Unveiling Semantic Information in Sentence Embeddings

Leixin Zhang[2=], **David Burian**[1=], **Vojtěch John**[1], **Ondřej Bojar**[1]

[1]Faculty of Mathematics and Physics, Charles University
[2]University of Tübingen, Germany
leixin.zh@gmail.com, david.burian@me.com,
vojtik.john@seznam.cz, bojar@ufal.mff.cuni.cz

Abstract

This study evaluates the extent to which semantic information is preserved within sentence embeddings generated from state-of-art sentence embedding models: SBERT and LaBSE. Specifically, we analyzed 13 semantic attributes in sentence embeddings. Our findings indicate that some semantic features (such as tense-related classes) can be decoded from the representation of sentence embeddings. Additionally, we discover the limitation of the current sentence embedding models: inferring meaning beyond the lexical level has proven to be difficult.

Keywords: sentence embedding, transformation vector, semantic information

1. Introduction

Word embeddings have frequently been used as input in deep neural networks. Sentence embeddings are supposed to encapsulate sentence meanings into vectors. However, representing an entire sentence as a vector of fixed length poses significant challenges. Obtaining sentence embeddings is not as straightforward as extracting word embeddings based on contextual information from text. Embeddings merely based on surrounding text can be less representative at the sentence level.

Additionally, evaluating the quality of sentence embeddings or assessing whether these embeddings effectively encapsulate the meanings of sentences often requires a human-annotated corpus with well-defined semantic categories or sentence similarity scores.

In this study, we convert Czech sentences in the COSTRA dataset into sentence embeddings using SBERT and LaBSE models. COSTRA dataset (Barančíková and Bojar, 2020) is a collection of Czech sentences with semantic labels. Each set consists of a 'seed' sentence and transformation sentences that are derived from the seeds. The objective of this study is to assess whether sentence embeddings trained by SBERT and LaBSE retain semantic information and whether vectors in the same transformation class (with some similarity in semantics) show affinity in high dimensional space, which is tested by using clustering and classification algorithms to investigate whether vectors from the same class can be distinguished from vectors of other classes in high dimensional space.

The content of our paper is structured as follows: Section 3 presents a detailed introduction to the COSTRA dataset and an overview of our evaluation methods. In Section 4, we implement the dimension reduction technique to visualize sentence embeddings in 2D graphs. Section 5 attempts to predict new sentence embeddings with extracted transformation vectors. Section 6 implements cluster separation tests to assess within-class cohesion and between-class separation for 13 transformation classes. In Section 7, supervised methods are employed to train and predict transformation labels. Finally, Section 8 compares the results in all evaluation tasks and discusses the separability of transformation vectors.

2. Previous Studies

In this section, we introduce previous research on sentence embeddings, as well as the evaluation methods employed for assessing sentence embeddings.

2.1. Previous Studies on Sentence Embeddings

Word embeddings represent word meanings in space, and sentence embeddings are supposed to encapsulate sentence meanings into vectors, ideally of fixed lengths. There are two approaches to generating sentence embeddings. One is unsupervised learning of sentence embeddings. For instance, Yang et al. (2018) and Arora et al.

[=] Authors with equal contribution.

5th Workshop on Designing Meaning Representations (DMR 2024) @LREC-COLING-2024, pages 39–47

Class	**Description**	**Example** (Translated from Czech)
seed	original sentence	*Four members of my family lost their lives.*
ban	negative imperative	*Four members of my family cannot lose their lives!*
possibility	possibility modality	*Four members of my family probably lost their lives.*
past	past tense	*In those days, four members of my family lost their lives.*
future	future tense	*Four members of my family will one day lose their lives.*
opposite meaning	opposite sense	*Four members of my family were born.*
generalization	make it more general	*Four people died.*
minimal change	minimal alteration	*Four members of that family lost their lives.*
nonsense	by shuffling words	*Life lost members of my family.*
different meaning	by shuffling words	*Four members of my family lost a member.*
formal sentence	a more formal style	*Four members of my family closed their eyes forever.*
simple sentence	a simplistic style	*Four people of my family died.*
nonstandard	a colloquial style	*Almost my whole family died there.*
paraphrase	paraphrase	*Four of my relatives died.*

Table 1: Seed and Transformation classes in COSTRA

(2019) proposed an unsupervised method to construct sentence embeddings. They calculate the weighted sum of word embeddings[1] and then remove principal components to enhance embedding quality.

