applied sciences

MDPI

Article

Analyzing and Controlling Inter-Head Diversity in Multi-Head Attention

Hyeongu Yun, Taegwan Kang and Kyomin Jung *

Department of Electrical and Computer Engineering, Seoul National University, Seoul 08826, Korea; youaredead@snu.ac.kr (H.Y.); zd9370@snu.ac.kr (T.K.)
* Correspondence: kjung@snu.ac.kr; Tel.: +82-(0)2-880-1937

Abstract: Multi-head attention, a powerful strategy for Transformer, is assumed to utilize information from diverse representation subspaces. However, measuring diversity between heads' representations or exploiting the diversity has been rarely studied. In this paper, we quantitatively analyze inter-head diversity of multi-head attention by applying recently developed similarity measures between two deep representations: Singular Vector Canonical Correlation Analysis (SVCCA) and Centered Kernel Alignment (CKA). By doing so, we empirically show that multi-head attention does diversify representation subspaces of each head as the number of heads increases. Based on our analysis, we hypothesize that there exists an optimal inter-head diversity with which a model can achieve better performance. To examine our hypothesis, we deeply inspect three techniques to control the inter-head diversity; (1) Hilbert-Schmidt Independence Criterion regularizer among representation subspaces, (2) Orthogonality regularizer, and (3) Drophead as zero-outing each head randomly in every training step. In our experiments on various machine translation and language modeling tasks, we show that controlling inter-head diversity leads to the best performance among baselines.

Keywords: multi-head attention; inter-head similarity; Transformer; machine translation; language modeling; Natural Language Processing; NLP

Citation: Yun, H.; Kang, T.; Jung, K. Analyzing and Controlling Inter-Head Diversity in Multi-Head Attention. *Appl. Sci.* **2021**, *11*, 1548. https://dx.doi.org/10.3390/app11041548

Academic Editor: Julian Szymanski, Andrzej Sobecki, Higinio Mora and Doina Logofătu
Received: 31 December 2020
Accepted: 3 February 2021
Published: 8 February 2021

1. Introduction

Since multi-head attention has been introduced by Vaswani et al. [1], it has become a standard setting across various Natural Language Processing (NLP) tasks. Vaswani et al. have stated that multi-head strategy can collocate information from different representation subspaces and thus improves the performance of attention mechanism, whereas single-head attention averages the information. Most of the state-of-the-arts models report that multi-head attention is helpful to increase their performances, including BERT [2] and XLNet [3] for language understanding, Transformer [1] for machine translation, and HIBERT [4] for document summarizing.

Despite its huge empirical success and dominant usage, few studies have explored the roles of the multi-head strategy to give us a better understanding on how it enhances a model's performance. Clark et al. [5] have analyzed attention maps of multi-head attention and showed that certain heads are relevant to specific linguistic phenomena. Similarly, Voita et al. [6] has analyzed that certain heads are respectively sensitive to various linguistic features by using layer-wise relevant propagation. Although these studies imply that there exists diversity of representation subspaces among multiple heads, their analyses are mainly focused on linguistic diversity.

In order to inspect essential effects of multi-head attention in representational subspaces, Li et al. [7] have proposed the disagreement score which measures cosine similarity between two heads' representation and maximized the disagreement score to diversify inter-head representations. Li et al. have shown that maximizing the disagreement score increases performance, which implies that inter-head statistics in multi-head attention are closely related to the model's performance. However, disagreement score has its limitation

since cosine similarity of two random vectors in high dimension are close to 1, as known as the curse of dimensionality.

To overcome the limitations of previous studies, we seek answers to following three fundamental questions; (1) Does multi-head strategy diversify the subspace representations of each head? (2) Can we finely optimize the degree of inter-head diversity without changing model's architecture? and finally (3) Does controlling inter-head diversity improve a model's performance?

We measure the inter-head similarity of multi-head attention with Singular Vector Canonical Correlation Analysis (SVCCA) [8] and Centered Kernel Alignment (CKA) [9], as they are recently developed tools to measure similarities of two deep representations. Applying these similarity measures, we empirically show that the diversity of multi-head representations does increase as the number of heads increases which is solid evidence supporting the statement of Vaswani et al. [1] that the multi-head strategy utilizes diverse representational subspaces. Furthermore, we suggest three techniques to optimize the degree of diversity among heads without architectural change of a model.

