On the differences between BERT and MT encoder spaces and how to address them in translation tasks

Vazquez , Raul

The Association for Computational Linguistics
2021-08

Vazquez , R , Celikkanat , H , Creutz , M & Tiedemann , J 2021 , On the differences between BERT and MT encoder spaces and how to address them in translation tasks . in J Kabbara , H Lin , A Paullada & J Vamvas (eds) , Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing : Student Research Workshop . The Association for Computational Linguistics , Stroudsburg , pp. 337-347 , Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing , 05/08/2021 . https://doi.org/10.18653/v1/2021.acl-srw.35

http://hdl.handle.net/10138/339869
https://doi.org/10.18653/v1/2021.acl-srw.35

On the differences between BERT and MT encoder spaces and how to address them in translation tasks

Raúl Vázquez raul.vazquez@helsinki.fi
Hande Celikkanat hande.celikkanat@helsinki.fi
Mathias Creutz mathias.creutz@helsinki.fi
Jörg Tiedemann jorg.tiedemann@helsinki.fi

University of Helsinki
Department of Digital Humanities

Abstract

Various studies show that pretrained language models such as BERT cannot straightforwardly replace encoders in neural machine translation despite their enormous success in other tasks. This is even more astonishing considering the similarities between the architectures. This paper sheds some light on the embedding spaces they create, using average cosine similarity, contextuality metrics and measures for representational similarity for comparison, revealing that BERT and NMT encoder representations look significantly different from one another. In order to address this issue, we propose a supervised transformation from one into the other using explicit alignment and fine-tuning. Our results demonstrate the need for such a transformation to improve the applicability of BERT in MT.

1 Introduction

Contextualized token representations produced by pretrained language models (LMs), in particular BERT (Devlin et al., 2019), have ushered in a new era, allowing the separation of unsupervised pretraining of powerful representation spaces, from the supervised training of task-specific, comparatively shallow classifiers on top of these representations. BERT-based models have consistently shown state-of-the-art performance in a variety of tasks, which is largely attributed to the rich information captured by the representations. These capabilities and its Transformer-based architecture suggest that BERT could improve neural machine translation (NMT) as well. However, as shown by Clinchant et al. (2019), although useful, information encoded by BERT is not sufficient by itself for successful MT. The reason for this is still an open question. Some of the most widely accepted hypotheses to date argue that either there is a fundamental discrepancy between the masked language modeling training objective of BERT compared to the generative, left-to-right nature of the MT objective (Song et al., 2019; Lewis et al., 2020) ; or that catastrophic forgetting (Goodfellow et al., 2015) takes place when learning the MT objective on top of the pretrained LM (Merchant et al., 2020). The latter could be caused by the large size of the training data typically used in MT,and by the high capacity decoder network used in MT because to fit the high-capacity model well on massive data requires a huge number of training steps. However, since on the one hand, the left-to-right constraint in MT is potentially more relevant for the decoders than the typically bidirectional encoder that has access to the entire input sequence, and on the other hand, BERT and other pre-trained LMs have been successfully used for other complex problems with large training data and high capacity classifiers (Liu and Lapata, 2019; Witteveen and Andrews, 2019; Huang et al., 2021), it is reasonable to assume there may be further reasons for these discrepancies.

We take a complementary stance and analyze the differences between the representation spaces produced by BERT and those produced by the MT objective. We therefore attempt to *align* these spaces, and investigate whether such an explicit alignment would reshape the BERT representation space to enable its use as an NMT encoder. To the best of our knowledge, this is the first study to investigate the intrinsic differences of pretrained LM and MT spaces, as well as the first attempt to explicitly align them. For reproducing our experiments, we make our code available at https://github.com/Helsinki-NLP/Geometry

2 Methodology

2.1 Comparing the Representation Spaces

Measures of Isotropy and Contextuality. We investigate how the embedding spaces of BERT

Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: Student Research Workshop, pages 337–347

and MT differ by making a layer-by-layer comparison of these spaces. First, we measure the *level of isotropy* of these spaces using the average cosine similarity (*AvgSim*) between the representations of uniformly randomly sampled words from different contexts (Ethayarajh, 2019). (An)isotropy corresponds to the degree of directional (non)uniformity in an embedding space, where perfect isotropy implies directional uniformity in the distribution word vectors. It is important to consider (an)isotropy when discussing contextuality since cosine similarity is relative to the directional uniformity of the sample space. Then, we also generalize *AvgSim* to using the Euclidean distance as our distance metric. Understanding how cosine similarity and the Euclidean distance interact allows for a more complete understanding of the space.

We also make a layer-wise comparison using two of the anisotropy-adjusted contextuality metrics presented in Ethayarajh (2019): *SelfSim:* average cosine similarity between the contextualized representations of a word across its occurrences in the dataset, and *IntraSim:* average cosine similarity between representations of words in a sentence and the sentence mean vector. Both metrics are corrected for anisotropy via subtracting the corresponding *AvgSim*, assuming *AvgSim* as a measure of anisotropy.

Measures of Representational Similarity. We measure the similarities between pairs of layers of both models using Representational Similarity Analysis (RSA) (Laakso and Cottrell, 2000; Kriegeskorte et al., 2008) and Projection-Weighted Canonical Correlation Analysis (PWCCA) (Morcos et al., 2018) as task-agnostic measures.

