` (Conneau et al., 2020), `RemBERT` (Chung et al., 2021), and `infoXLM Large` (Chi et al., 2021). The architecture and the configurations are available in Appendix C.1.

	Low-resource			Mid-resource							High-resource			
	Se	Si	Ta	Ar	Br-Pt	Cz	Ja	Ko	Pt	Ro	En	Es	Fr	Avg
§3.1.1 Word Vector Average														
XLM-R Large	0.499	0.246	0.259	0.335	0.460	0.416	0.265	0.388	0.394	0.420	0.318	0.383	0.610	0.384
RemBERT	0.370	0.120	0.242	0.336	0.439	0.509	0.254	0.387	0.362	0.450	0.406	0.518	0.673	0.390
infoXLM Large	0.475	0.204	0.322	0.424	0.494	0.511	0.370	0.450	0.389	0.490	0.427	0.551	0.622	0.440
§3.1.2 Smooth Inverse Frequency														
XLM-R Large	0.589	0.193	0.266	0.504	0.461	0.544	0.294	0.459	0.483	0.505	0.400	0.577	0.762	0.464
RemBERT	0.557	0.261	0.197	0.532	0.438	0.607	0.245	0.459	0.415	0.518	0.530	0.641	0.751	0.473
infoXLM Large	0.640	0.336	0.426	0.602	0.491	0.618	0.426	0.539	0.483	0.538	0.534	0.662	0.792	0.545
§3.1.3 Sentence Encoders														
LaBSE	0.756	0.499	0.551	0.690	0.708	0.782	0.761	0.705	0.730	0.715	0.722	0.808	0.891	0.716
MiniLM-L12-v2	0.708	0.087	0.258	0.791	0.677	0.865	0.778	0.744	0.720	0.796	0.844	0.855	0.802	0.686
LASER	0.597	0.292	0.201	0.674	0.653	0.743	0.736	0.689	0.656	0.686	0.697	0.796	0.820	0.581
§3.1.4 LLM Prompting														
Llama-3.1-8B-Instruct ZS	0.746	0.396	0.549	0.713	0.735	0.759	0.827	0.729	0.747	0.706	0.801	0.802	0.806	0.717
Llama-3.1-8B-Instruct FS-En	0.720	0.414	0.360	0.646	0.682	0.753	0.699	0.706	0.700	0.693	0.761	0.742	0.798	0.667
Llama-3.1-8B-Instruct FS-Mono	0.745	0.485	0.313	0.681	0.723	0.689	0.789	0.750	0.735	0.740	0.761	0.678	0.862	0.689
Llama-3.1-8B-Instruct FS-CoT	0.276	0.364	0.525	0.705	0.596	0.759	0.811	0.705	0.590	0.621	0.725	0.698	0.599	0.613
Mistral-8B-Instruct ZS	0.360	0.085	0.354	0.741	0.598	0.740	0.807	0.717	0.565	0.655	0.743	0.775	0.705	0.603
Mistral-8B-Instruct FS-En	0.512	0.313	0.256	0.730	0.663	0.803	0.819	0.754	0.647	0.682	0.759	0.802	0.705	0.650
Mistral-8B-Instruct FS-Mono	0.569	0.327	0.351	0.733	0.697	0.558	0.687	0.750	0.716	0.737	0.759	0.799	0.671	0.643
Mistral-8B-Instruct FS-CoT	0.628	0.342	0.311	0.610	0.554	0.717	0.807	0.695	0.622	0.632	0.764	0.716	0.750	0.627
§3.1.5 Top LLM-based Encoders on MTEB (Muennighoff et al., 2023)														
NV-Embed-v2	0.745	0.343	0.397	0.785	0.775	0.857	0.792	0.792	0.806	0.799	0.843	0.881	0.896	0.743
bge-en-icl	0.725	0.292	0.322	0.788	0.691	0.824	0.762	0.743	0.739	0.734	0.805	0.822	0.839	0.668
stella_en_1.5B_v5	0.651	0.124	0.341	0.643	0.742	0.825	0.784	0.536	0.774	0.704	0.860	0.871	0.889	0.663
SFR-Embedding-2_R	0.836	0.399	0.334	0.788	0.772	0.861	0.778	0.785	0.811	0.791	0.863	0.869	0.896	0.750
gte-Qwen2-7B-instruct	0.695	0.280	0.397	0.818	0.723	0.816	0.772	0.759	0.740	0.759	0.816	0.857	0.878	0.711
gte-Qwen2-1.5B-instruct	0.591	0.312	0.353	0.746	0.724	0.776	0.751	0.737	0.756	0.682	0.853	0.864	0.888	0.690
§3.2.1 Training Transformers														
XLM-R Large	0.866	0.810	0.832	0.842	0.811	0.870	0.833	0.801	0.835	0.825	0.887	0.891	0.898	0.846
RemBERT	0.868	0.821	0.841	0.848	0.820	0.876	0.841	0.808	0.841	0.829	0.889	0.899	0.900	0.852
infoXLM Large	**0.869**	**0.825**	**0.848**	**0.851**	**0.825**	**0.878**	**0.844**	**0.811**	**0.848**	**0.839**	**0.893**	**0.904**	**0.905**	**0.880**
§3.2.2 Training LLM-based Encoders														
stella_en_1.5B_v5	0.692	0.392	0.651	0.665	0.651	0.763	0.751	0.592	0.753	0.682	0.765	0.752	0.844	0.689
gte-Qwen2-1.5B-instruct	0.683	0.420	0.770	0.648	0.683	0.758	0.732	0.638	0.672	0.650	0.741	0.770	0.864	0.691

Table 2: Spearman (r) correlation between model predictions and human annotations. The best result for each language (any method) is in **bold**, and the best unsupervised result is underlined.

3.2.2 Training LLM-Encoders

We also selected two small LLM-Encoders that could be trained on an NVidia L40 48G GPU. We concatenated training sets from all the languages from MUSTS and trained these encoders using the sentence transformers architecture (details in Appendix C.2) (Reimers and Gurevych, 2019). We specifically used `gte-Qwen2-1.5B-instruct` and `stella_en_1.5B_v5`.

4 Results and Analysis

Table 2 shows the Spearman correlation between predictions of each model and human annotations. We describe our main findings in the following list.

(*i*) **LLMs as Unsupervised STS Methods** - The results show that both LLM-based unsupervised STS approaches, encoders and prompting, provide excellent results in mid and high-resource languages. Interestingly, some of these LLMs do not even directly support languages such as Czech and Romanian, yet they provide impressive results for the STS task. LLM-based encoders jointly outperformed prompting in nine out of ten mid/high resource languages. Finally, there is no clear winner among the four prompting strategies. However, we notice **FS-Mono** provides better results than **FS-En** in the majority of languages.

Both LLM-based encoders and prompting do not perform well in low-resource languages, particularly Sinhala and Tamil. Interestingly, SIF, a simple STS method, produced competitive results compared to LLM-based methods in low-resource languages. From the experimented unsupervised methods, `LABSE` (Feng et al., 2022) provided the best results for low-resource languages.

(*ii*) **MUSTS ranking *vs.* MTEB Ranking** - The results in row §3.1.5, Table 2 show that model ranking in MTEB is not reflected in MUSTS. For instance, `SFR-Embedding-2_R`, the fourth-ranked model in MTEB, ranked first among the text em-

bedding models in MUSTS. We attribute this discrepancy primarily to the language coverage of MUSTS and suggest that NLP benchmarks should incorporate low-resource languages for a fairer evaluation.

(*iii*) **Supervised STS Models** - Training transformer models provided the best results for STS. Among the multilingual transformer models tested, `info-XLM Large` achieved the highest performance across all 13 languages.

Interestingly, fine-tuning LLM-based encoders on MUSTS using the Sentence Transformers architecture did not lead to overall improvements. However, it did enhance performance in low-resource languages. We believe that techniques such as contrastive instruction tuning (Wang et al., 2024b) should be further explored to achieve better results.

5 Conclusions

In this paper, we compiled MUSTS, the most comprehensive multilingual STS benchmark up to date. MUSTS spans over 13 languages and includes carefully selected STS datasets in terms of task definition and human annotations. We make MUSTS publicly available together with the fine-tuning and the evaluation code, as well as a public leaderboard[1].

We evaluated more than 25 STS approaches on MUSTS. Our results showed that STS remains a challenging task, especially for low-resource languages. Furthermore, we show that LLM-based methods thrive in the STS task for mid and high-resource languages but struggle in low-resource languages. Our findings highlight the need for fair multilingual evaluations in STS.

Limitations

We only used the models that could be experimented on an NVidia L40 48G GPU. Due to these computational limitations, we did not experiment with the larger models. Larger models may perform differently in this STS task.

The examples used in the few-shot scenarios (§3.1.4) were randomly sampled since we do not have the knowledge to prepare good-quality examples for all languages. Results might be different if these examples were carefully chosen by native speakers.

[1]Available at https://github.com/TharinduDR/MUSTS

Ethical Considerations

All the datasets released with MUSTS are publicly available with a CC BY 4.0 licence. Furthermore, all the models that we experimented with in this paper are publicly available in HuggingFace (Lhoest et al., 2021).

Acknowledgements

We would like to thank the anonymous reviewers for their positive and valuable feedback. We further thank the creators of the datasets used in this paper for making the datasets publicly available for our research.

The experiments in this paper were conducted in UCREL-HEX (Vidler and Rayson, 2024). We would like to thank John Vidler for the continuous support and maintenance of the UCREL-HEX infrastructure, which enabled the efficient execution of our experiments.

