Discrete Cosine Transform as Universal Sentence Encoder

Nada Almarwani [1,2] and **Mona Diab** [1,3]
[1] Dep. of Computer Science, The George Washington University
[2] Dep. of Computer Science, College of Computer Science and Engineering, Taibah University
[3] Facebook AI Research
nmarwani@taibah.edu.sa, mdiab@fb.com

Abstract

Modern sentence encoders are used to generate dense vector representations that capture the underlying linguistic characteristics for a sequence of words, including phrases, sentences, or paragraphs. These kinds of representations are ideal for training a classifier for an end task such as sentiment analysis, question answering and text classification. Different models have been proposed to efficiently generate general purpose sentence representations to be used in pretraining protocols. While averaging is the most commonly used efficient sentence encoder, Discrete Cosine Transform (DCT) was recently proposed as an alternative that captures the underlying syntactic characteristics of a given text without compromising practical efficiency compared to averaging. However, as with most other sentence encoders, the DCT sentence encoder was only evaluated in English. To this end, we utilize DCT encoder to generate universal sentence representation for different languages such as German, French, Spanish and Russian. The experimental results clearly show the superior effectiveness of DCT encoding in which consistent performance improvements are achieved over strong baselines on multiple standardized datasets.

1 Introduction

Recently, a number of sentence encoding representations have been developed to accommodate the need of sentence-level understanding; some of these models are discussed in (Hill et al., 2016; Logeswaran and Lee, 2018; Conneau et al., 2017), yet most of these representations have focused on English only.

To generate sentence representations in different languages, the most obvious solution is to train monolingual sentence encoders for each language. However, training a heavily parameterized monolingual sentence encoder for every language is inefficient and computationally expensive, let alone the impact on the environment. Thus, utilizing a non-parameterized model with ready-to-use word embeddings is an efficient alternative to generate sentence representations in various languages.

A number of non-parameterized models have been proposed to derive sentence representations from pre-trained word embeddings (Rücklé et al., 2018; Yang et al., 2019; Kayal and Tsatsaronis, 2019). However, most of these models, including averaging, disregard structure information, which is an important aspect of any given language. Recently, Almarwani et al. (2019) proposed a structure-sensitive sentence encoder, which utilizes Discrete Cosine Transform (DCT) as an efficient alternative to averaging. The authors show that this approach is versatile and scalable because it relies only on word embeddings, which can be easily obtained from large unlabeled data. Hence, in principle, this approach can be adapted to different languages. Furthermore, having an efficient, ready-to-use language-independent sentence encoder can enable knowledge transfer between different languages in cross-lingual settings, empowering the development of efficient and performant NLP models for low-resource languages.

In this paper, we empirically investigate the generality of DCT representations across languages as both a single language model and a cross-lingual model in order to assess the effectiveness of DCT across different languages.

2 DCT as sentence Encoder

In signal processing domain DCT is used to decompose signal into component frequencies revealing dynamics that make up the signal and transitions within (Shu et al., 2017). Recently, DCT has been adopted as a way to compress textual information

*Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics
and the 11th International Joint Conference on Natural Language Processing (Short Papers)*, pages 419–426
August 1–6, 2021. ©2021 Association for Computational Linguistics

(Kayal and Tsatsaronis, 2019; Almarwani et al., 2019). A key observation in NLP is that word vectors obey laws of algebra King – Man + Woman = (approx.) Queen (Mikolov et al., 2013). Thus, given word embeddings, cast a sentence as a multi-dimensional signal over time, in which DCT is used to summarize the general feature patterns in word sequences and compress them into fixed-length vectors (Kayal and Tsatsaronis, 2019; Almarwani et al., 2019).

Mathematically, DCT is an invertible function that maps an input sequence of N real numbers to the coefficients of N orthogonal cosine basis functions of increasing frequencies (Ahmed et al., 1974). The DCT components are arranged in order of significance. The first coefficient (c[0]) represents the sum of the input sequence normalized by the square length, which is proportional to the average of the sequence (Ahmed et al., 1974). The lower-order coefficients represent lower signal frequencies which correspond to the overall patterns in the sequence. For example, DCT is used for compression by preserving only the coefficients with large magnitudes. These coefficients can be used to reconstruct the original sequence exactly using the inverse transform (Watson, 1994).