RSA, originally developed for neuroscience, and later adopted for quantifying the similarity between neural networks (Chrupała and Alishahi, 2019; Abnar et al., 2019) works by taking a set of input stimuli of size n, and running them through the models to be compared. For each model, the activations to each of the n stimuli points are pairwise compared to each other using a similarity metric to compute a an adjacency matrix of size $[n \times n]$ between the stimuli points obtained. These matrices are then contrasted against each other using the Pearson's correlation coefficient, giving a measure of the "representational similarity".

PWCCA is an extension over the SVCCA (Singular Value Canonical Correlation Analysis) distance measure (Raghu et al., 2017), which combines Singular Value Decomposition (SVD) and Canonical Correlation Analysis (CCA) (Hotelling, 1936). CCA is invariant to linear transforms, hence, it is useful for finding shared structures across representations which are superficially dissimilar, making it a good tool for comparing the representations across groups of networks and for comparing representations. Specifically, given the two sets of n corresponding representations from two models, PWCCA performs (1) SVD over the dimension space to prune redundant dimensions, (2) CCA to find linear transformations of the two spaces' dimensions, which are maximally correlated to each other, and (3) a weighted average of the resulting correlation coefficients, which favor the ones that are more relevant to the underlying representations.

2.2 Aligning the Representation Spaces

We present two methods to align the BERT space to that of the MT encoder: *(i)* an explicit alignment transformation that forces BERT representations to better match those of the MT encoder, and *(ii)* an implicit alignment effect achieved by a fine-tuning process which uses translation as its objective.

Explicit Alignment Transformation. We build upon Cao et al. (2020), maximizing the *contextual alignment* the model can achieve via the average accuracy on the *contextual word retrieval task*. This method presents several advantages that can be leveraged in our work. It is multilingual, it respects contextuality of the embeddings, and it makes use of rather reliable, widely used and not-memory intensive alignment algorithms (Brown et al., 1993; Och and Ney, 2003)

The task, as originally posed by Cao et al. (2020) is as follows. Given a parallel pre-aligned corpus C of source-target pairs (s, t), and one word within a source sentence, the objective is to find the corresponding target word. Let each sentence pair (s, t) have word pairs, denoted $a(s, t) = (i_1, j_1), ..., (i_m, j_m)$, containing position tuples (i, j) such that the words s_i and t_j are translations of each other. We use a regularized loss function $Loss = L + \lambda R$ so that L aligns the embeddings from one model, $f_1(i, s)$, to the ones of the other model $f_2(j, t)$:

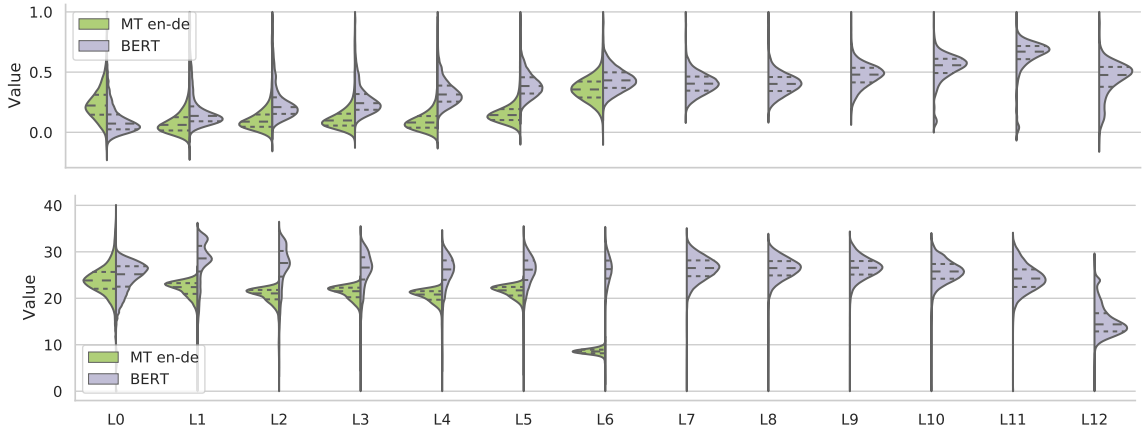

Figure 1: Cosine similarity *(top)* and Euclidean distance *(bottom)* distributions between randomly sampled words. Note that BERT has 12 layers and MT encoder has 6 layers, so the layers should be compared according to their relative positions, such as comparing the final layer of BERT to the final layer of MT encoder.

$$L(f_1, f_2; C) = -\sum_{\substack{(s,t)\in C \\ (i,j)\in a(s,t)}} \sum \text{sim}\left(f_1(i,s), f_2(j,t)\right)$$

$$R(f; C) = \sum_{s\in C} \sum_{i=1}^{len(t)} \|f_1(i,s) - f_1^{\circ}(i,s)\|_2^2$$

where f_1° denotes the pretrained model 1 before alignment and R is the regularization term that imposes a penalty if the target embeddings stray too far from their initialization. We validate using a version of Cao et al. (2020) word retrieval task using a nearest neighbor retrieval function:

$$N(i, s|f_1, f_2) = \underset{t\in C, 0\geq j\geq len(t)}{\arg\max} \text{sim}\left(f_1(i,s), f_2(j,t)\right)$$

We propose to modify the regularized loss function $Loss = L + \lambda R$ so that L aligns the embeddings from one model, $f_1(i,s)$, to the ones of another model, $f_2(j,t)$, and also use a regularization term R to impose a penalty if the aligned embeddings stray too much. In contrast with Cao et al. (2020), this allows for alignment between embeddings produced by different models. Specifically, we align the representations in the final layer of the pretrained language model, to that of the encoder of the MT model. Although in this work, we focus on aligning the different representations for the same word to each other, aligning embedding spaces of different languages and different models is also an interesting future direction.