References

Mohamed Abdalla, Krishnapriya Vishnubhotla, and Saif Mohammad. 2023. What makes sentences semantically related? a textual relatedness dataset and empirical study. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 782–796, Dubrovnik, Croatia. Association for Computational Linguistics.

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Iñigo Lopez-Gazpio, Montse Maritxalar, Rada Mihalcea, German Rigau, Larraitz Uria, and Janyce Wiebe. 2015. SemEval-2015 task 2: Semantic textual similarity, English, Spanish and pilot on interpretability. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 252–263, Denver, Colorado. Association for Computational Linguistics.

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2014. SemEval-2014 task 10: Multilingual semantic textual similarity. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 81–91, Dublin, Ireland. Association for Computational Linguistics.

Eneko Agirre, Carmen Banea, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2016. SemEval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 497–511, San Diego, California. Association for Computational Linguistics.

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. SemEval-2012 task 6: A pilot on semantic textual similarity. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 385–393, Montréal, Canada. Association for Computational Linguistics.

Eneko Agirre, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, and Weiwei Guo. 2013. *SEM 2013 shared task: Semantic textual similarity. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, pages 32–43, Atlanta, Georgia, USA. Association for Computational Linguistics.

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2017. A simple but tough-to-beat baseline for sentence embeddings. In *International Conference on Learning Representations*.

Vuk Batanović, Miloš Cvetanović, and Boško Nikolić. 2018. Fine-grained semantic textual similarity for Serbian. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Rémi Cardon and Natalia Grabar. 2020. A French corpus for semantic similarity. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6889–6894, Marseille, France. European Language Resources Association.

Daniel Cer, Mona Diab, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Dhivya Chandrasekaran and Vijay Mago. 2021. Evolution of semantic similarity—a survey. *ACM Comput. Surv.*, 54(2).

Xuanting Chen, Junjie Ye, Can Zu, Nuo Xu, Rui Zheng, Minlong Peng, Jie Zhou, Tao Gui, Qi Zhang, and Xuanjing Huang. 2023. How robust is gpt-3.5 to predecessors? a comprehensive study on language understanding tasks. *arXiv preprint arXiv:2303.00293*.

Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2021. InfoXLM: An information-theoretic framework for cross-lingual language model pre-training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3576–3588, Online. Association for Computational Linguistics.

Hyung Won Chung, Thibault Fevry, Henry Tsai, Melvin Johnson, and Sebastian Ruder. 2021. Rethinking embedding coupling in pre-trained language models. In *International Conference on Learning Representations*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Slawomir Dadas, Michał Perełkiewicz, and Rafał Poświata. 2020. Evaluation of sentence representations in Polish. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1674–1680, Marseille, France. European Language Resources Association.

Ona de Gibert, Graeme Nail, Nikolay Arefyev, Marta Bañón, Jelmer van der Linde, Shaoxiong Ji, Jaume Zaragoza-Bernabeu, Mikko Aulamo, Gema Ramírez-Sánchez, Andrey Kutuzov, Sampo Pyysalo, Stephan Oepen, and Jörg Tiedemann. 2024. A new massive multilingual dataset for high-performance language technologies. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 1116–1128, Torino, Italia. ELRA and ICCL.

Nisansa De Silva. 2019. Survey on publicly available sinhala natural language processing tools and research. *arXiv preprint arXiv:1906.02358*.

Stefan Daniel Dumitrescu, Petru Rebeja, Beata Lorincz, Mihaela Gaman, Andrei Avram, Mihai Ilie, Andrei Pruteanu, Adriana Stan, Lorena Rosia, Cristina Iacobescu, Luciana Morogan, George Dima, Gabriel Marchidan, Traian Rebedea, Madalina Chitez, Dani Yogatama, Sebastian Ruder, Radu Tudor Ionescu, Razvan Pascanu, and Viorica Patraucean. 2021. Liro: Benchmark and leaderboard for romanian language tasks. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Erick Rocha Fonseca, Leandro Borges dos Santos, Marcelo Criscuolo, and Sandra Maria Aluísio. 2016. Visão geral da avaliação de similaridade semântica e inferência textual. *Linguamática*, 8(2):3–13.

Jiyeon Ham, Yo Joong Choe, Kyubyong Park, Ilji Choi, and Hyungjoon Soh. 2020. KorNLI and KorSTS:

New benchmark datasets for Korean natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 422–430, Online. Association for Computational Linguistics.

Kevin Heffernan, Onur Çelebi, and Holger Schwenk. 2022. Bitext mining using distilled sentence representations for low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2101–2112, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hansi Hettiarachchi, Tharindu Ranasinghe, Paul Rayson, Ruslan Mitkov, Mohamed Gaber, Damith Premasiri, Fiona Anting Tan, and Lasitha Randunu Chandrakantha Uyangodage. 2025. Overview of the first workshop on language models for low-resource languages (LoResLM 2025). In *Proceedings of the First Workshop on Language Models for Low-Resource Languages*, pages 1–8, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hiroki Iida and Naoaki Okazaki. 2021. Incorporating semantic textual similarity and lexical matching for information retrieval. In *Proceedings of the 35th Pacific Asia Conference on Language, Information and Computation*, pages 582–591, Shanghai, China. Association for Computational Linguistics.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Jcs Kadupitiya, Surangika Ranathunga, and Gihan Dias. 2016. Sinhala short sentence similarity calculation using corpus-based and knowledge-based similarity measures. In *Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing (WSSANLP2016)*, pages 44–53, Osaka, Japan. The COLING 2016 Organizing Committee.

Jenna Kanerva, Filip Ginter, Li-Hsin Chang, Iiro Rastas, Valtteri Skantsi, Jemina Kilpeläinen, Hanna-Mari Kupari, Jenna Saarni, Maija Sevón, and Otto Tarkka. 2021. Finnish paraphrase corpus. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 288–298, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.

Kentaro Kurihara, Daisuke Kawahara, and Tomohide Shibata. 2022. JGLUE: Japanese general language understanding evaluation. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2957–2966, Marseille, France. European Language Resources Association.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2024. Nv-embed: Improved techniques for training llms as generalist embedding models. *arXiv preprint arXiv:2405.17428*.

Quentin Lhoest, Albert Villanova del Moral, Yacine Jernite, Abhishek Thakur, Patrick von Platen, Suraj Patil, Julien Chaumond, Mariama Drame, Julien Plu, Lewis Tunstall, Joe Davison, Mario Šaško, Gunjan Chhablani, Bhavitvya Malik, Simon Brandeis, Teven Le Scao, Victor Sanh, Canwen Xu, Nicolas Patry, Angelina McMillan-Major, Philipp Schmid, Sylvain Gugger, Clément Delangue, Théo Matussière, Lysandre Debut, Stas Bekman, Pierric Cistac, Thibault Goehringer, Victor Mustar, François Lagunas, Alexander Rush, and Thomas Wolf. 2021. Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–184, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Chaofan Li, MingHao Qin, Shitao Xiao, Jianlyu Chen, Kun Luo, Yingxia Shao, Defu Lian, and Zheng Liu. 2024. Making text embedders few-shot learners. *arXiv preprint arXiv:2409.15700*.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Wanqin Ma, Chenyang Yang, and Christian Kästner. 2024. (why) is my prompt getting worse? rethinking regression testing for evolving llm apis. In *Proceedings of the IEEE/ACM 3rd International Conference on AI Engineering - Software Engineering for AI*, CAIN '24, page 166–171, New York, NY, USA. Association for Computing Machinery.

Manuel Mager, Elisabeth Mager, Alfonso Medina-Urrea, Ivan Vladimir Meza Ruiz, and Katharina Kann. 2018. Lost in translation: Analysis of information loss during machine translation between polysynthetic and fusional languages. In *Proceedings of the Workshop on Computational Modeling of Polysynthetic Languages*, pages 73–83, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Tamzeed Mahfuz, Satak Kumar Dey, Ruwad Naswan, Hasnaen Adil, Khondker Salman Sayeed, and Haz Sameen Shahgir. 2025. Too late to train, too early to use? a study on necessity and viability of

low-resource Bengali LLMs. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 1183–1200, Abu Dhabi, UAE. Association for Computational Linguistics.

Goutam Majumder, Vikrant Rajput, Partha Pakray, Sivaji Bandyopadhyay, and Benoit Favre. 2024. Text summary evaluation based on interpretable semantic textual similarity. *Multimedia Tools and Applications*, 83(1):3233–3258.

Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 216–223, Reykjavik, Iceland. European Language Resources Association (ELRA).

Ruslan Mitkov, Le An Ha, Halyna Maslak, Tharindu Ranasinghe, and Vilelmini Sosoni. 2023. Automatic generation of multiple-choice test items from paragraphs using deep neural networks. In *Advancing Natural Language Processing in Educational Assessment*, pages 77–89. Routledge.

Jane Morris and Graeme Hirst. 1991. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics*, 17(1):21–48.

Wenchuan Mu and Kwan Hui Lim. 2024. Modelling text similarity: A survey. In *Proceedings of the 2023 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '23, page 698–705, New York, NY, USA. Association for Computing Machinery.

Jonas Mueller and Aditya Thyagarajan. 2016. Siamese recurrent architectures for learning sentence similarity. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1).

Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. 2023. 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.