In NLP, Kayal and Tsatsaronis (2019) applied DCT at the word level to reduce the dimensionality of the embeddings size, while Almarwani et al. (2019) applied it along the sentence length as a way to compress each feature in the embedding space independently. In both implementations, the top coefficients are concatenated to generate the final representation for a sentence. As shown in (Almarwani et al., 2019), applying DCT along the features in the embeddings space renders representations that yield better results. Also, Zhu and de Melo (2020) noted that similar to vector averaging the DCT model proposed by (Almarwani et al., 2019) yields better overall performance compared to more complex encoders, thus, in this work, we adopt their implementation to extract sentence-level representations.

Specifically, given a sentence matrix $N \times d$, a sequence of DCT coefficients $c[0], c[1], ..., c[K]$ are calculated by applying the DCT type II along the d-dimensional word embeddings, where $c[K] = \sqrt{\frac{2}{N}} \sum_{n=0}^{N-1} v_n \cos \frac{\pi}{N}(n + \frac{1}{2})K$ (Shao and Johnson, 2008). Finally, a fixed-length sentence vector of size Kd is generated by concatenating the first K DCT coefficients, which we refer to as $c[0 : K]$.[1]

Task	Description
SentLen	Length prediction
WC	Word Content analysis
BShift	Word order analysis
TreeDepth	Tree depth prediction
Tense	Verb tense prediction
CoordInv	Coordination Inversion
SubjNum	Subject number prediction
ObjNum	Object number prediction
SOMO	Semantic odd man out

Table 1: Probing Tasks as described in (Conneau et al., 2018; Ravishankar et al., 2019).

3 Multi-lingual DCT Embeddings

3.1 Experimental Setups and Results

In our study, DCT is used to learn a separate encoder for each language from existing monolingual word embeddings. To evaluate DCT embeddings across different languages, we used the probing benchmark provided by Ravishankar et al. (2019), which includes a set of multi-lingual probing datasets.[2] The benchmark covers five languages: English, French, German, Spanish and Russian, derived from Wikipedia. The task set comprises 9 probing tasks, summarized in Table 1, that address varieties of linguistic properties including surface, syntactic, and semantic information (Conneau et al., 2018; Ravishankar et al., 2019). Ravishankar et al. (2019) used the datasets to evaluate different sentence encoders trained by mapping sentence representations to English. Unlike Ravishankar et al. (2019), we use the datasets to evaluate DCT embeddings for each language independently. As a baseline, in addition to the DCT embeddings, we use vector averaging to extract sentence representations from the pre-trained embeddings.

For model evaluations, we utilize the SentEval framework introduced in (Conneau and Kiela, 2018). In all experiments, we use a single-layer MLP on top of DCT sentence embeddings with the following parameters: kfold=10, batch_size=128, nhid=50, optim=adam, tenacity=5, epoch_size=4.

[1] Unlike (Almarwani et al., 2019), we note no further improvements with larger coefficients, thus, we only report the results of $1 \leq K \leq 4$.

[2] Refer to (Conneau et al., 2018) and (Ravishankar et al., 2019) for more details about the probing tasks.