Implicit Alignment via Fine-tuning. We fine-tune a hybrid model consisting of BERT in the encoder side that sends its representations to a pretrained MT decoder. We then use smoothed cross entropy loss as our training objective to fine-tune BERT representations for performing MT. The outputs of BERT are passed through a linear projection layer to match the dimension of the MT decoder and then fed into the decoder in the same way as in the standard Transformer architecture.

3 Comparing The Embedding Spaces.

We compare the representation spaces produced by BERT and the encoder of a Transformer trained on the MT task. BERT is composed of 12 layers, plus an initial input embedding layer, with a dimension of 768. The MT system we apply consists of an input embedding layer followed by 6 Transformer layers with a hidden dimension of 512. We use the pretrained `bert-base-uncased` model, as well as the pretrained English-German translation model `opus-mt-en-de`, both from the HuggingFace library (Wolf et al., 2019). Following Ethayarajh (2019), we extract embeddings using data from the SemEval Semantic Textual Similarity tasks from 2012 to 2016 (Agirre et al., 2016).

Average similarity between random tokens. Figure 1 presents the layer-wise cosine similarity *(top)* and the Euclidean distance *(bottom)* distributions of randomly sampled words.The behavior of BERT in Figure 1*(top)* is consistent with the findings of Ethayarajh (2019). The level of anisotropy of the embedding representations throughout layers of BERT increases towards higher layers, with the exception of a slight drop at the last layer (L12), considering the average cosine similarity of the rep-

resentations as a proxy measure of anisotropy. Further, we notice Figure 1*(bottom)* that BERT embeddings follow an inverted U-shape. This, together with the *AvgSim* trend, means that the embedding space starts by stretching out and becoming narrower, later on to spread out shorter embeddings in layer 12, in line with (Voita et al., 2019).

The MT-based representations, however, look significantly different. The cosine-based *AvgSim* follows an almost U-like trend: it starts from a relatively high level at layer 0, then immediately drops and stays low throughout the middle layers, before a sudden increase at the final layer (L6). In particular:

1. a high average similarity of the MT embeddings in layer 0 is striking since the representations are not yet that "contextualized" this early in the model, and

2. the gradual increase of average similarity in BERT layers, versus the very steep increase at the last layer of MT model.

Behavior (1) might be caused by the shared source-target vocabularies and the embedding layer in the MT model in the encoder and the decoder being shared. Such shared processing can result in a seeming inflation of the cosine similarity of randomly selected vectors, which actually belong to two different language spaces. To test for this hypothesis, we check the average Euclidean distance between randomly selected tokens in Figure 1-*bottom*. Interestingly, we do not observe considerable high levels of closeness between random words in layer 0, and the distribution is widespread. That is, the embeddings are organized in a narrow cone but have a wide range of lengths. This behaviour might arise from the system needing to represent both languages in the same space, and the interplay between training the embeddings layer at the target side while needing to keep source embeddings apart enough - future work is necessary to confirm this. Motivated by these findings, we emphasize that using both metrics and observing how they interact allows for a more complete understanding of the representation spaces. [1]

Finding (2) is more relevant to our main question of the differences between the geometries of

[1]Cosine similarity does not take into account the magnitude of the vectors at play, making it susceptible to the existence a large value in one of the entries of a high-dimensional vector, while Euclidean distance is hard to interpret in high-dimensional spaces and it is unbounded from above.

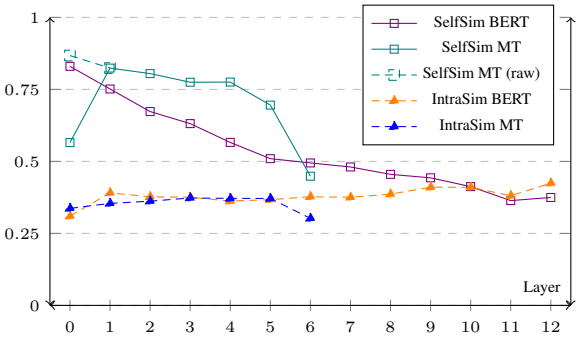

Figure 2: Comparison of contextualization for BERT and MT spaces using *SelfSim* and *IntraSim*. We also present the raw *SelfSim* before anisotropy correction.

BERT and MT. Both metrics show a more gradual increase in the closeness of random tokens in the BERT over layers, as compared to an abrupt increase in the MT space. Therefore, we can deduce that the MT model can keep random representations successfully apart for all but the uppermost of the layers. We hypothesize that this monotonously increasing levels of closeness of random token embeddings in BERT may be contributing to its suboptimal machine translation performance. To verify this hypothesis, in section 4 we present results on MT performance after alignment and in section 4.1 we show how the alignment method changes the embeddings distributions.