Satkunanantham Nilaxan and Surangika Ranathunga. 2021. Monolingual sentence similarity measurement using siamese neural networks for sinhala and tamil languages. In *2021 Moratuwa Engineering Research Conference (MERCon)*, pages 567–572.

Sungjoon Park, Jihyung Moon, Sungdong Kim, Won Ik Cho, Ji Yoon Han, Jangwon Park, Chisung Song, Junseong Kim, Youngsook Song, Taehwan Oh, Joohong Lee, Juhyun Oh, Sungwon Lyu, Younghoon Jeong, Inkwon Lee, Sangwoo Seo, Dongjun Lee, Hyunwoo Kim, Myeonghwa Lee, Seongbo Jang, Seungwon Do, Sunkyoung Kim, Kyungtae Lim, Jongwon Lee, Kyumin Park, Jamin Shin, Seonghyun Kim, Lucy Park, Lucy Park, Alice Oh, Jung-Woo Ha (NAVER AI Lab), Kyunghyun Cho, and Kyunghyun Cho. 2021. Klue: Korean language understanding evaluation. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1.

Shenbin Qian, Archchana Sindhujan, Minnie Kabra, Diptesh Kanojia, Constantin Orasan, Tharindu Ranasinghe, and Fred Blain. 2024. What do large language models need for machine translation evaluation? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3660–3674, Miami, Florida, USA. Association for Computational Linguistics.

Tharindu Ranasinghe, Hansi Hettiarachchi, Nadeesha Pathirana, Damith Premasiri, Lasitha Uyangodage, Isuri Anuradha, Alistair Plum, Paul Rayson, and Ruslan Mitkov. 2025. Sinhala encoder-only language models and evaluation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, Vienna, Austria. Association for Computational Linguistics.

Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. 2019a. Enhancing unsupervised sentence similarity methods with deep contextualised word representations. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 994–1003, Varna, Bulgaria. INCOMA Ltd.

Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. 2019b. Semantic textual similarity with Siamese neural networks. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1004–1011, Varna, Bulgaria. INCOMA Ltd.

Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. 2020. TransQuest: Translation quality estimation with cross-lingual transformers. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5070–5081, Barcelona, Spain (Online). International Committee on Computational Linguistics.

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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Md Shajalal and Masaki Aono. 2018. Semantic textual similarity in bengali text. In *2018 International Conference on Bangla Speech and Language Processing (ICBSLP)*, pages 1–5.

Jakub Sido, Michal Seják, Ondřej Pražák, Miloslav Konopík, and Václav Moravec. 2024. Czech news dataset for semantic textual similarity. *Language Resources and Evaluation*, pages 1–18.

Vésteinn Snæbjarnarson, Annika Simonsen, Goran Glavaš, and Ivan Vulić. 2023. Transfer to a low-resource language via close relatives: The case study on Faroese. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 728–737, Tórshavn, Faroe Islands. University of Tartu Library.

Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2023. One embedder, any task: Instruction-finetuned text embeddings. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1102–1121, Toronto, Canada. Association for Computational Linguistics.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

John Vidler and Paul Rayson. 2024. UCREL - Hex; a shared, hybrid multiprocessor system. https://github.com/UCREL/hex.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Yuxia Wang, Minghan Wang, and Preslav Nakov. 2024a. Rethinking STS and NLI in large language models. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 965–982, St. Julian's, Malta. Association for Computational Linguistics.

Zhimeng Wang, Pinzheng Wang, Juntao Li, Yibin Chen, and Min Zhang. 2024b. Achieving stronger generation via simple contrastive tuning. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 8986–8999, Miami, Florida, USA. Association for Computational Linguistics.

John Wieting, Taylor Berg-Kirkpatrick, Kevin Gimpel, and Graham Neubig. 2019. Beyond BLEU: Training neural machine translation with semantic similarity. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4344–4355, Florence, Italy. Association for Computational Linguistics.

A Annotation Guidelines

All the datasets in MUSTS follow the annotation guidelines in Table 3.

5	*The two sentences are completely equivalent, as they mean the same thing.*
	The bird is bathing in the sink. Birdie is washing itself in the water basin.
4	*The two sentences are mostly equivalent, but some unimportant details differ.*
	Two boys on a couch are playing video games. Two boys are playing a video game.
3	*The two sentences are roughly equivalent, but some important information differs/missing.*
	John said he is considered a witness but not a suspect. "He is not a suspect anymore." John said.
2	*The two sentences are not equivalent, but share some details.*
	They flew out of the nest in groups. They flew into the nest together.
1	*The two sentences are not equivalent, but are on the same topic.*
	The woman is playing the violin. The young lady enjoys listening to the guitar.
0	*The two sentences are completely dissimilar.*
	The black dog is running through the snow. A race car driver is driving his car through the mud.

Table 3: Similarity scores with explanations and English examples from Agirre et al. (2013).

B Prompts

As a starting point, we used the prompt templates provided by Wang et al. (2024a), developed through a comprehensive experimental study, as the basis for our study. We made a few key updates. First, we revised the task description to explicitly mention 'semantic textual similarity' to avoid confusion with syntactic similarity, which our initial experiments suggested could occur when using just the term 'similarity'. We also added a new sentence to each prompt to clarify the expected output, aiming to reduce post-processing errors when extracting the scores. Table 4 summarises the final prompt templates.

C Supervised STS Models

C.1 Training Transformers

We trained the transformer models using the architecture in Figure 1.

- All our models use the AdamW (Loshchilov and Hutter, 2019) optimiser with a weight decay of 1e-8, learning rate of 2e-5, a warmup ratio of

task_description: Determine the semantic textual similarity between the following two sentences (S1, S2). The score should be ranging from 0.0 to 5.0, and can be a decimal.	
Strategy	**Prompt**
ZS	{task_description} Return the score only following the prefix 'Score:' without any other text or explanations. S1: {sentence1} S2: {sentence2}
FS	Five demonstration examples: Example 1: S1: Ebola UK: NHS staff 'panicked' after suspected Ebola cases. S2: UK says investigating 2 suspected MERS cases. Score: 0.0 Example 2: ... {task_description} Return the score only following the prefix 'Score:' without any other text or explanations. S1: {sentence1} S2: {sentence2}
FS-CoT	{task_description} Return the explanation and score only following the prefixes 'Explain:' and 'Score:' without any other text. Six demonstration examples with explanation for each: Example 1: S1: The black dog is running through the snow. S2: A race car driver is driving his car through the mud. Explain: S1 and S2 are completely dissimilar. Score: 0.0 Example 2: ... S1: {sentence1} S2: {sentence2}

Table 4: Prompt templates used for experiments

0.06 from the training data and are trained for five epochs with a batch size of 32 (gradient accumulation is applied when needed), and a maximum length of 512 tokens. The values of the hyper-parameters (including the number of training epochs) were set to fixed values in order to ensure consistency between all the models.

- All the models were evaluated while training using a development set that had one-fifth of the rows separated from the training set before the start of the training process.
- The best checkpoints were selected on the development set. We use the Spearman correlation as the checkpoint selection criterion.
- We trained our models on an NVidia L40 48G GPU. All models were trained with half-precision (fp16) using the default PyTorch implementation.

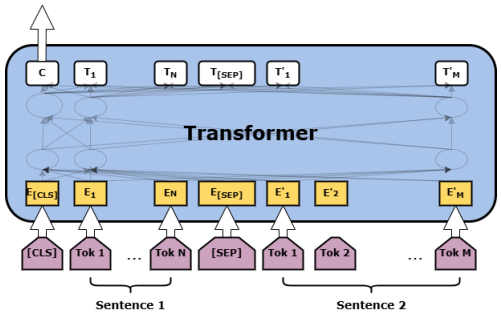


Figure 1: Architecture for using Transformers in STS.

C.2 Training LLM-based Encoders

We trained the LLM encoders using the architecture in Figure 2.

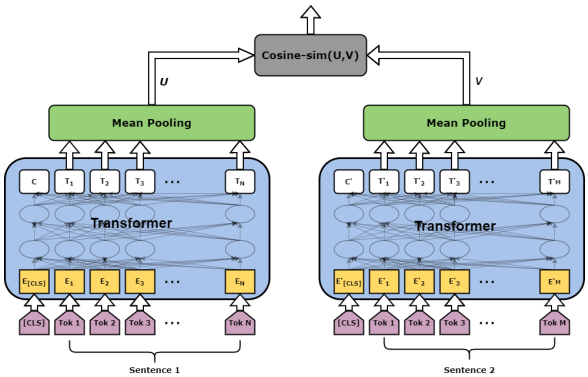


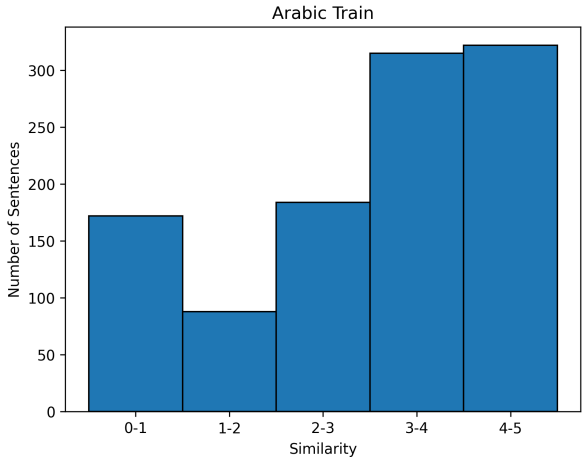
Figure 2: Architecture for using Transformers in STS.