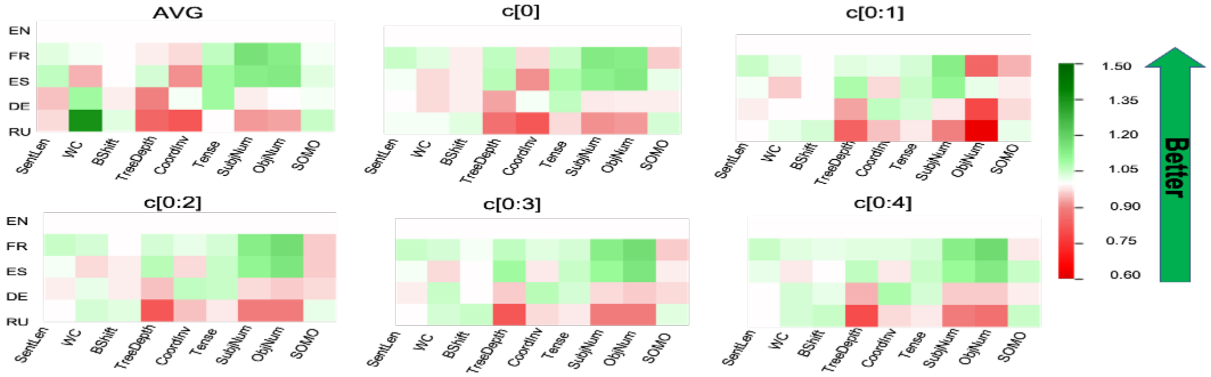

Figure 1: Results of the probing tasks comparing XX languages performance relative to English. White indicates a value of 1, demonstrating parity in performance with English. Red indicates better English performance while green indicates better XX Lang results.

For the word embeddings, we relied on the publicly available pre-trained FastText embeddings introduced in (Grave et al., 2018).[3]

Results: Figure 1 shows a heat-map reflecting the probing results of the different languages relative to English. Overall, French (FR) seems to be the closest to English (EN) followed by Spanish (ES) then German (DE) and then finally Russian (RU) across the various DCT coefficients. Higher coefficients reflect majority better performance across most tasks for FR, ES and DE. We see the most variation with worse results than English on the syntactic tasks of TreeDepth, CoordInv, Tense, SubjNum and ObjNum for RU. SOMO stands out for RU where it outperforms EN. The variation in Russian might be due to the nature of RU being a more complex language that is morphologically rich with flexible word order (Toldova et al., 2015).

In terms of the performance per number of DCT coefficients, we observe consistent performance gain across different languages that is similar to the English result trends. Specifically, for the surface level tasks, among the DCT models the $c[0]$ model significantly outperforms the AVG with an increase of $\sim$30 percentage points in all languages. The surface level tasks (SentLen and WC) show the most notable variance in performance, in which the highest results are obtained using the $c[0]$ model. However, the performance decreases in all languages when K is increased. On the other hand, for all languages, we observe a positive effect on the model's performance with larger K in both the syntactic and semantic tasks. The complete numerical results are presented in the Appendix in Table 5.

4 Cross-lingual Mapping based on DCT Encoding

4.1 Approach

Aldarmaki and Diab (2019)proposed sentence-level transformation approaches to learn context-aware representations for cross-lingual mappings. While the word-level cross-lingual transformations utilize an aligned dictionary of word embeddings to learn the mapping, the sentence-level transformations utilize a large dictionary of parallel sentence embeddings. Since sentences provide contexts that are useful for disambiguation for the individual word's specific meaning, sentence-level mapping yields a better cross-lingual representation compared to word-level mappings.

A simple model like sentence averaging can be used to learn transformations between two languages as shown in (Aldarmaki and Diab, 2019). However, the resulting vectors fail to capture structural information such as word order, which may result in poor cross-lingual alignment. Therefore, guided by the results shown in (Aldarmaki and Diab, 2019), we further utilize DCT to construct sentence representations for the sentence-level cross-lingual modeling.

4.2 Experiments Setups and Results

For model evaluation, we use the same cross-lingual evaluation framework introduced in (Aldarmaki and Diab, 2019). Intuitively, sentences tend to be clustered with their translations when their vectors exist in a well-aligned cross-lingual space. Thus, in this framework, cross-lingual mapping ap-

[3] Available at: https://fasttext.cc.

proaches are evaluated using sentence translation retrieval by calculating the accuracy of correct sentence retrieval. Formally, the cosine similarity is used to find the nearest neighbor for a given source sentence from the target side of the parallel corpus.