Nevertheless, the dominant method in prior research for generating sentence embeddings is supervised learning towards the relations (e.g. natural language inference, Conneau et al., 2017) we want to get from the embeddings.

The sequence-to-sequence architecture was used to generate sentence embeddings in machine translation tasks, with the encoder's output serving as the sentence representation. LASER (Artetxe and Schwenk, 2019) is an instance. It is a multilingual LSTM-based encoder-decoder model trained on parallel corpora across 93 languages (Goswami et al., 2021). However, it is challenged due to the suboptimal semantic representation. Reimers and Gurevych (2020) state that LASER fails in assessing the similarity of sentence pairs, despite its good performance in identifying exact translations.

More recently, transformer and BERT-based models have received increased attention. SBERT (Reimers and Gurevych, 2019) stands as a state-of-the-art model for generating sentence embeddings (Ham and Kim, 2021). Multilingual models have also been studied in recent years. Reimers and Gurevych (2020) fine-tune the monolingual SBERT model (Reimers and Gurevych, 2019) with a parallel corpus that includes 50 languages and leveraged knowledge distillations. Chidambaram et al. (2019) propose mUSE (Multilingual Universal Sentence Encoder), trained on parallel data in 16 languages. LaBSE (Feng et al., 2022) is another multilingual BERT-based model, trained on a dual encoder with 6 billion sentence translation pairs across 109 languages. These three multilingual models have demonstrated strong performance in previous studies (Devine et al., 2021; Reimers and Gurevych, 2020; Ham and Kim, 2021). In our study, we use SBERT and LaBSE, two models that support the Czech language to generate sentence embeddings.

[1]The actual deep learning tasks in which the word embeddings obtained can vary, such as autoregressive (e.g. LSTM) or non-autoregressive language modelling.

2.2. Sentence Embedding Evaluation

The evaluation of sentence embeddings in previous studies includes linguistic probing tests, semantic similarity tests, and other downstream classification tests (Conneau and Kiela, 2018).

Linguistic probing tasks start with investigating surface information, like decoding sentence lengths or assessing whether the original words can be detected from a sentence embedding (Adi et al., 2016). The syntactic evaluation examines whether sentence embeddings can detect neighbouring word shifts, part of speech tags, coordination inversion, number or gender agreement, depth of the syntactic tree, etc. (Perone et al., 2018; Pimentel et al., 2020; Hupkes et al., 2018). Other downstream classification tasks involve sentiment analysis and opinion polarity (Perone et al., 2018, Conneau et al., 2018).

The semantic similarity test is also popular in sentence embedding evaluation. Models are assessed by computing the correlation between the human-labeled similarity scores of sentence pairs and the model-predicted distance (e.g. cosine dis-

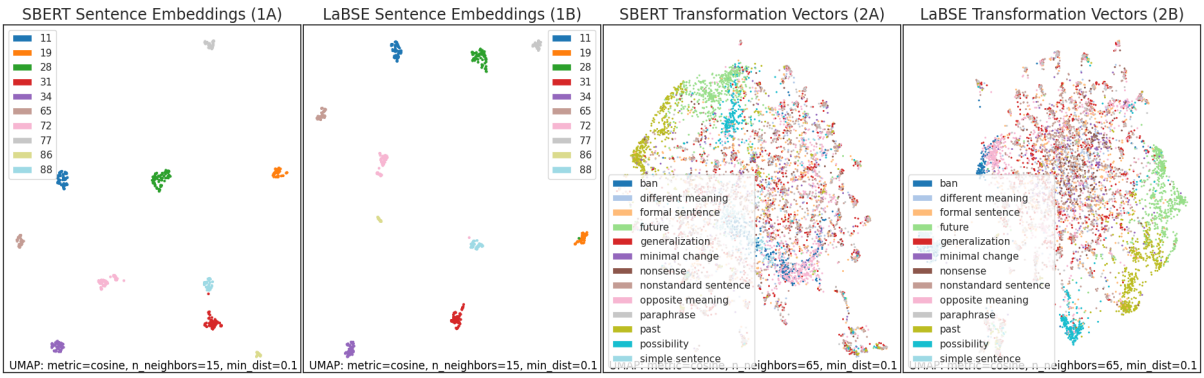

Figure 1: Visualization of sentence embeddings (1A) & (1B) and transformation vectors (2A) & (2B). (1A) & (1B) illustrate sentence embeddings of 10 randomly selected seeds and their corresponding transformed sentences. Each set of a seed sentence and its derived sentences is indicated by a seed index and represented with a distinct colour.

tance) of two sentence embeddings.