- All models were trained using the same configurations mentioned in Appendix C.1, apart from the learning rate and warmup ratio, which were set to 1e-6 and 0.1, respectively.

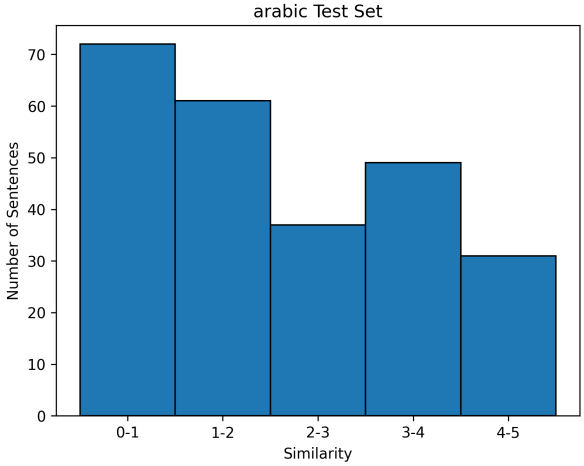
D MUSTS Statistics

In the following figures, we show label distribution and token overlap for each language.

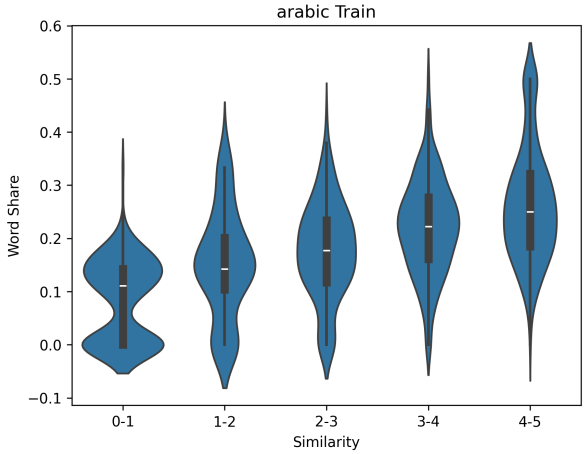
Arabic



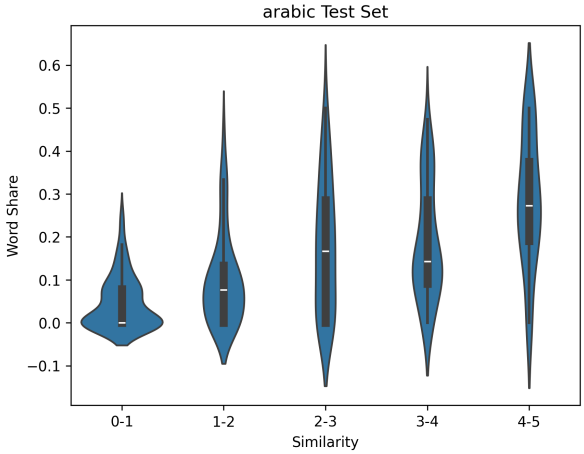
(1) Similarity distribution of Ar train set



(2) Similarity distribution of Ar test set

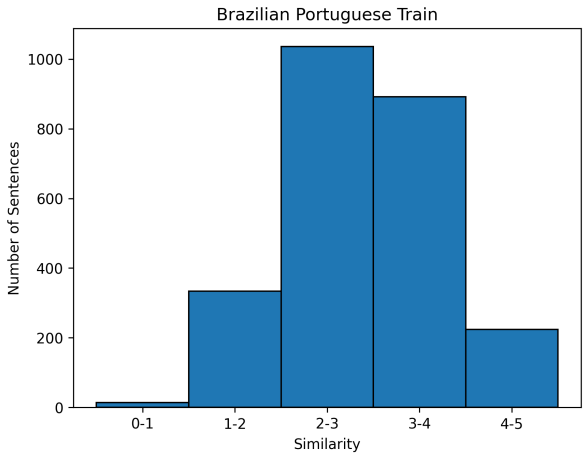


(3) Word share against similarity bins in Ar train set

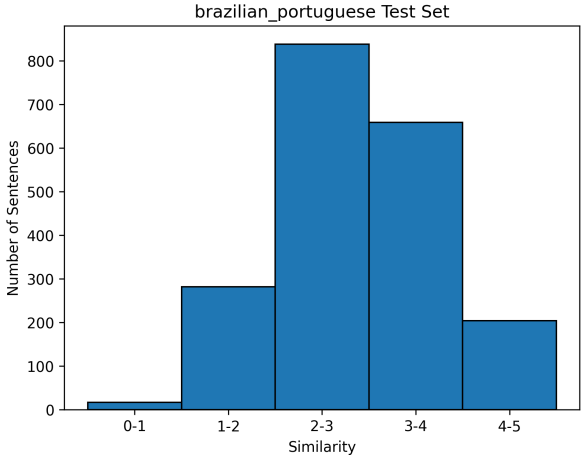


(4) Word share against similarity bins in Ar test set

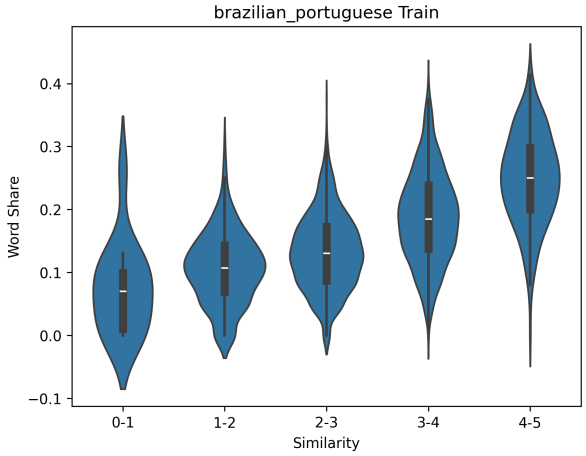
Brazilian Portuguese



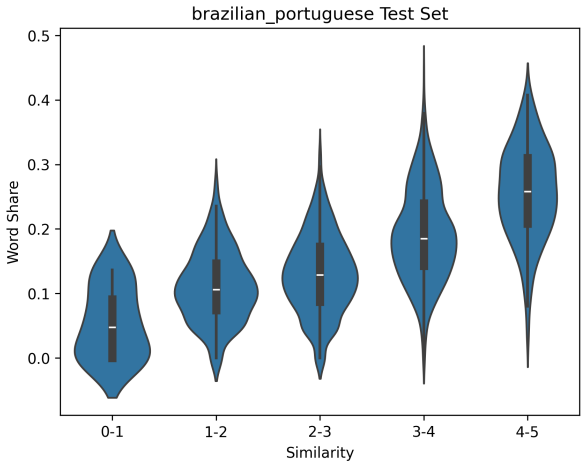
(1) Similarity distribution of Br-Pt train set



(2) Similarity distribution of Br-Pt test set

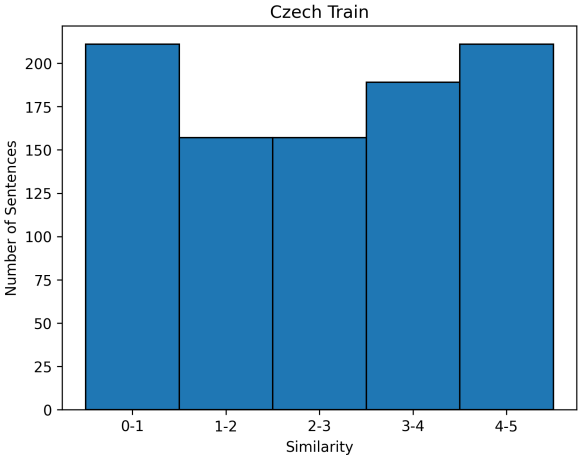


(3) Word share against similarity bins in Br-Pt train set

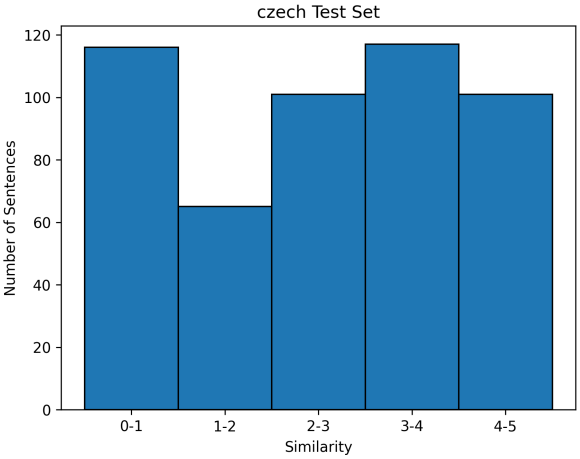


(4) Word share against similarity bins in Br-Pt test set

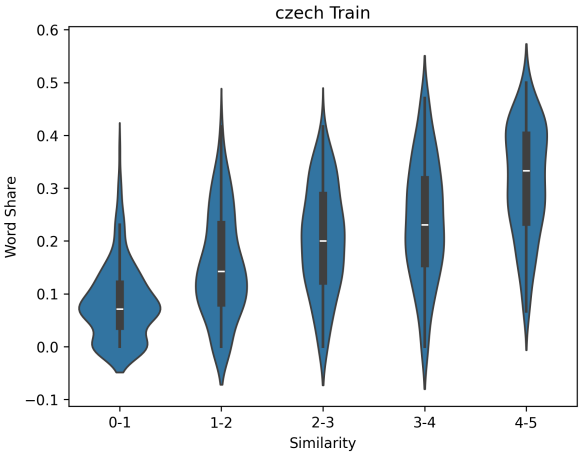
Czech



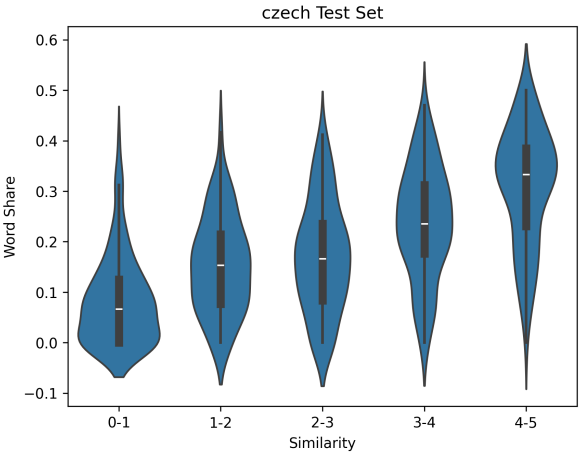
(1) Similarity distribution of Cz train set