4.3 Evaluation Datasets and Results

To demonstrate the efficacy of cross-lingual mapping using the sentence-level representation generated by DCT models, similarly to Aldarmaki and Diab (2019), we used the WMT'13 data set that includes EN, ES and DE languages (Bojar et al., 2013). We further used five language pairs from the WMT'17 translation task to evaluate the effectiveness of DCT-based embeddings. Specifically, we used a sample of 1 million parallel sentences from WMT'13 common-crawl data; this subset is the same one used in (Aldarmaki and Diab, 2019).[4] To assess efficacy of the DCT models for the cross-lingual mapping, we reported the performances of the sentence translation retrieval task within the WMT'13 test set, which includes EN, ES, and DE as test languages (Bojar et al., 2013). Specifically, we first used the 1M parallel sentences for the alignment between source languages (ES and DE) to a target language (EN) independently. We evaluated the translation retrieval performance in all language directions, from source languages to English: ES-EN and DE-EN, as well as between the sources languages: ES-DE.

Similarly, we conduct a series of experiments on 5 different language pairs from the WMT'17 translation task, which includes DE, Latvian (LV), Finnish (FI), Czech (CS), and Russian (RU), each of which is associated with an English translation (Zhang et al., 2018).[5] For each language pair, we sampled 1M parallel sentences from their training corpus for the cross-lingual alignment between each source language and EN. Also, we used the test set available for each language pair to evaluate the translation retrieval performances.

In our experiments, we evaluate the translation retrieval performance in all language directions using three type of word embeddings: 1- a publicly available pre-trained word embeddings in which we show the performance of DCT against averaging, which we refer to hereafter as out-of-domain embeddings as shown in Table 2. 2- Also, we ran additional experiments in which we used a domain specific word embedding (that we trained on genre that is similar to the translation task) and 3-contextualized word embedding, which we refer to hereafter as in-domain embeddings as shown in Table 3.

Lang pair	AVG	$c[0]$	$c[0:1]$	$c[0:2]$	$c[0:3]$
Lang→EN					
ES→EN	65.67	64.87	71.26	**71.80**	70.13
DE→EN	51.80	50.30	57.23	**58.13**	56.57
RU→EN	45.22	52.75	61.91	**64.35**	63.33
CS→EN	41.87	42.50	52.89	54.99	**55.05**
FI→EN	40.46	42.00	47.57	**47.80**	46.16
LV→EN	21.26	40.13	51.42	56.37	**60.16**
EN→Lang					
EN→ES	69.97	69.50	73.73	**73.87**	71.73
EN→DE	67.50	66.23	**69.27**	68.70	65.83
EN→RU	38.09	44.29	54.73	59.51	**60.94**
EN→CS	39.73	40.40	50.99	54.00	**54.12**
EN→FI	39.34	42.52	51.67	**52.59**	51.74
EN→LV	15.83	33.55	47.08	53.22	**55.72**
Lang1→Lang2					
DE→ES	43.80	42.20	49.50	**51.20**	51.17
ES→DE	57.67	56.46	**60.53**	59.83	57.87

Table 2: Sentence translation retrieval accuracy based on out of domain pre-trained Fasttext embeddings. Arrows indicate the direction, with English (EN), Spanish (ES), German (DE), Russian (RU), Czech (CS), Finnish (FI) , Turkish (TR), and Latvian (LV).

Out-of-domain embeddings: For all language pairs, DCT-based models outperform AVG and c[0] models in the sentence translation retrieval task. In the direction $\rightarrow EN$, while the c[0:2] model achieve the highest accuracy for ES, DE, RU, and FI languages, the c[0:3] model achieved the highest accuracy for CS and LV languages. Specifically, the c[0:2] model yields increases of 5.59%-30% in the direction from source languages (ES, DE, RU, and FI) to English compared to the AVG model. Also, while the c[0:3] model yielded an increase of 13% gains over the baseline for CS, it provides the most notable increase of 38% for LV. For the opposite directions $EN \rightarrow source$, the DCT-based embeddings model also outperformed AVG and c[0] models. In particular, we observed accuracy gains of at least 3.81% points using more coefficients in DCT-based models compared to the AVG and c[0] models for all languages. A similar trend is observed in the zero-shot translation retrieval between the two non English languages (ES and DE), in which DCT-based models outperform the AVG and c[0] models.