Similarity between tokens of the same form. *SelfSim* will be high in less contextualized models, because such models use similar representations for each occurrence of the same token. Highly contextualized models will have lower *SelfSim* since every occurrence of the word will have a different representation. Comparing the two spaces (Figure 2), we again observe different trends. *SelfSim* steadily drops for BERT except for the last layer, showing an increase in the contextuality of the representations. For the MT model, on the other hand, we observe a steep drop at layer 6, indicating a sudden increase in contextuality here. All in all, BERT gradually increases contextualization whereas the MT encoder tends to model individual lexical concepts in most layers before adding a strong contextual influence in the last one.

Once again, we see a different behavior in layer 0 of the MT model, which is characterized by low *SelfSim* in the embedding layer. This a direct result of the high *AvgSim* value at the embeddings layer (due to the shared vocabulary space) which is the anisotropy correction factor for *SelfSim*. We

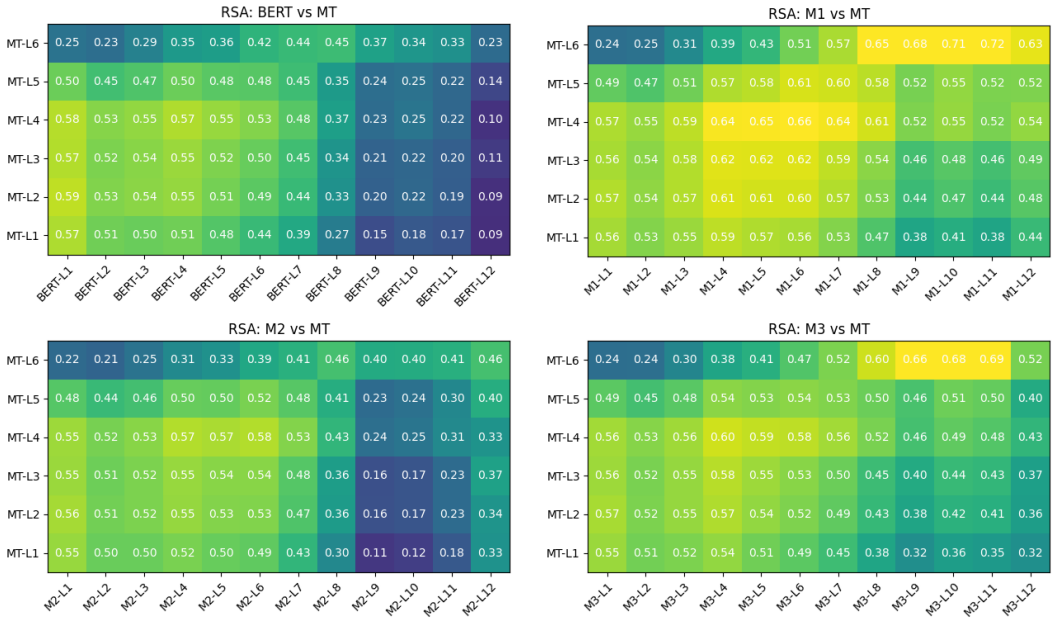

Figure 3: Representation similarity analysis between of out-of-box BERT, as well as the aligned models M1/M2/M3, with MT model from HuggingFace.

deduce that anisotropy-corrected *SelfSim* cannot straightforwardly be interpreted as a measure of contextuality in the embeddings layer of MT models with a shared source-target vocabulary. For comparison, we, therefore, also present the uncorrected *SelfSim (raw)* value (dashed line) for this layer, which confirms this reasoning.

Similarity between tokens within the same sentence. We check the average similarity between tokens in the same sentence (*IntraSim*). Figure 2 reveals different behavior between the two models. In particular, we see a smooth increase over the layers for both models until the penultimate layer, pointing to an increasing level of in-sentence contextualization, as shown by the embeddings of the words in the same sentence gradually coming together. However, the behavior at the final layer is different between the two models. We observe an increase in *IntraSim* for the BERT model at the last layer, in contrast to the drop at the last layer of the MT model. In other words, the MT model is suddenly discriminating between the words in the sentence at layer 6, just before passing information to the decoder. We hypothesize that it may be useful for the MT decoder to have access to representations that are less contextualized at a source sentence level, since it still needs to add semantic information for decoding into the target language. Notice that *SelfSim* and *IntraSim* decrease for final

	Encoder	Explicit alignment	Fine-tuning
MTbaseline	Trf	✗	✗
huggingface en-de	(6-layers)	✗	✗
M1:align	BERT	✓	✗
M2:fine-tune	(12-layers)	✗	✓
M3:align+fine-tune		✓	✓

Table 1: Model setups. **MTbaseline** and **huggingface en-de** are baseline models which use Transformer ("Trf") as encoder. **M1**, **M2** and **M3** utilize various combinations of the proposed alignment strategies.

layer of the MT model. That is, similarity of word forms in different contexts is decreasing greatly and similarity of words to the mean sentence vector is (to a smaller degree) also decreasing. This might be an indication of the different constraints MT models have on contextualization. For example, the model may have a tendency to pay strong attention to syntactic and positional information, instead of focusing on shared semantics of the sentence.