(2) Similarity distribution of Cz test set

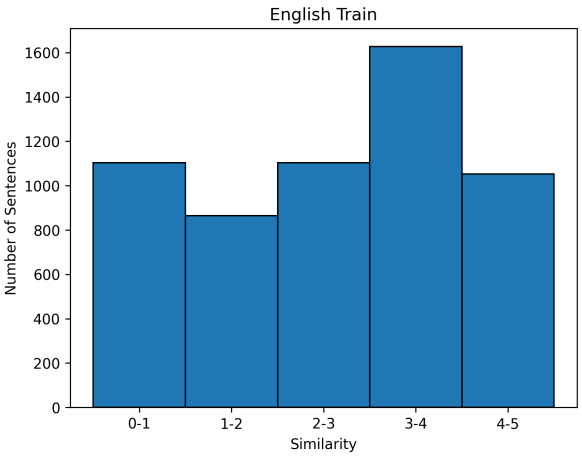


(3) Word share against similarity bins in Cz train set

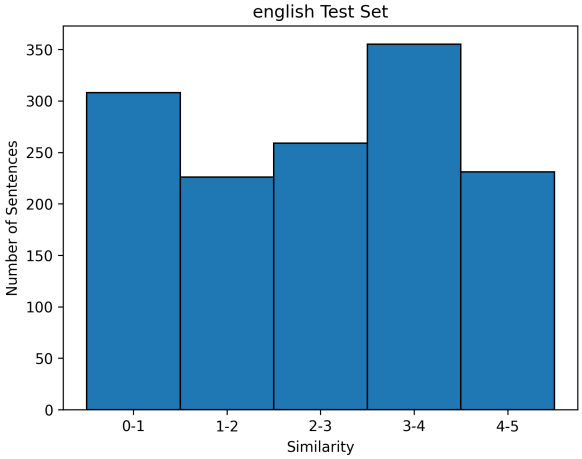


(4) Word share against similarity bins in Cz test set

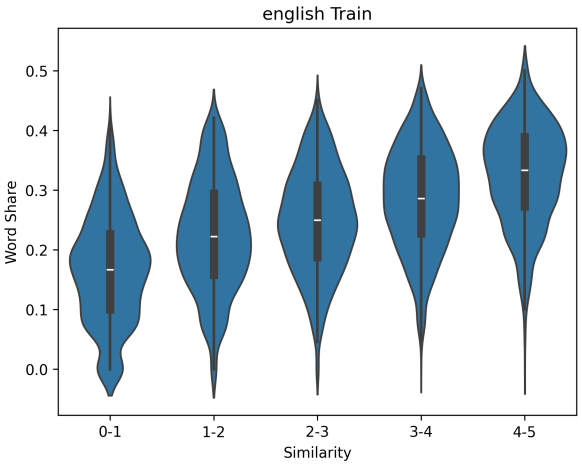
English



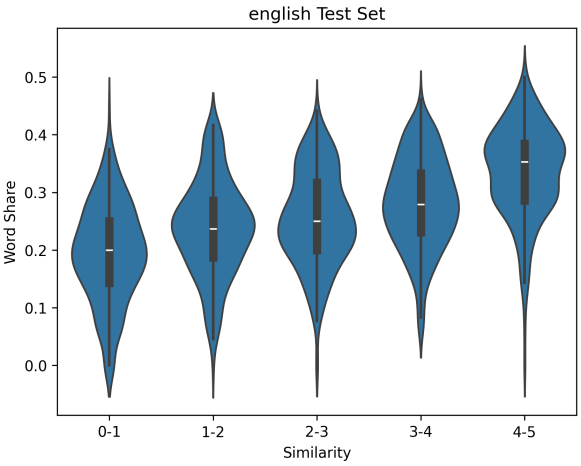
(1) Similarity distribution of En train set



(2) Similarity distribution of En test set

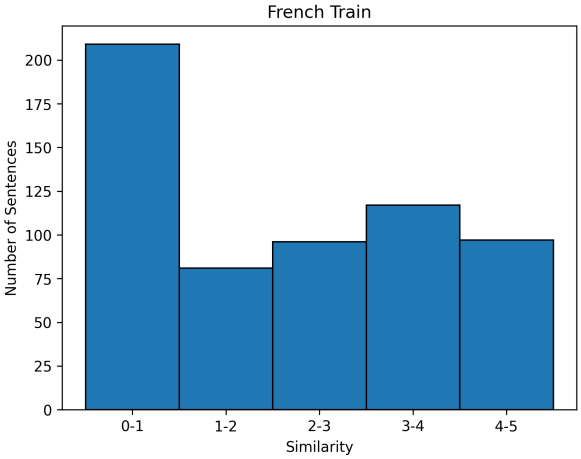


(3) Word share against similarity bins in En train set

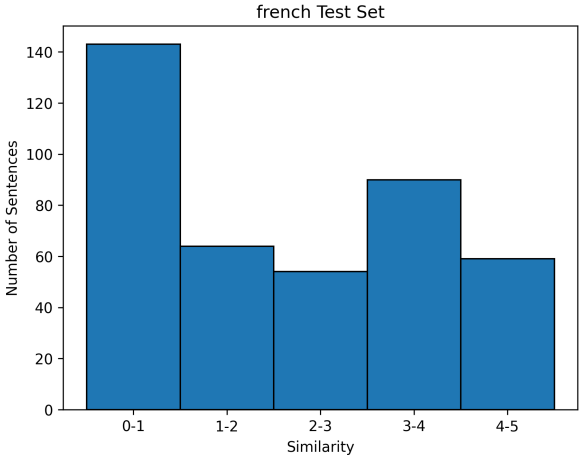


(4) Word share against similarity bins in En test set

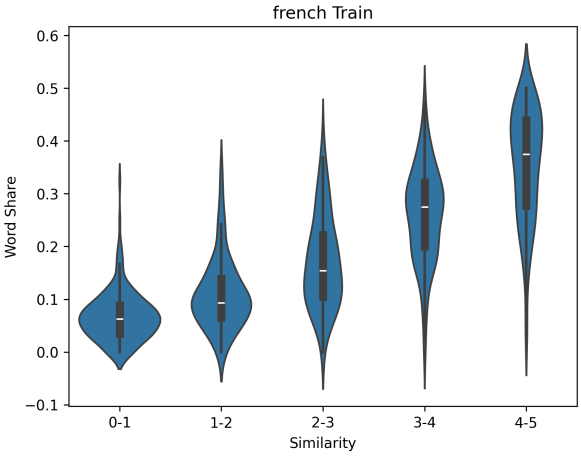
French



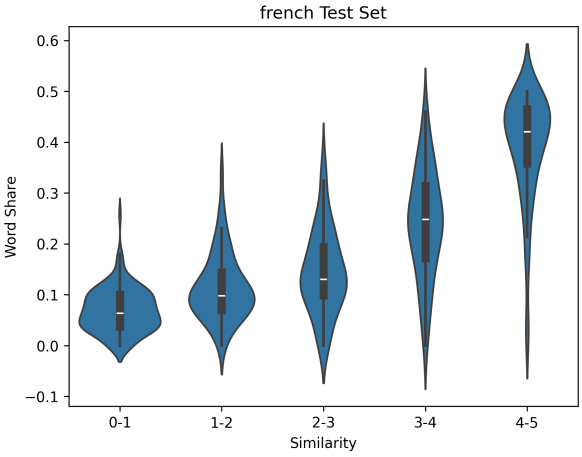
(1) Similarity distribution of Fr train set