[4] Evaluation scripts and WMT'13 dataset as described in (Aldarmaki and Diab, 2019) are available in https://github.com/h-aldarmaki/sent_translation_retrieval

[5] The pre-processed version of the WMT'17 dataset was used. For more information refer to (Zhang et al., 2018).

Lang pair	Embed	AVG	$c[0]$	$c[0:1]$	$c[0:2]$	$c[0:3]$
Lang→EN						
ES→EN	FT	82.97	82.40	**84.50**	83.97	82.90
	BERT	92.10	92.00	**93.23**	93.13	92.20
DE→EN	FT	79.33	78.73	**81.87**	80.20	77.93
	BERT	89.76	89.66	**91.83**	91.20	90.57
EN→Lang						
EN→ES	FT	82.33	82.07	**85.47**	84.60	83.17
	BERT	93.63	93.66	**94.10**	94.00	92.80
EN→DE	FT	74.73	74.50	**79.10**	78.70	76.90
	BERT	91.30	91.43	**91.90**	91.53	90.30
Lang1→Lang2						
DE→ES	FT	73.27	72.20	**77.43**	75.96	74.60
	BERT	87.80	87.57	90.23	**90.36**	88.96
ES→DE	FT	68.90	68.07	**73.97**	73.10	72.43
	BERT	87.70	87.70	**89.67**	89.50	88.53

Table 3: Accuracy using in-domain FastText (FT) and Contextualized mBERT embeddings. The best results for each row in **Bold** & for each direction in **gray**.

In-domain embeddings: To ensure comparability to state-of-the-art results, we further utilized in-domain FastText embeddings as those used in (Aldarmaki and Diab, 2019) as well as contextualized-based word embeddings. For the in-domain FastText embeddings, the FastText (Bojanowski et al., 2017) is utilized to generate word embeddings from 1 Billion Word benchmark (Chelba et al., 2014) for English, and equivalent subsets of about 400 million tokens from WMT'13 (Bojar et al., 2013) news crawl data. For the contextualized-based embeddings, we utilized multilingual BERT (mBERT) introduced in (Devlin et al., 2019) as contextual word embeddings, in which representations from the last BERT layer are taken as word embeddings. As shown in Table 3, using in-domain word embeddings yields stronger results compared to the pre-trained embeddings we use in the previous experiments as illustrated in Table 2. On the other hand, we observe additional improvements using mBERT as word embeddings on all models. Furthermore, increasing K has positive effect on both embeddings, in which $c[0:1]$ demonstrate performance gains compared to other models in all language directions. This trend is clearly observed in the zero-shot performance between the non English languages.

Furthermore, as shown in Table 4, we obtained a state-of-the-art result using mBERT $c[0:1]$ with **91.83%** average accuracy across all translation directions compared to the 84.03% average accuracy of ELMo as reported in (Aldarmaki and Diab, 2019).

Model	**Average Accuracy**
FastText (dict) [ALD2019]	69.04
ELMo (word) [ALD2019]	82.23
FastText (word) [ALD2019]	74.00
FastText AVG (sent) [ALD2019]	76.92
ELMo AVG (sent) [ALD2019]	84.03
FastText $c[0]$ (sent)	76.33
FastText $c[0:1]$ (sent)	80.39
FastText $c[0:2]$ (sent)	79.42
FastText $c[0:3]$ (sent)	77.99
mBERT AVG (sent)	90.38
mBERT $c[0]$ (sent)	90.34
mBERT $c[0:1]$ (sent)	**91.83**
mBERT $c[0:2]$ (sent)	91.62
mBERT $c[0:3]$ (sent)	90.56

Table 4: The average accuracy of various models across all language retrieval directions as reported in (Aldarmaki and Diab, 2019), refer to as [ALD2019] in the table, along with the different DCT-based models in this work, in which (word) refers to word-level mapping, (sent) refers to sentence-level mapping, and (dict) refers to the baseline (using a static dictionary for mapping). **Bold** shows the best overall result.