Layer-wise similarity analysis between models. Figures 3-*top left* and 4-*top-left* present the results of the representational similarity analysis (RSA) and projection-weighted canonical correlation analysis (PWCCA) between out-of-the-box BERT and the MT model representational spaces. Both analyses depict higher similarity values between the lower layers of the models. At the lower layers, the

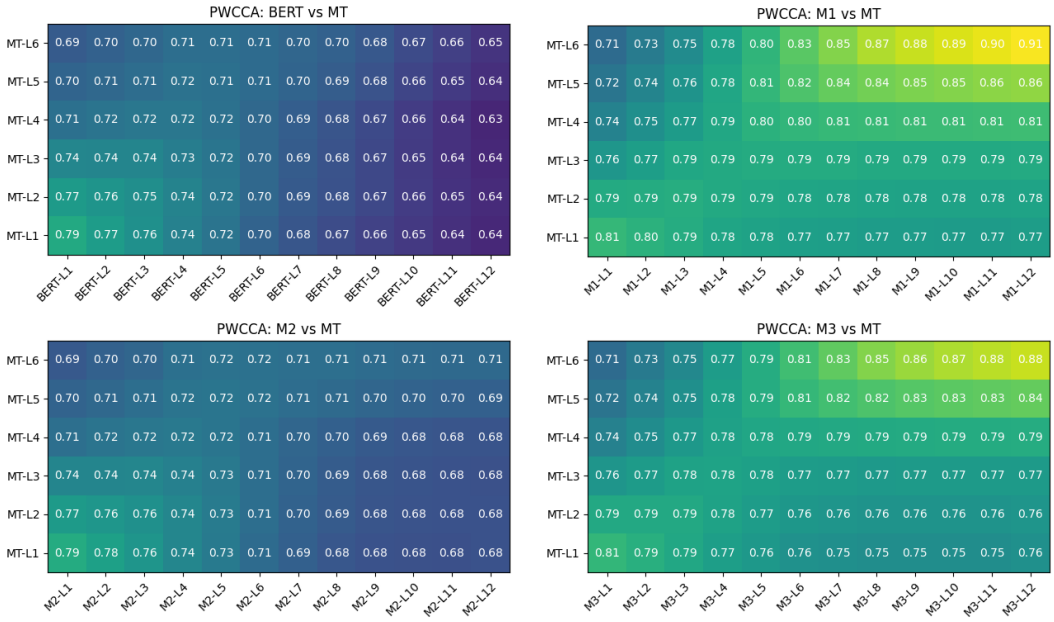

Figure 4: Representational similarity analysis of BERT, HuggingFace MT model, and aligned models M1/M2/M3.

representations are not yet changed so much from their initial starting point, so it is to be expected that they are more similar between the models. Towards the higher levels, though, the similarity decreases. The BERT representations gain distance from the MT representations, reaching the lowest similarity between the BERT-L12 and the MT layers.

4 Aligning the Representation Spaces

To address the discrepancies observed in the BERT and the MT encoder embedding spaces, we use the transformations from section 2.2. We use five different setups (Table 1). Two of these use 6-layered Transformer encoders and serve as baselines: the **MTbaseline** model, a transformer-based MT model trained from scratch with the fine-tuning data (Table 2), and **Huggingface en–de** a state-of-the-art, pretrained Transformer model. We compare the proposed alignment methods using **M1**, which uses only the explicit alignment transformation strategy, **M2**, which uses the implicit alignment via fine-tuning strategy, and the hybrid **M3**, which combines the two strategies.

Data. We use data from the English-German sections of the MuST-C dataset (Di Gangi et al., 2019), Europarl (Koehn, 2005), extracted using OpusTools (Aulamo et al., 2020) and the development tarball from the WMT2019 news translation shared task (Bojar et al., 2019) in the proportions indicated in Table 2. We test using the MuST-C provided

	Train		Val.
	Explicit Alignment	Fine-Tuning	
Europarl	45K	150K	1.5K
MuST-C	45K	150K	1.5K
newstest	13K	13K	500
Total	102K	313K	3.5K

Table 2: Train and validation splits for the datasets.

test-split, newstest2014 (Bojar et al., 2014) and newstest2015 (Bojar et al., 2015), which were excluded from the train data. All of the data splits are attainable using our repository.

We purposefully restrict the data amount used for training the alignments. Such aligned systems should be able to work under less intensive resource requirements. The size of the training data for both methods varies, because we try to keep the explicit alignment comparable to what was originally proposed for mBERT (Cao et al., 2020), whereas the implicit alignment via fine-tuning requires more data since the MT decoder is also to be fine-tuned.

Results. Table 3 presents the BLEU scores for five setups. Notably, we see that by explicitly aligning the embedding spaces in a supervised way (**M1**) the system is already able to perform translation reasonably well. Besides being data efficient, due to its simplicity, the alignment method used for **M1** is also memory efficient and fast to train. We think that this shows how applying the simple alignment procedure described in section 2.2 can be