(2) Similarity distribution of Fr test set

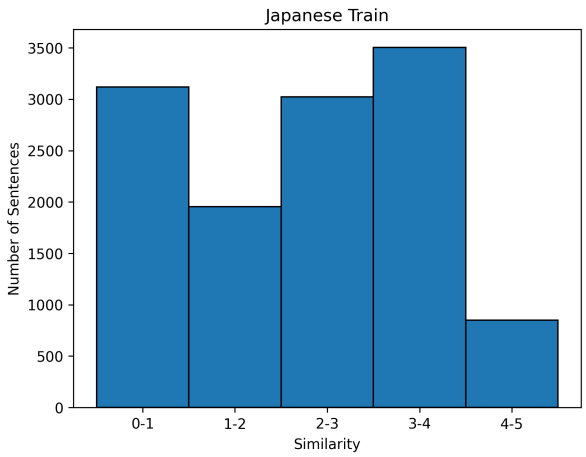


(3) Word share against similarity bins in Fr train set

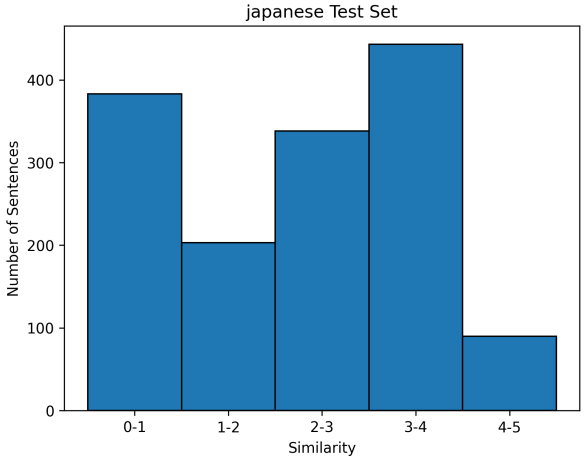


(4) Word share against similarity bins in Fr test set

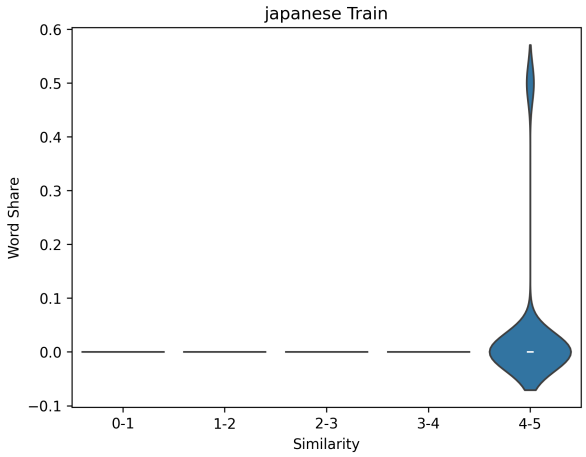
Japanese



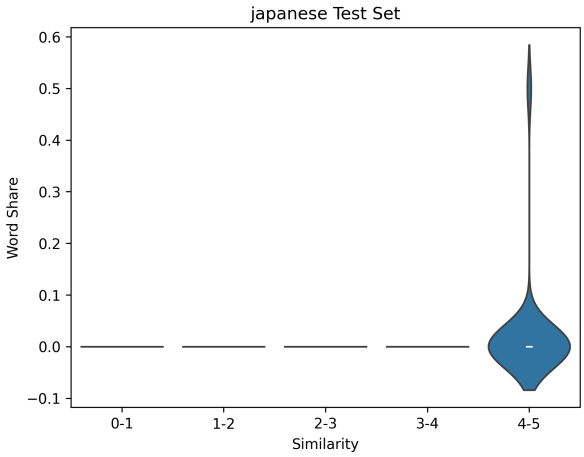
(5) Similarity distribution of Jp train set



(6) Similarity distribution of Jp test set

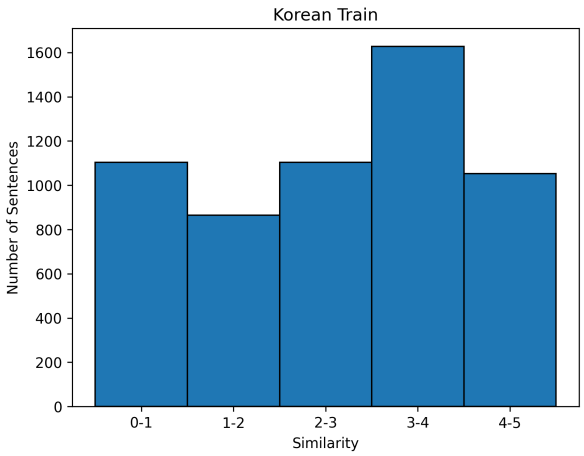


(7) Word share against similarity bins in Jp train set

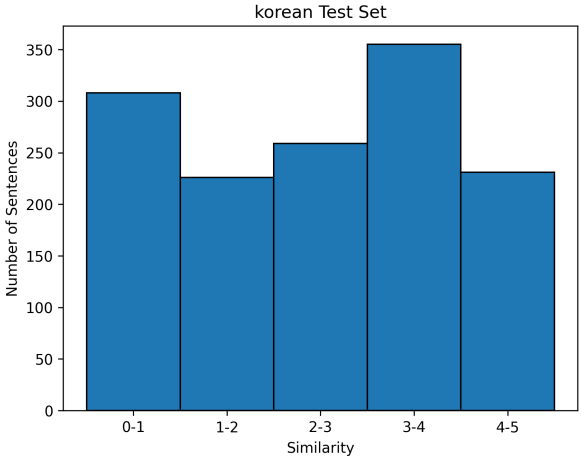


(8) Word share against similarity bins in Jp test set

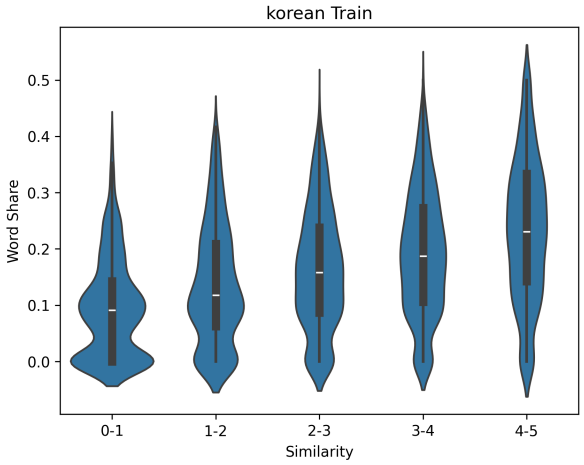
Korean



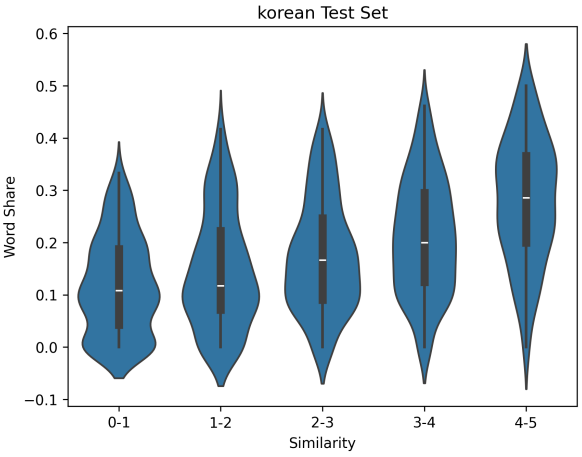
(1) Similarity distribution of Ko train set



(2) Similarity distribution of Ko test set

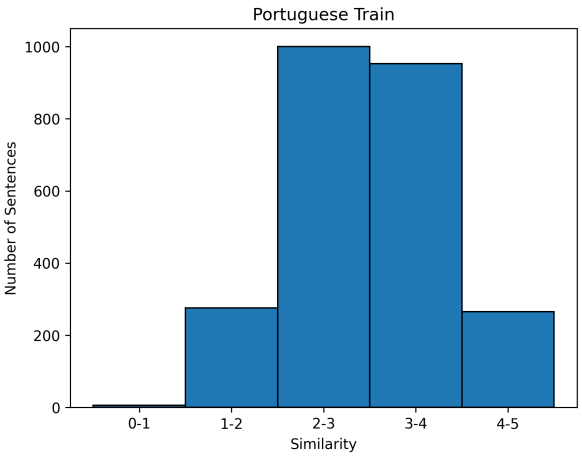


(3) Word share against similarity bins in Ko train set

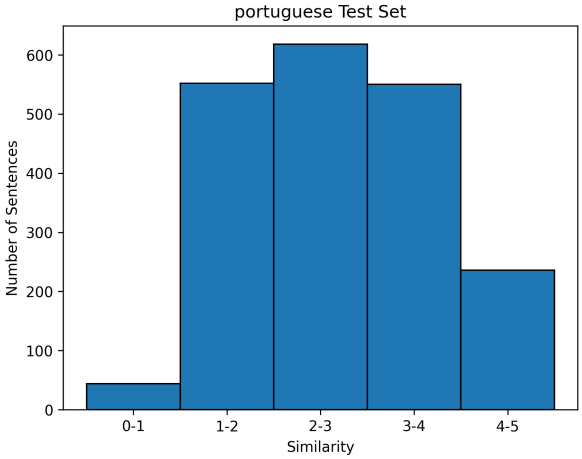


(4) Word share against similarity bins in Ko test set

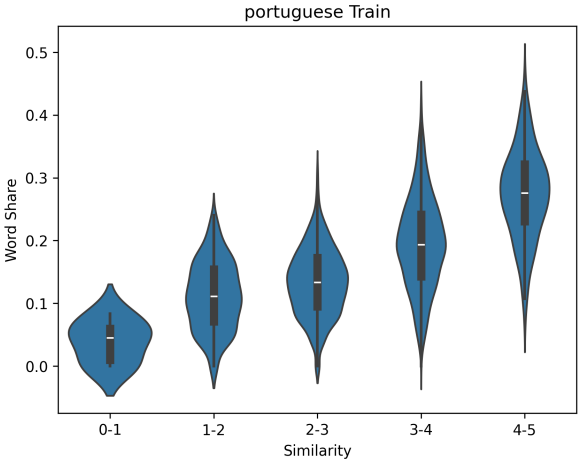
Portuguese



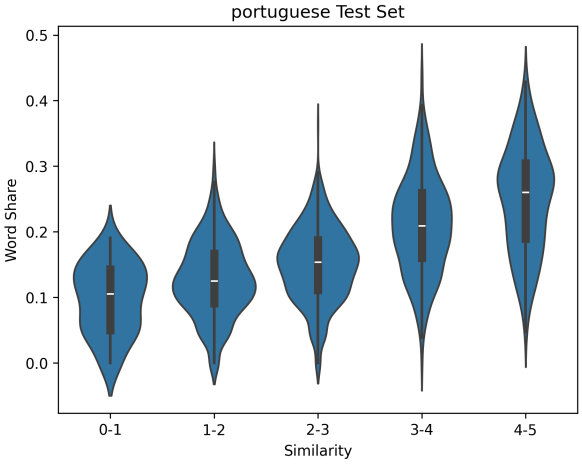
(1) Similarity distribution of Pt train set