However, many semantic studies on sentence embeddings often fall short in providing insights into instances where models consistently underperform. Our research adopts a novel approach, potentially serving as a controlled experiment. By maintaining consistency in the seed sentences' information while altering only specific features in 13 classes, our research offers advantages in examining embedding transformations in detail.

3. Dataset and Sentence Embeddings

COSTRA (Barančíková and Bojar, 2020) is the evaluation dataset in our study. It comprises 6,968 Czech sentences, out of which 126 are seed sentences. The remaining sentences are transformation sentences derived from the seed sentences. These transformation sentences are categorized into 13 classes. Table 1 presents the descriptions of the 13 transformation classes and example sentences translated from the Czech COSTRA dataset.

In our study, we use SBERT[2] and LaBSE, two multilingual models with Czech language support to generate sentence embeddings. We differentiate two types of vectors: sentence embeddings and transformation vectors. **Sentence embeddings** are generated directly from SBERT and LaBSE models. **Transformation vectors** are vectors with their corresponding seed embeddings subtracted, in order to remove additional information from the seed sentence. In other words, given a transformed e.g. generalized sentence (with its embedding denoted as $generalization_i$ for short), we also consider the corresponding seed sentence (with the seed embedding denoted as $seed_i$) The transformation vector of this sentence pair is represented as $generalization_i$ - $seed_i$.

In the following sections, we aim to study whether transformation vectors in one class demonstrate a clustering tendency (within class cohesion) and whether they can be distinguished from transformation classes of other types (between-class separation).

4. Dimension Reduction and Visualization

This section presents a preliminary study of sentence embeddings and transformation vectors through dimension reduction and visualization. UMAP (Uniform Manifold Approximation and Projection) (McInnes et al., 2018) was employed as our dimension reduction technique and visualization tool.[3]

Firstly, we explore the spatial distribution of the sentence embeddings. Our assumption is that a seed sentence, sharing more identical words with its derived sentences, may lead to closer proximity to its transformed sentences than sentences belonging to other seed sets. To test the hypothesis, we randomly visualize 10 seed sentences along with sentences that are derived from them. Secondly, our analysis aims to explore whether transformation vectors (obtained by subtracting seed embeddings from their sentence embeddings) within the same transformation class (e.g. future transformation vectors) tend to group together.

Sentence embeddings from SBERT and LaBSE are depicted in Figure 1 (1A) & (1B). Each set

[2]To produce SBERT Sentence embeddings we used pre-trained multilingual model '*paraphrase-multilingual-MiniLM-L12-v2*'.

[3]PCA and T-SNE are also tested in the initial experiments, while the performance is much worse than UMAP, thus not presented in the paper.

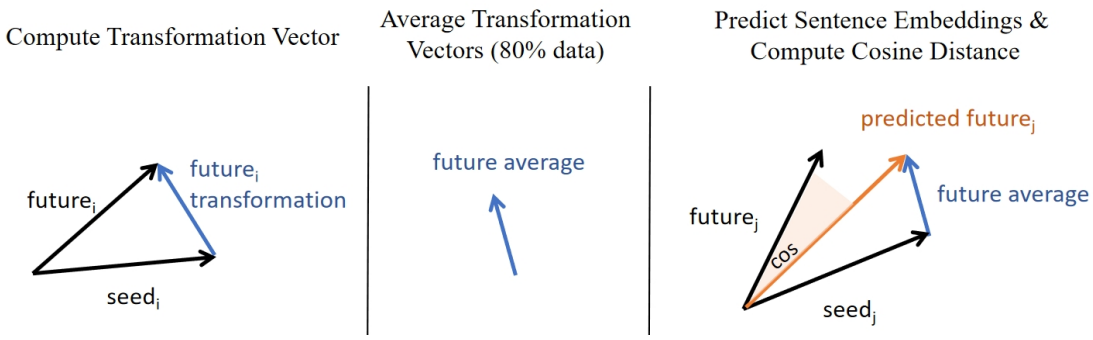

Figure 2: Cosine Similarity Computation between True and Predicted Sentence Embeddings

of sentence embeddings (the seed sentence and sentences derived from it) generally forms a cluster, suggesting that sentences tend to be situated close to their seed sentences.