5 Conclusion

In this paper, we extended the application of DCT encoder to multi- and cross-lingual settings. Experimental results across different languages showed that similar to English using DCT outperform the vector averaging. We further presented a sentence-level-based approach for cross-lingual mapping without any additional training parameters. In this context, the DCT embedding is used to generate sentence representations, which are then used in the alignment process. Moreover, we have shown that incorporating structural information encoded in the lower-order coefficients yields significant performance gains compared to the AVG in sentence translation retrieval.

Acknowledgments

We thank Hanan Aldarmaki for providing us the in-domain FastText embeddings and for sharing many helpful insights. We would also like to thank 3 anonymous reviewers for their constructive feedback on this work.

References

Nasir Ahmed, T_ Natarajan, and Kamisetty R Rao. 1974. Discrete cosine transform. *IEEE transactions on Computers*, 100(1):90–93.

Hanan Aldarmaki and Mona Diab. 2019. Context-aware cross-lingual mapping. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3906–3911, Minneapolis, Minnesota. Association for Computational Linguistics.

Nada Almarwani, Hanan Aldarmaki, and Mona Diab. 2019. Efficient sentence embedding using discrete cosine transform. 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 3672–3678, Hong Kong, China. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Ondřej Bojar, Christian Buck, Chris Callison-Burch, Christian Federmann, Barry Haddow, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2013. Findings of the 2013 Workshop on Statistical Machine Translation. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 1–44, Sofia, Bulgaria. Association for Computational Linguistics.

Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. 2014. One billion word benchmark for measuring progress in statistical language modeling. In *INTERSPEECH 2014, 15th Annual Conference of the International Speech Communication Association, Singapore, September 14-18, 2014*, pages 2635–2639. ISCA.

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.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Felix Hill, Kyunghyun Cho, and Anna Korhonen. 2016. Learning distributed representations of sentences from unlabelled data. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1367–1377, San Diego, California. Association for Computational Linguistics.

Subhradeep Kayal and George Tsatsaronis. 2019. EigenSent: Spectral sentence embeddings using higher-order dynamic mode decomposition. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4536–4546, Florence, Italy. Association for Computational Linguistics.

Lajanugen Logeswaran and Honglak Lee. 2018. An efficient framework for learning sentence representations.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Vinit Ravishankar, Lilja Øvrelid, and Erik Velldal. 2019. Probing multilingual sentence representations with X-probe. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pages 156–168, Florence, Italy. Association for Computational Linguistics.

Andreas Rücklé, Steffen Eger, Maxime Peyrard, and Iryna Gurevych. 2018. Concatenated power mean word embeddings as universal cross-lingual sentence representations. *arXiv preprint arXiv:1803.01400*.

Xuancheng Shao and Steven G Johnson. 2008. Type-ii/iii dct/dst algorithms with reduced number of arithmetic operations. *Signal Processing*, 88(6):1553–1564.

Xiao Shu, Xiaolin Wu, and Bolin Liu. 2017. A study on quantization effects of dct based compression. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 3500–3504. IEEE.

S Toldova, O Lyashevskaya, A Bonch-Osmolovskaya, and M Ionov. 2015. Evaluation for morphologically rich language: Russian nlp. In *Proceedings on the International Conference on Artificial Intelligence (ICAI)*, page 300. The Steering Committee of The World Congress in Computer Science, Computer . . . .

Andrew B Watson. 1994. Image compression using the discrete cosine transform. *Mathematica journal*, 4(1):81.

Ziyi Yang, Chenguang Zhu, and Weizhu Chen. 2019. Parameter-free sentence embedding via orthogonal basis. 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 638–648, Hong Kong, China. Association for Computational Linguistics.

Biao Zhang, Deyi Xiong, and Jinsong Su. 2018. Accelerating neural transformer via an average attention network. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1789–1798, Melbourne, Australia. Association for Computational Linguistics.