(2) Similarity distribution of Pt test set

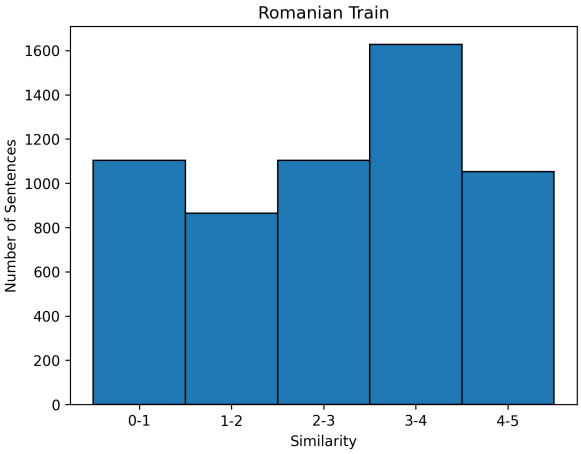


(3) Word share against similarity bins in Pt train set

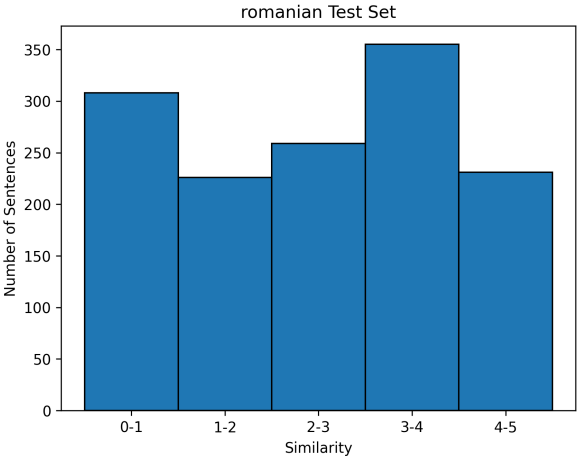


(4) Word share against similarity bins in Pt test set

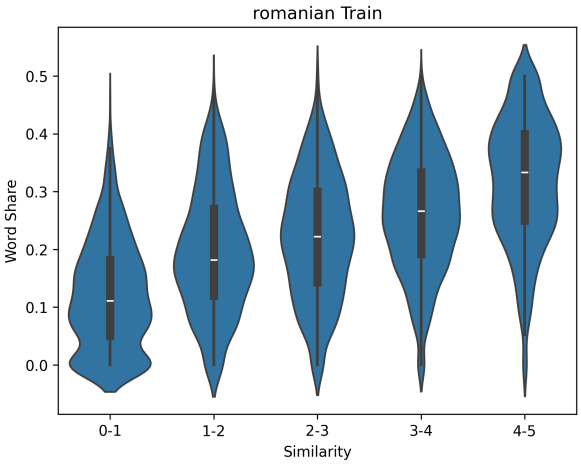
Romanian



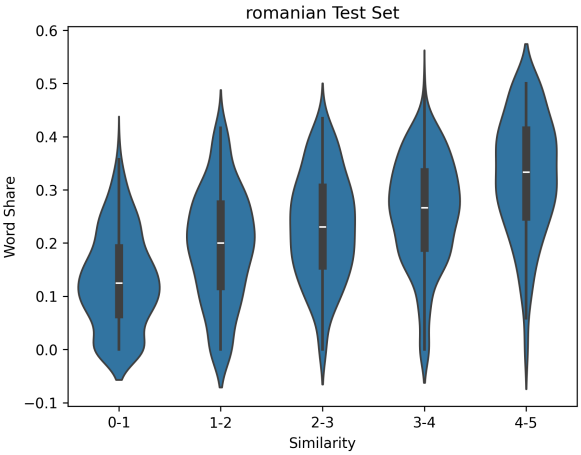
(1) Similarity distribution of Ro train set



(2) Similarity distribution of Ro test set

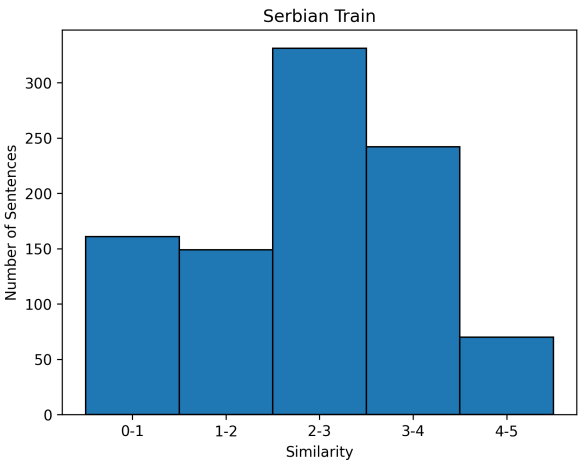


(3) Word share against similarity bins in Ro train set

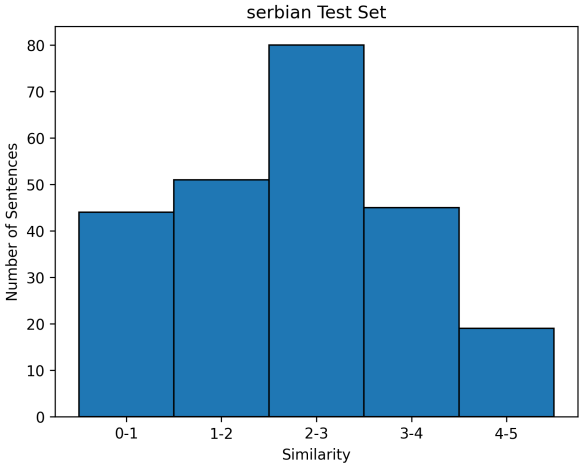


(4) Word share against similarity bins in Ro test set

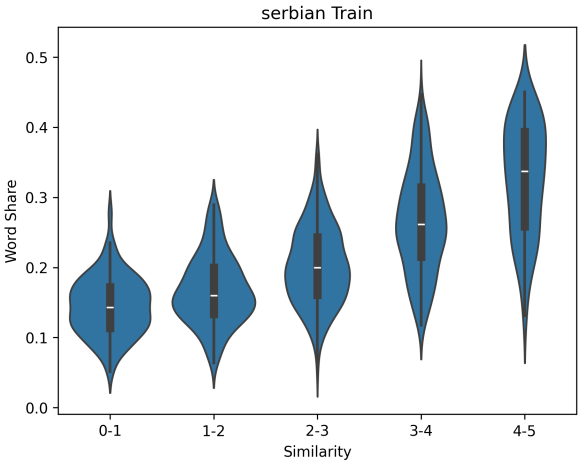
Serbian



(1) Similarity distribution of Se train set



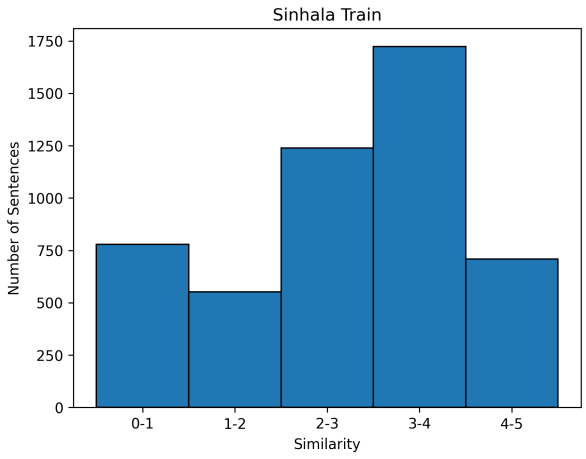
(2) Similarity distribution of Se test set



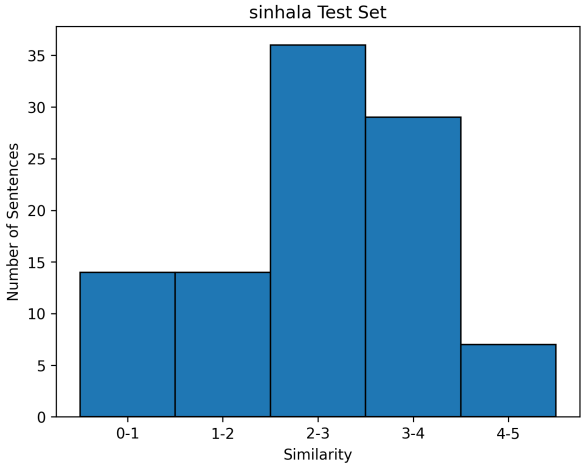
(3) Word share against similarity bins in Se train set