In the results in Figure 1 (2A) & (2B), the tendency of the transformation vectors of the same class clustering together is observed only for certain classes, particularly tense-related classes ('past' and 'future'). Some classes form a cluster with only a part of the sentences, such as 'opposite meaning' and 'simple sentences'. However, transformation vectors of other classes (e.g. 'nonstandard sentence' and 'generalization') are dispersed across the space.

Additionally, it is worth noting that despite the different model architectures, and different lengths/dimensions of sentence embeddings of SBERT and LaBSE, their visualization results after the dimension reduction display comparable behaviour.

5. Predictive Capacity of Transformation Vectors

Section 4 demonstrates that transformation vectors in some (though not all) transformation classes are grouped together after dimension reduction. This section further evaluates the potential of transformation vectors to predict other sentence embeddings based on their seed embeddings. We assume the following property holds for transformation vectors: given a future-tense transformation vector ($future_i$ - $seed_i$), and the embedding of a different seed ($seed_j$), we can predict the embedding future_sentence$_j$ (sentence of its future tense) using Equation 1.

$$\mathrm{future_j} = \mathrm{future_i} - \mathrm{seed_i} + \mathrm{seed_j} \quad (1)$$

In the actual experiment, 80% of the sentences in each class are used to extract transformation vectors. We compute the average of the 80% transformation vectors to predict the sentence embeddings for the remaining 20% of the sentences (as shown in the illustration in Figure 2). The quality of transformation vectors is assessed using the cosine similarity between the predicted sentence embeddings and the true sentence embeddings.

class	SBERT	LaBSE
possibility	0.94	0.95
past	0.93	0.91
future	0.92	0.91
different meaning	0.91	0.91
nonsense	0.90	0.90
formal sentence	0.88	0.88
minimal change	0.87	0.92
ban	0.85	0.91
paraphrase	0.82	0.81
nonstandard sentence	0.81	0.82
simple sentence	0.81	0.79
opposite meaning	0.75	0.83
generalization	0.70	0.66

Table 2: Cosine Similarity of predicted embeddings and true derivation sentence embeddings

5.1. Cosine Distance between Predicted and True Embeddings

The results in Table 2 show that the majority of the transformation classes have a cosine similarity score above 0.8. These findings imply that a number of predicted vectors lie close to their true sentence embeddings, especially those in 'possibility' and 'past' classes, both with very high scores.

However, in contrast, the 'generalization' class exhibits the lowest score (0.70 in SBERT and 0.66 in LaBSE), falling below the baseline (ranging from 0.72 to 0.78), obtained by using the same dataset but with shuffled transformation labels within each seed set.

This could be attributed to the varying degrees

of transformation when a seed sentence is transformed into multiple generalization forms. If the transformation vectors do not align in a consistent vector direction, relying on the average of 80% of the vectors is inaccurate in predicting sentence embeddings. It is also worth mentioning that the cosine distance of the baseline with shuffled transformation labels reaches 0.72, suggesting that the embeddings of any arbitrary sentence and the arbitrary transformation of the sentence are close to each other.

5.2. Cosine Distance across Classes

To deal with the aforementioned challenge of varying transformation degrees within a class and the limitation of assessing transformation vectors solely relying on cosine distance from their true embeddings, we extend our assessment to the cosine distance of predicted sentence embeddings with actual embeddings across 13 classes.

Our underlying assumption is that although transformation vectors with varying degrees might not exhibit a consistent vector direction in space, transformation vectors in one class may still be restricted within a region that is distinguishable from the regions of other transformation classes. As a result, predicted sentence embeddings should show the highest cosine similarity with sentence embeddings of the target class, compared to those from other classes. For example, the sentence embedding predicted by the 'generalization' transformation vector, is compared with the true embedding *generalization*$_i$ (with the assumed highest cosine similarity), as well as with sentence embeddings of other classes derived from *seed*$_i$, such as *past*$_i$, *ban*$_i$, *nonsense*$_i$, etc. (with an assumed lower cosine similarity).

Figure 3 displays the results of the comparison across classes. Each row is normalized using min-max normalization. Darker hues indicate closer to 1, while lighter hues indicate scores near 0. We call it normalized predictability score, measuring how well the embeddings of the target classes are predicted from the transformation vectors of the source class.