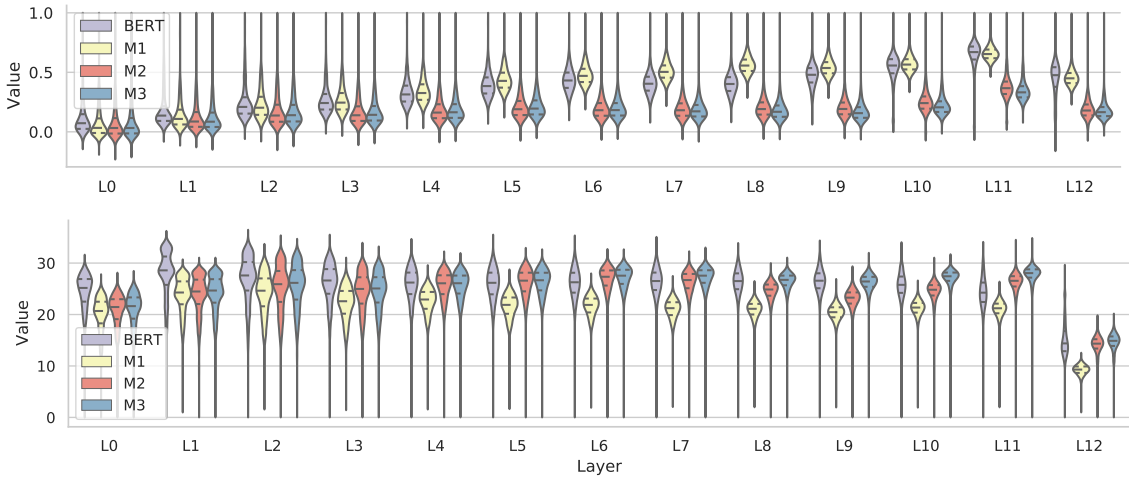

Figure 5: Comparison of out-of-box BERT and MT models, against the aligned models M1/M2/M3, in terms of the Cosine similarity *(top)* and Euclidean distance *(bottom)* distributions between randomly sampled words.

	MuST-C	newstest	
		2014	2015
MTbaseline	29.9	14.5	17.6
huggingface en-de	33.7	28.3	31.1
M1:align	21.4	18.1	18.9
M2:fine-tune	33.8	23.9	28.0
M3:align+fine-tune	34.1	25.0	29.2

Table 3: BLEU scores for EN-DE test sets.

used to make the rich world-knowledge captured by BERT accessible for NMT by making the embedding spaces compatible. In section 4.1, we investigate the distributional changes in the embeddings spaces caused by the alignments.

We also notice that fine-tuning in **M2** works quite well. We highlight how data efficient this method is. After training for 1 epoch we obtain already over 30 BLEU points for MuST-C and after 3 epochs of fine-tuning we achieve results comparable with the **huggingface en-de model**. On MuST-C data, **M3** yields similar results, notably however, it converges much faster. At only 1% of the 1st epoch ($\sim$ 3K utterances) it achieves already 85% of its performance in both test sets, and with 10K utterances it starts to converge. The results obtained with **newstest 2014** and **newstest 2015** follow a similar trend, yet fail to surpass the huggingface model – a state-of-the-art MT model trained with all available EN-DE resources ($\sim$ 350.7M parallel sentences) from OPUS (Tiedemann, 2012). However, in all cases, we observe a better performance than the **MTbaseline**, an MT model trained with the same restricted data. These results indicate that BERT can indeed be used as an MT encoder, but only with a careful alignment procedure that overcomes the incompatibilities between the encoders.

4.1 The Aligned BERT Space

Finally, we check the effects of the alignment schemes on the geometry of the BERT space. Here, our specific question of interest is in which ways the BERT-produced embedding space became more similar (or not) to the MT space after applying the alignment methods.

AvgSim. Figure 5 shows layer-wise cosine similarity *(top)* and Euclidean distance *(bottom)* distributions of random words of the aligned models.

While all three distributions are different from the original BERT, **M1** is the least different in terms of where the distribution is centered, but even here the distributions are less skewed/more symmetrical, with respect to the cosine similarity. However, the Euclidean distance results show that **M1** consistently produces shorter word vectors. This aligned model is hence creating a space that is as narrow as BERT's, but not as elongated. This might be due to the regularization term in the supervised alignment not allowing the embeddings to drift too far from its pre-optimized setting, as well as the alignment being explicitly done for the last layer.[2] For both metrics, **M2** and **M3** are noticeably different compared to the original BERT and similar to each other. This indicates that aligning via fine-tuning propagates information in such a way that the space

[2]We see changes in the distributions of all layers due to backpropagation of information at training time.

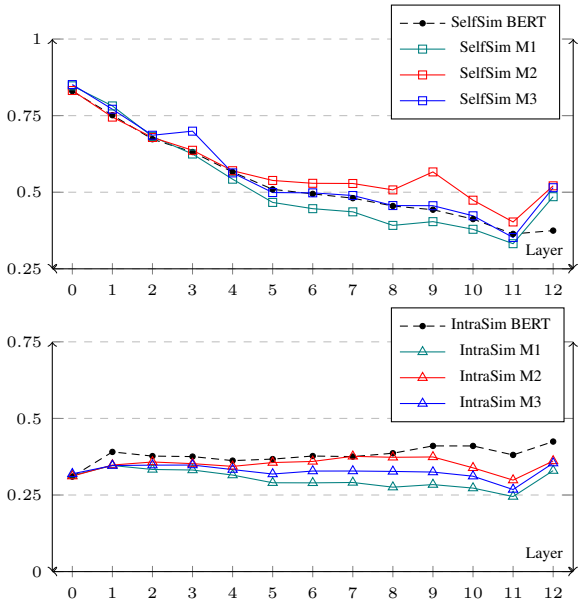

Figure 6: Comparison of BERT and the aligned models M1/M2/M3, in terms of *SelfSim* and *IntraSim*.

is reshaped drastically. The increase in the BLEU scores discussed above correlates with the amount of change we observe in the distance distributions.