(4) Word share against similarity bins in Se test set

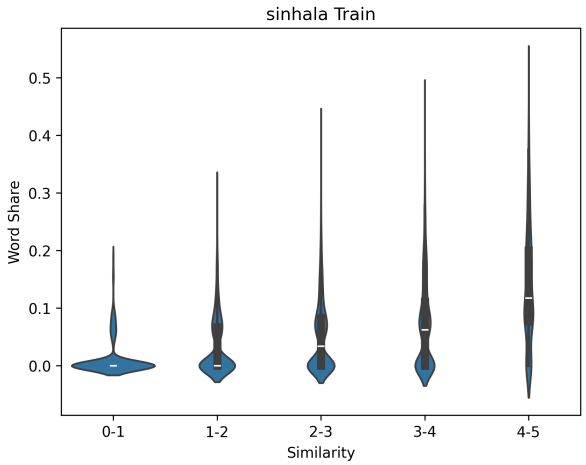
Sinhala



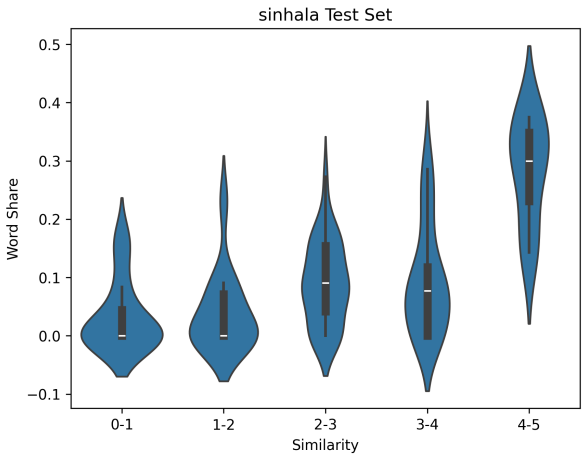
(1) Similarity distribution of Si train set



(2) Similarity distribution of Si test set

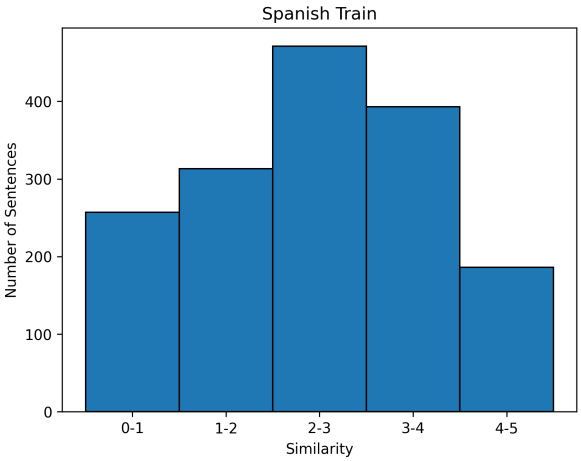


(3) Word share against similarity bins in Si train set

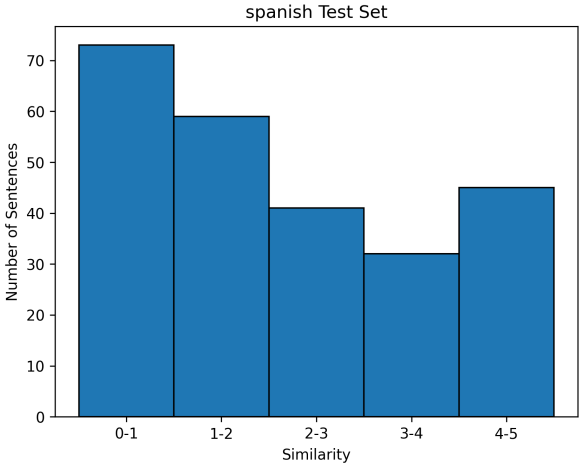


(4) Word share against similarity bins in Si test set

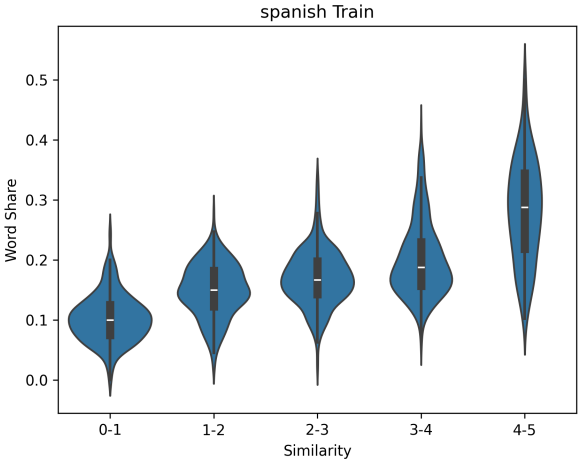
Spaish



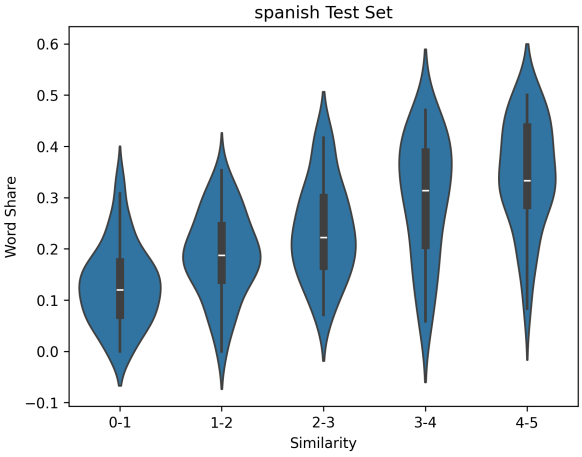
(1) Similarity distribution of Es train set



(2) Similarity distribution of Es test set

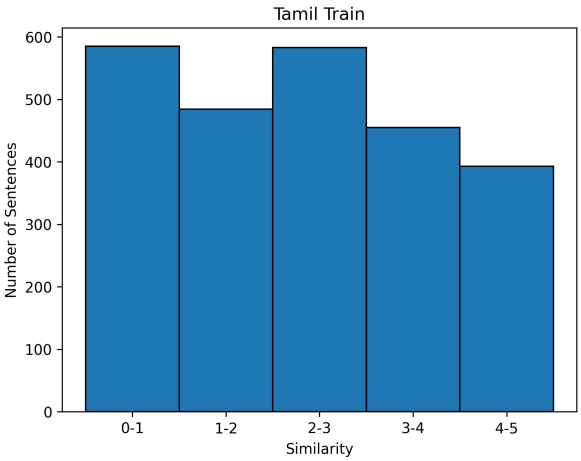


(3) Word share against similarity bins in Es train set

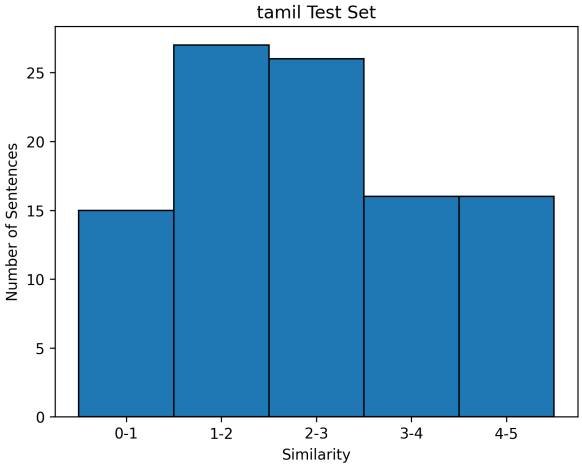


(4) Word share against similarity bins in Es test set

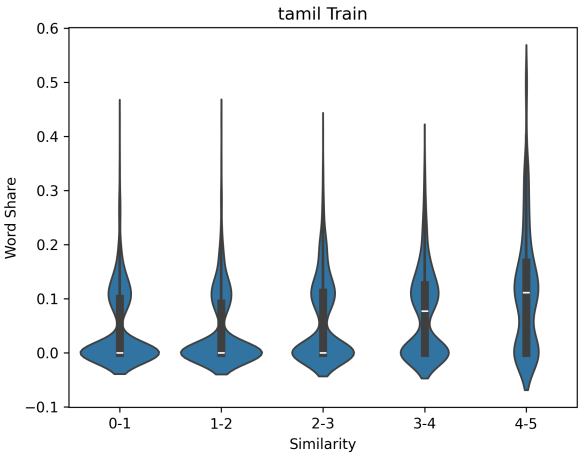
Tamil



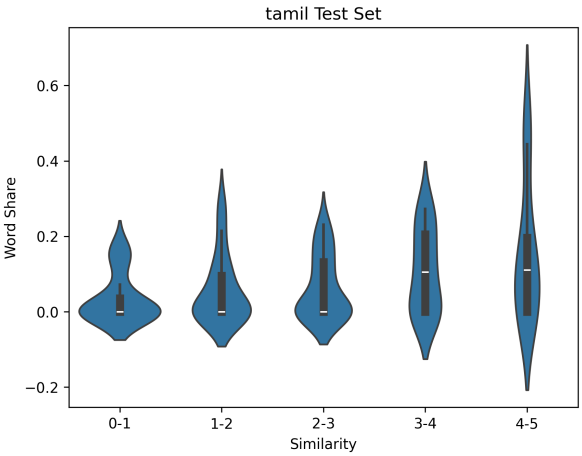
(1) Similarity distribution of Ta train set



(2) Similarity distribution of Ta test set



(3) Word share against similarity bins in Ta train set



(4) Word share against similarity bins in Ta test set