The results suggest that the diagonal cells typically get the darkest hue and the remaining cells in the same row often display lighter shades. It implies a generally higher cosine similarity between the predicted embeddings and the actual embeddings of the target class compared to embeddings of other classes. In particular, the sentence embedding of 'ban' is the best-predicted class, although its cosine similarity score discussed in Section 5.1 does not rank high among the 13 classes.

However, the predictability varies across transformation classes. In the results of SBERT, the predictions of four classes ('different meaning', 'minimal change', 'non-sense', and 'paraphrase') display the highest cosine similarity scores with embeddings in a different class. For instance, the predicted embeddings of 'different meaning' show the highest cosine similarity with 'minimal change' embeddings, while the predicted 'non-sense' embeddings correlate most strongly with the true embeddings of 'different meaning'. Additionally, the cosine similarity values of the 'formal sentence' and 'simple sentence' classes are not sufficiently distinguished from the values of other classes.

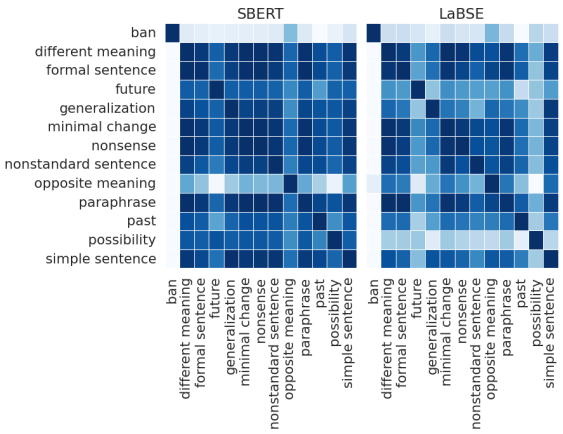

Figure 3: Cosine similarity between true and predicted embeddings. (Each row is normalized with min-max normalization. Darker hues indicate scores closer to 1, while lighter hues indicate scores near 0.)

We note that LaBSE outperforms SBERT in this experiment. There is only one instance of incongruence: predicted 'paraphrase' embeddings exhibit the highest cosine similarity with sentence embeddings of 'different meaning'. The generally better performance of LaBSE can also be observed in Figure 3.

6. Cluster Separation Test

This section analyzes whether the transformation vectors of the same class cluster together and are separated from other classes in space. We present a cluster separation test using the Calinski-Harabasz index.

$$\mathrm{CH} = \left[\frac{\sum_{k=1}^{K} n_k \|c_k - c\|^2}{K-1}\right] / \left[\frac{\sum_{k=1}^{K}\sum_{i=1}^{n_k} \|d_i - c_k\|^2}{N-K}\right] \quad (2)$$

The Calinski-Harabasz index[4] (Equation 2) measures the ratio of between-cluster dispersion to

[4] K means the number of clusters; n_k is the number of points in k_{th} cluster; c_k represents the number of points and centroid of the k_{th} cluster; c is the global centroid; N is the total number of data points.

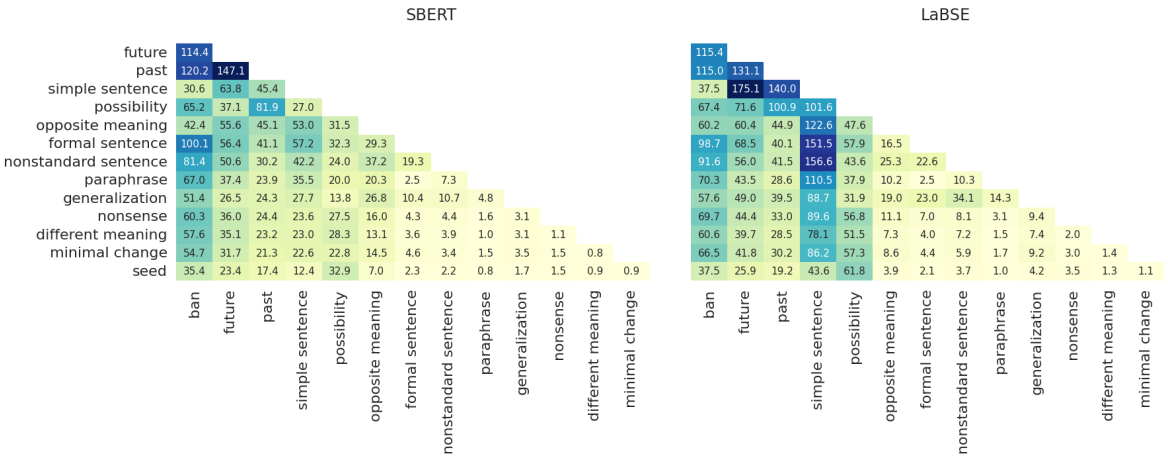

Figure 4: Pairwise Calinski-Harabasz index of transformation vectors from SBERT and LaBSE.