SelfSim and IntraSim. Figure 6-*top* shows considerable change in the *SelfSim* of M1/M2/M3 following the alignment. Now all three models show an abrupt increase in the similarity of tokens of the same form in the ultimate layer. In other words, these models are retrieving information related to the specific word form, just before passing the information to the decoder. This finding is in line with (Voita et al., 2019), who find that the MT decoder seems to require more information about the specific word form's representation, as compared to the overly contextual representations that the pretrained language models tend to produce.

Figure 6-*bottom* compares the after-alignment *IntraSim* with before-alignment case. Note that the M1/M2/M3 values in general are lower than the BERT, throughout the layers. This confirms the previous findings that the word forms seem to retain their original representations more, and adjusting to the sentence context less.

Layer-wise similarity analysis between models. Figures 3 and 4 show how the responses of M1/M2/M3 become significantly similar to that of the MT model post-alignment. Note that interestingly the explicit alignment method is particularly successful in achieving similarity to the MT model, in terms of similarities between responses to pairs of stimuli (as measured by RSA) and correlation of model responses over changing stimuli (as measured by PWCCA). However, as shown in Table 3, model M1 is outperformed by M2 and M3, which might be related to the anisotropy levels of M1 being similar to those of BERT (Figure 5).

5 Related Work

Analysis of contextualized representations. While there has been huge efforts to analyze word representations, most of it has been conducted using probing tasks (McCann et al., 2017; Conneau and Kiela, 2018; Conneau et al., 2018; Hewitt and Manning, 2019). Similarly, Merchant et al. (2020) study the effects of fine-tuning BERT representations on a specific set of probing tasks and analyse the change in the contextual representations using similarity analysis. Mimno and Thompson (2017) quantitatively studied static word representations produced with skip-gram with negative sampling. Their work was extended by Ethayarajh (2019) for contextualized embeddings, in which they use word level measures of contextuality to contrast ELMo (Peters et al., 2018), GPT-2 (Radford et al., 2019) and BERT (Devlin et al., 2019). Voita et al. (2019) present a comparison of contextualized representations trained with different objectives, using CCA and and mutual information to study information flow across networks. They conclude that although MT-produced representations do get refined with context, the change in those is not as extreme as for masked LM-produced representations (BERT-like), which is in line with our observations of higher *SelfSim* and lower *IntraSim* (i.e. not ultra-contextualized embeddings) for MT and aligned models as compared to BERT.

Pretrained LMs in NMT. Clinchant et al. (2019) present a systematic comparison of methods to integrate BERT into NMT models, including using BERT at the embedding level or for initializing an encoder. Zhu et al. (2020) propose a BERT-fused MT system that uses additional attention modules between the outputs of BERT and the encoder and decoder of the Transformer, increasing the model parameters by the number of parameters the chosen BERT flavour has. Yang et al. (2020) proposes a similar strategy, though using BERT outputs only in the encoder, and a three-fold training technique. Imamura and Sumita (2019) propose a simple yet effective two-stage optimization technique that first freezes BERT, and then fine-tunes

over the full model parameters set. We argue that this is similar to the align and fine-tune approach we propose for incorporating BERT into MT. Finally, a number of studies leverage pretraining techniques. MASS (Song et al., 2019) is partly inspired by BERT, but it is pretrained in NMT and is tailored to match the way prediction is done in NMT (left-to-right). Liu et al. (2020) enhance transformer-based MT systems performance by using a BART pretraining technique (Lewis et al., 2020) in a multilingual fashion to initialize an NMT system.

Alignment. Numerous methods have been proposed for aligning (contextualized) word representations (Och and Ney, 2003; Ruder et al., 2019). Wang et al. (2019) learn an optimal linear transformation between embedding spaces. Schuster et al. (2019) propose a similar approach using the centroids of the instances of the same word in different contexts. Our work is closer to Cao et al. (2020), which use a resource-efficient algorithm that takes into account the contextuality of embeddings.

6 Conclusion

This paper provides an analysis of the intrinsic differences between BERT and machine translation encoders. We compare the representation spaces of both models and pinpoint discrepancies between them. We show that this mismatch can be remedied through an alignment strategy, which successfully reshapes BERT into an effective MT encoder. We also study the effects that the alignment methods have on the geometry of the embeddings spaces.

Acknowledgments

This work is part of the FoTran project, funded by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement № 771113).

We also acknowledge the CSC – IT Center for Science Ltd., for computational resources.

References

Samira Abnar, Lisa Beinborn, Rochelle Choenni, and Willem Zuidema. 2019. Blackbox meets blackbox: Representational similarity and stability analysis of neural language models and brains. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 191–203, Florence, Italy. 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.

Mikko Aulamo, Umut Sulubacak, Sami Virpioja, and Jörg Tiedemann. 2020. OpusTools and parallel corpus diagnostics. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 3782–3789. European Language Resources Association.

Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Christof Monz, Matt Post, and Lucia Specia, editors. 2014. *Proceedings of the Ninth Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Baltimore, Maryland, USA.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, André Martins, Christof Monz, Matteo Negri, Aurélie Névéol, Mariana Neves, Matt Post, Marco Turchi, and Karin Verspoor, editors. 2019. *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*. Association for Computational Linguistics, Florence, Italy.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Barry Haddow, Matthias Huck, Chris Hokamp, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Carolina Scarton, Lucia Specia, and Marco Turchi. 2015. Findings of the 2015 workshop on statistical machine translation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 1–46, Lisbon, Portugal. Association for Computational Linguistics.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Steven Cao, Nikita Kitaev, and Dan Klein. 2020. Multilingual alignment of contextual word representations. In *International Conference on Learning Representations*.

Grzegorz Chrupała and Afra Alishahi. 2019. Correlating neural and symbolic representations of language. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, page 2952–2962, Florence, Italy.

Stephane Clinchant, Kweon Woo Jung, and Vassilina Nikoulina. 2019. On the use of BERT for neural machine translation. In *Proceedings of the 3rd*

Workshop on Neural Generation and Translation, pages 108–117, Hong Kong. Association for Computational Linguistics.

Alexis Conneau and Douwe Kiela. 2018. Senteval: An evaluation toolkit for universal sentence representations. *arXiv preprint arXiv:1803.05449*.

Alexis Conneau, Germán 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.

Mattia A. Di Gangi, Roldano Cattoni, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2019. MuST-C: a Multilingual Speech Translation Corpus. 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 2012–2017, Minneapolis, Minnesota. Association for Computational Linguistics.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of BERT, ELMo, and GPT-2 embeddings. 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 55–65, Hong Kong, China. Association for Computational Linguistics.

Ian J. Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. 2015. An empirical investigation of catastrophic forgetting in gradient-based neural networks.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. 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 4129–4138, Minneapolis, Minnesota. Association for Computational Linguistics.

Harold Hotelling. 1936. Relations between two sets of variates. *Biometrika*, 28:321–337.

Wen-Chin Huang, Chia-Hua Wu, Shang-Bao Luo, Kuan-Yu Chen, Hsin-Min Wang, and Tomoki Toda. 2021. Speech recognition by simply fine-tuning bert.

Kenji Imamura and Eiichiro Sumita. 2019. Recycling a pre-trained BERT encoder for neural machine translation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 23–31, Hong Kong. Association for Computational Linguistics.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86. Citeseer.

Nikolaus Kriegeskorte, Marieke Mur, and Peter A. Bandettini. 2008. Representational similarity analysis-connecting the branches of systems neuroscience. *Frontiers in systems neuroscience*, 2(4).

Aarre Laakso and Garrison Cottrell. 2000. Content and cluster analysis: assessing representational similarity in neural systems. *Philosophical psychology*, 13(1):47–76.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. 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 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *Advances in Neural Information Processing Systems*, volume 30, pages 6294–6305. Curran Associates, Inc.

Amil Merchant, Elahe Rahimtoroghi, Ellie Pavlick, and Ian Tenney. 2020. What happens to BERT embeddings during fine-tuning? In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 33–44, Online. Association for Computational Linguistics.

David Mimno and Laure Thompson. 2017. The strange geometry of skip-gram with negative sampling. In *Proceedings of the 2017 Conference on Empirical*

Methods in Natural Language Processing, pages 2873–2878, Copenhagen, Denmark. Association for Computational Linguistics.

Ari S. Morcos, Maithra Raghu, and Samy Bengio. 2018. Insights on representational similarity in neural networks with canonical correlation. In *Advances in Neural Information Processing Systems*, pages 5727–5836. Curran Associates, Inc.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. 2017. SVCCA: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc.

Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2019. A survey of cross-lingual word embedding models. *Journal of Artifical Intelligence Research*, 65:569–631.

Tal Schuster, Ori Ram, Regina Barzilay, and Amir Globerson. 2019. Cross-lingual alignment of contextual word embeddings, with applications to zero-shot dependency parsing. 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 1599–1613, Minneapolis, Minnesota. Association for Computational Linguistics.

Kitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. Mass: Masked sequence to sequence pre-training for language generation. In *International Conference on Machine Learning (ICML)*.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in opus. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA).

Elena Voita, Rico Sennrich, and Ivan Titov. 2019. The bottom-up evolution of representations in the transformer: A study with machine translation and language modeling objectives. 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 4396–4406, Hong Kong, China. Association for Computational Linguistics.

Yuxuan Wang, Wanxiang , Jiang Guo, Yijia Lui, and Ting Liu. 2019. Cross-lingual bert transformation for zero-shot dependency parsing. 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 5721–5727, Hong Kong, China. Association for Computational Linguistics.

Sam Witteveen and Martin Andrews. 2019. Paraphrasing with large language models. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 215–220, Hong Kong. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Jiacheng Yang, Mingxuan Wang, Hao Zhou, Chengqi Zhao, Yong Yu, Weinan Zhang, and Lei Li. 2020. Towards making the most of bert in neural machine translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, New York, USA.

Jinhua Zhu, Yingce Xia, Lijun Wu, Di He, Tao Qin, Wengang Zhou, Houqiang Li, and Tieyan Liu. 2020. Incorporating bert into neural machine translation. In *International Conference on Learning Representations*.