Xunjie Zhu and Gerard de Melo. 2020. Sentence analogies: Linguistic regularities in sentence embeddings. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3389–3400, Barcelona, Spain (Online). International Committee on Computational Linguistics.

A Appendices

Table 5 shows the complete numerical results for the probing tasks on all languages.

	Language	AVG	c[0]	c[0:1]	c[0:2]	c[0:3]	c[0:4]
SentLen	EN	56.28	**89.03**	88.91	88.95	88.7	88.08
	ES	59.92	89.59	90.00	89.8	89.73	**90.05**
	FR	57.9	**93.72**	93.44	93.14	92.82	92.38
	DE	53.41	**88.81**	88.36	88.16	87.54	87.69
	RU	54.42	**89.66**	89.12	89.18	88.26	88.04
WC	EN	26.97	**66.69**	64.55	62.49	60.39	59.08
	ES	25.4	**64.80**	62.18	60.62	58.76	57.64
	FR	27.14	**68.60**	66.13	64.71	62.8	61.04
	DE	29.33	**64.99**	64.52	63.93	63.12	61.54
	RU	36.33	**67.50**	65.58	64.69	62.69	61.32
Bshift	EN	54.78	54.98	54.58	54.86	54.81	**55.58**
	ES	54.7	54.52	54.53	54.21	54.71	**55.77**
	FR	54.69	54.7	54.68	54.91	55.53	**56.50**
	DE	54.23	54.22	54.35	54.43	54.6	**56.46**
	RU	56.48	56.8	56.81	56.28	57.4	**58.51**
TreeDepth	EN	41.34	45.18	48.64	49.84	49.44	**50.47**
	ES	42.9	48.53	52.29	53.34	**53.87**	53.54
	FR	41.06	47.68	50.05	51.65	**52.27**	52.15
	DE	37.06	41.97	45.14	47.33	**47.55**	47.36
	RU	35.27	39.21	40.76	**41.02**	40.65	40.51
Tense	EN	86.49	89.23	91.83	92.17	**92.26**	92.21
	ES	94.52	95.97	**96.68**	96.67	96.62	96.53
	FR	91.96	94.06	95.7	95.96	**96.12**	95.99
	DE	94.13	94.71	95.82	**96.44**	96.28	95.92
	RU	86.07	86.39	90.28	**90.4**	90.16	90.38
CoordInv	EN	73.47	74.22	84.56	**87.20**	87.03	87.19
	ES	67.08	68.13	81.61	84.15	85.17	**85.77**
	FR	71.06	71.12	85.97	88.03	89.21	**89.61**
	DE	74.25	74.33	89.99	92.52	93.45	**94.09**
	RU	60.33	60.77	79.95	83.13	84.03	**84.34**
SubjNum	EN	76.46	77.41	80.49	81.68	81.76	**82.31**
	ES	86.4	86.68	89.34	90.42	90.12	**90.84**
	FR	88.48	88.62	91.05	92.23	92.72	**92.76**
	DE	75.94	75.78	78.79	78.9	79.25	**79.28**
	RU	70.47	70.44	72.31	72.81	73.12	**73.13**
ObjNum	EN	68.44	69.71	71.78	73.24	73.98	**74.93**
	ES	78.31	79.23	82.21	83.96	85.2	**85.7**
	FR	77.47	78.5	83.74	85.82	86.92	**88.1**
	DE	68.38	68.74	69.88	70.41	71.14	**71.90**
	RU	63.9	63.79	65.33	65.32	**65.54**	65.11
SOMO	EN	50.12	50.91	**51.72**	51.71	51.36	50.42
	ES	51.7	51.98	51.34	49.62	50.71	**53.07**
	FR	**50.7**	48.85	48.87	49.44	49.56	49.36
	DE	**50.57**	50.47	49.99	49.99	49.99	49.99
	RU	52.49	52.91	52.86	52.8	53.07	**53.13**

Table 5: DCT embeddings Performance per language compared to AVG. **EN**=English, **ES**=Spanish, **FR**=French, **DE**=German, and **RU**=Russian