SBERT	LaBSE	mixSBERT	mixLaBSE
28.415	44.885	0.563	0.565

Table 3: Cluster separation test on 13 classes

inter-cluster dispersion. A higher value signifies well-separated clusters (Caliński and Harabasz, 1974).

In this study, we compute CH-Index in two ways. Firstly, we compare the performance of the two models by assessing transformation vectors in the 13 classes. Secondly, we conduct a pairwise test to assess the degree of separation of transformation classes in pairs.

We establish benchmarks for the CH Index by mixing up the transformation labels of the dataset. The CH index scores for 13 classes are shown in Table 3. LaBSE has a better performance than SBERT. Nevertheless, both models significantly outperform the baselines. Figure 4 presents the results of pairwise testing. Two baselines of mixed transformation labels have CH index values ranging from 0.392 to 0.899 for SBERT, and from 0.332 to 1.556 for LaBSE.

We observed that 'ban' and 'future' generally exhibit higher values, suggesting their better separation from other classes and within-class cohesion. In the results of LaBSE model, 'simple sentence' is the class with the highest CH-index scores, followed by 'ban', 'future' and 'possibility'. While for SBERT, the advantages of 'simple sentence' and 'possibility' classes are not observed. It indicates the discrepancies in the distribution patterns of transformation vectors in space obtained from SBERT and LaBSE.

Additionally, pairwise tests also show that other classes such as 'different meaning', 'minimal change' and 'paraphrase' often fall below the benchmark in both SBERT and LaBSE, suggesting insufficient separability of their transformation vectors in these classes.

7. Classification Task

In previous experiments, we utilized methods such as visualization, sentence embedding prediction, and clustering separation to assess the quality of transformation vectors from SBERT and LaBSE. This section introduces supervised methods to investigate whether transformation vectors can be decoded to predict transformation labels.

The classifiers used in our experiments consist of Random Forests, Support Vector Machine (SVM), and K-Nearest Neighbors (KNN). Depending on their unique strengths, these classifiers may decode transformation vectors in distinct ways. Random Forests use specific criteria and feature-based splitting to classify data (Breiman, 2001; Cutler et al., 2012). SVM has the ability to map inputs into high-dimensional spaces using the kernel trick (Schölkopf et al., 1999; Smola and Schölkopf, 2004). KNN adopts a local distance-based approach and assigns labels based on the known labels of neighbouring data points. We intend to investigate the potential of these diverse methods to extract semantic information (transformation labels) from transformation vectors.

In addition to the sentence embeddings from SBERT and LaBSE, we also generated TF-IDF weighted encoding of all vocabulary in COSTRA. The additional TF-IDF embeddings aim to assess the influence of lexical factors on classification performance. In other words, we aim to test whether certain words are unique to a particular transformation class, thereby potentially enhancing the prediction accuracy. Similarly to other tasks in our study, we use the mixed-up SBERT as our baseline.

The results in Figure 5 indicate high F1 scores

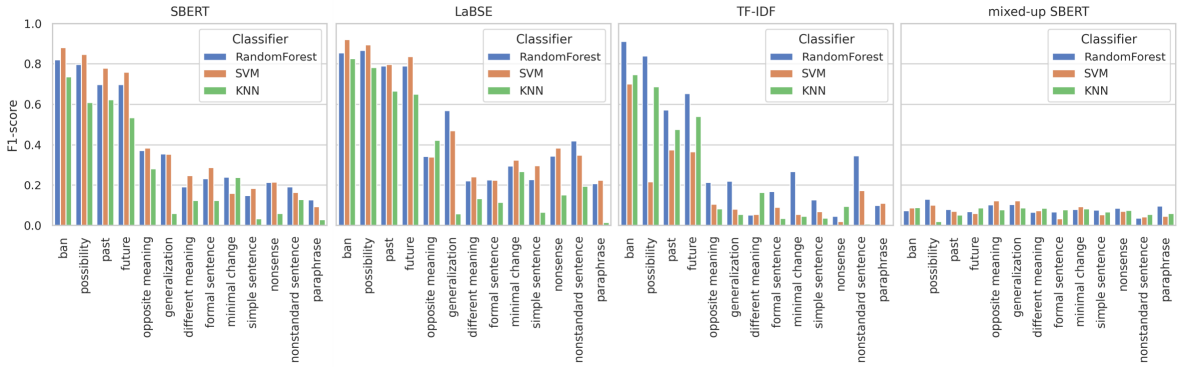

Figure 5: F1-scores for transformation label prediction

for four transformation classes: 'ban', 'possibility', 'past', and 'future'. The comparably high F1 score of TF-IDF embeddings suggests the substantial impact of the lexical factor on the predictability of these classes. In other words, sentences in these four classes tend to contain particular words that are unique to a class, contributing to their superior predictability.

Additionally, 'generalization' from LaBSE exhibits F1 scores higher than those of SBERT and TF-IDF. It on the one hand suggests that LaBSE outperforms SBERT in these two instances. On the other hand, it also implies that LaBSE may have a better ability to capture semantic information beyond the word level.

8. Discussion

In this section, we compare the results of the evaluation tasks implemented in our study and then discuss the separability of transformation vectors and to what extent the semantic features can be decoded from sentence embeddings.

8.1. Summary of Results in Evaluation Tasks

Transformation vectors in four transformation classes ('ban', 'possibility', 'past', and 'future') demonstrate good performance in almost all evaluation tasks: dimension reduction & visualization, sentence embedding prediction, cluster separation, and classification, and show consistent results in both models. This is in line with their pronounced separability from other classes. In contrast, some classes exhibit weak performance in almost all evaluation tasks, for instance, 'paraphrase', 'minimal change', 'formal sentence', and 'nonsense'.

Nevertheless, certain classes display varying performance across our four evaluation tasks and two models. For example, the LaBSE transformation vectors in the 'simple sentence' class excel in the sentence embedding prediction task (Figure 3) and the cluster separation test (Figure 4), but not in the classification task as shown in Figure 5.

The dimension reduction and visualization techniques may provide insight to speculate the reasons for such variations. Figure 1 displays that the clusters of the 'opposite meaning' and 'simple sentence' classes are formed only by some of the vectors in these two classes. The remaining data points within these two classes are dispersed throughout the space. This property (some data gathered together but some dispersed in space for a class) introduces complexity when assessing their separability with a single value in evaluation tests. Different evaluation methods may emphasize distinct properties of the vectors in a class and decode them in different manners. This could provide insight into the observed variations in performance for these classes across different evaluation tasks.

This analysis also suggests that while dimension reduction is criticized for the loss of information in high-dimensional spaces, it can instead offer supplementary insights when combined with visualization.

8.2. Separability Analysis

In the section above, we discussed that transformation vectors in some classes are not separable from others. It could be attributed to at least two factors. One factor is the inherent difficulty in distinguishing these classes from the rest, while the other factor is related to the limitations of the models themselves.

We notice that certain classes are inherently challenging to separate. For instance, sentences in the 'minimal change' class are less distinguishable from those in the 'different meaning' class. 'Paraphrase' is less distinguishable from 'simple sentence', 'formal sentence' and 'nonstandard sentence', simply because all of them are also a form of a paraphrase. The models' poor performance in evaluation tests may potentially correspond to the uncertainty inherent in human judgment. In other words, these classes might also pose difficulties in

differentiation even for human assessors.

The second reason for weak performance in some tests lies in the models' limitations in capturing semantic information. For example, both models show relatively low prediction accuracy for 'nonsense' and 'opposite meaning' (with F1 for 'nonsense' < 0.4; 'opposite meaning' < 0.5), two types that are easy to detect for human assessors.

The good classification results of TF-IDF embeddings also reveal that the separability of classes can to a considerable extent stem from purely lexical factors. This observation suggests that inferring meaning beyond the lexical level is difficult for the two models, and sentence embeddings generated by current models lack a comprehensive representation of sentence meaning.

9. Conclusion

Our study analyzed sentence embeddings generated from two multilingual models: SBERT and LaBSE, evaluating using the Czech COSTRA dataset to test whether some semantic information is preserved and can be decoded from sentence embeddings.

Our visualization firstly demonstrates that transformation sentences are situated in proximity to their respective seed sentences in the vector space. To assess the semantic attributes of 13 transformation classes exemplified in the COSTRA dataset, we examined transformation vectors, obtained by subtracting seed embeddings from sentence embeddings to eliminate the original seed sentence information.

In addition to dimension reduction and visualization, we conducted three other evaluation tasks: sentence embedding prediction, cluster separation, and transformation label classification. Our findings indicate that both models exhibit comparable performance, with LaBSE slightly outperforming SBERT in certain evaluation tasks.

Furthermore, our analysis highlights that transformation vectors for some classes show better separability from other classes and reach better evaluation scores in evaluation tasks. However, the good outcome may be attributed to specific words that are exclusive to a particular class, as suggested by similarly good results obtained using simple TF-IDF. Although the lower performance observed in other transformation types may be due to their inherent difficulty in class detection, the limitations of the current models are not negligible: inferring meaning beyond the lexical level has proven to be challenging for them.

10. Acknowledgements

This work was partially supported by GAČR EXPRO grant NEUREM3 (19-26934X) and by the Grant Agency of Charles University in Prague (GAUK 244523).

11. Bibliographical References

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2016. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. *arXiv preprint arXiv:1608.04207*.

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2019. A simple but tough-to-beat baseline for sentence embeddings. 5th International Conference on Learning Representations, ICLR 2017 ; Conference date: 24-04-2017 Through 26-04-2017.

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Petra Barančíková and Ondřej Bojar. 2020. Costra 1.1: An inquiry into geometric properties of sentence spaces. In *Text, Speech, and Dialogue: 23rd International Conference, TSD 2020, Brno, Czech Republic, September 8–11, 2020, Proceedings*, pages 135–143. Springer.

Leo Breiman. 2001. Random forests. *Machine learning*, 45:5–32.

Tadeusz Caliński and Jerzy Harabasz. 1974. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27.

Muthu Chidambaram, Yinfei Yang, Daniel Cer, Steve Yuan, Yunhsuan Sung, Brian Strope, and Ray Kurzweil. 2019. Learning cross-lingual sentence representations via a multi-task dual-encoder model. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pages 250–259, Florence, Italy. Association for Computational Linguistics.

Alexis Conneau and Douwe Kiela. 2018. SentEval: An evaluation toolkit for universal sentence representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark. Association for Computational Linguistics.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia. Association for Computational Linguistics.

Adele Cutler, D Richard Cutler, and John R Stevens. 2012. Random forests. *Ensemble machine learning: Methods and applications*, pages 157–175.

Peter Devine, Yun Sing Koh, and Kelly Blincoe. 2021. Evaluating unsupervised text embeddings on software user feedback. In *2021 IEEE 29th International Requirements Engineering Conference Workshops (REW)*, pages 87–95.

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.

Koustava Goswami, Sourav Dutta, Haytham Assem, Theodorus Fransen, and John P. McCrae. 2021. Cross-lingual sentence embedding using multi-task learning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9099–9113, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jiyeon Ham and Eun-Sol Kim. 2021. Semantic alignment with calibrated similarity for multilingual sentence embedding. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1781–1791, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Dieuwke Hupkes, Sara Veldhoen, and Willem Zuidema. 2018. Visualisation and'diagnostic classifiers' reveal how recurrent and recursive neural networks process hierarchical structure. *Journal of Artificial Intelligence Research*, 61:907–926.

Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

Christian Samuel Perone, Roberto Silveira, and Thomas S. Paula. 2018. Evaluation of sentence embeddings in downstream and linguistic probing tasks. *ArXiv*, abs/1806.06259.

Tiago Pimentel, Josef Valvoda, Rowan Hall Maudslay, Ran Zmigrod, Adina Williams, and Ryan Cotterell. 2020. Information-theoretic probing for linguistic structure. *arXiv preprint arXiv:2004.03061*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online. Association for Computational Linguistics.

Bernhard Schölkopf, Christopher JC Burges, Alexander J Smola, et al. 1999. *Advances in kernel methods: support vector learning*. MIT press.

Alex J Smola and Bernhard Schölkopf. 2004. A tutorial on support vector regression. *Statistics and computing*, 14:199–222.

Ziyi Yang, Chenguang Zhu, and Weizhu Chen. 2018. Parameter-free sentence embedding via orthogonal basis. In *Conference on Empirical Methods in Natural Language Processing*.