We first focus on trainability of CKA because CKA is differentiable and its gradients can be easily computed with popular frameworks such as Tensorflow [10]. We adopt Hilbert-Schmidt Independence Criterion (HSIC) inspired by CKA as an augmented loss in order to directly diversify the inter-head diversity of a model.

Then, we revisit the orthogonality regularizer that adds disagreement loss [7] between representations of heads. Surprisingly, opposed to the expectation of Li et al. [7] expected, we empirically show that the orthogonality regularizer does not force a model's inter-head diversity to increase measured in SVCCA and CKA. Instead, we find that it helps a model by encouraging top-few SVCCA directions to be closer which can be interpreted as core representations [11].

Lastly, we inspect Drophead method [12] by which a model randomly drops outputs of each head at training to show that we also can decrease the inter-head diversity without architectural change. Drophead reduces an effective number of heads at each training step and hence increases the inter-head similarity, while a model also benefits from the advantages of Dropout [13].

We test our methods on various tasks including De-En IWSLT17 corpus [14], Zh-En in UN parallel corpus [15] on machine translation, and also PTB corpus on language modeling. Our results show that the suggested three methods complement each other and find the optimal inter-head diversity. The models with our methods achieve higher performances compared to their baselines in all experiments.

2. Related Works

As the multi-head strategy has shown its strength in many NLP tasks, there have been several attempts to analyze it with various approaches. By evaluating attention weights of ambiguous nouns in machine translation, Tang et al. [16] have shown that multi-head attention tends to focus on ambiguous tokens more than general tokens. Clark et al. [5] and Raganato et al. [17] also have analyzed attention weights and concluded that each head plays different roles to understand syntactic features. Voita et al. [6] and Michel et al. [18] have claimed that most of the heads can be pruned once the model trained as they have analyzed the multi-head mechanism via layer-wise relevant propagation and ablating heads respectively.

On the other hand, several works have tried to analyze the similarity between representation spaces of neural networks in favor of achieving interpretability. Li et al. [19] have proposed alignment methods with a correlation of neurons' responses and claimed that core representations are shared between different networks while some rare representations are learned only in one network. More recently, Raghu et al. [8] have first applied CCA as a similarity measure and proposed SVCCA in order to pick out perturbing directions from deep representations, and Morcos et al. [20] have suggested Projection Weighted CCA (PWCCA) as a method to make SVCCA more reflective to the subspaces of representations

via projection. Kornblith et al. [9] have proposed CKA as a more robust similarity measure to small numbers of samples using a normalized index of HSIC with kernel methods.

Towards the interpretability of the deep representation, some studies have utilized similarity measures of deep representations. Maheswaranathan et al. [21] have applied CCA, SVCCA, and CKA to Recurrent Neural Networks (RNN) and discovered that the geometry of RNN varies by tasks, but the underlying scaffold is universal. Kudugunta et al. [22] have applied SVCCA across languages on multilingual machine translation to show there are shared representations among language representations. Bau et al. [11] also have applied SVCCA to identify meaningful directions in machine translation and showed that top-few directions in SVCCA similarity are core representations since they are critical to a model's performance when erased.

Closely related to our orthogonal loss, decorrelation methods have been proposed in node level [23–25] and in group of nodes level [7,26]. Rodriguez et al. [23], Xie et al. [24], and Bansal et al. [25] have shown that decorrelating each node through orthogonal constraint can achieve higher performances. Li et al. [7] have applied the decorrelating term to multi-head attention, which inspires us to use orthogonal constraints in order to control inter-head diversity. Gu et al. [26] have showed that cosine similarity based constraint in group of nodes can achieve higher performances, as it improves generalization capacity of the model.

3. Similarity Measures for Multi-Head Attention

3.1. Multi-Head Attention

Multi-head attention is first suggested by Vaswani et al. [1] as a strategy that diversifies representation subspaces in order to fully utilize a model's capability. We briefly review how single-head and multi-head attention operates.

For single-head attention, an output matrix $X' \in \mathbb{R}^{L\times d}$ with its inputs (a query vector $q' \in \mathbb{R}^d$, a key matrix $K' \in \mathbb{R}^{L\times d}$, and a value matrix $V' \in \mathbb{R}^{L\times d}$) is computed as follows;

$$X' = \text{softmax}(\frac{q'K'^T}{\sqrt{d}})V', \tag{1}$$

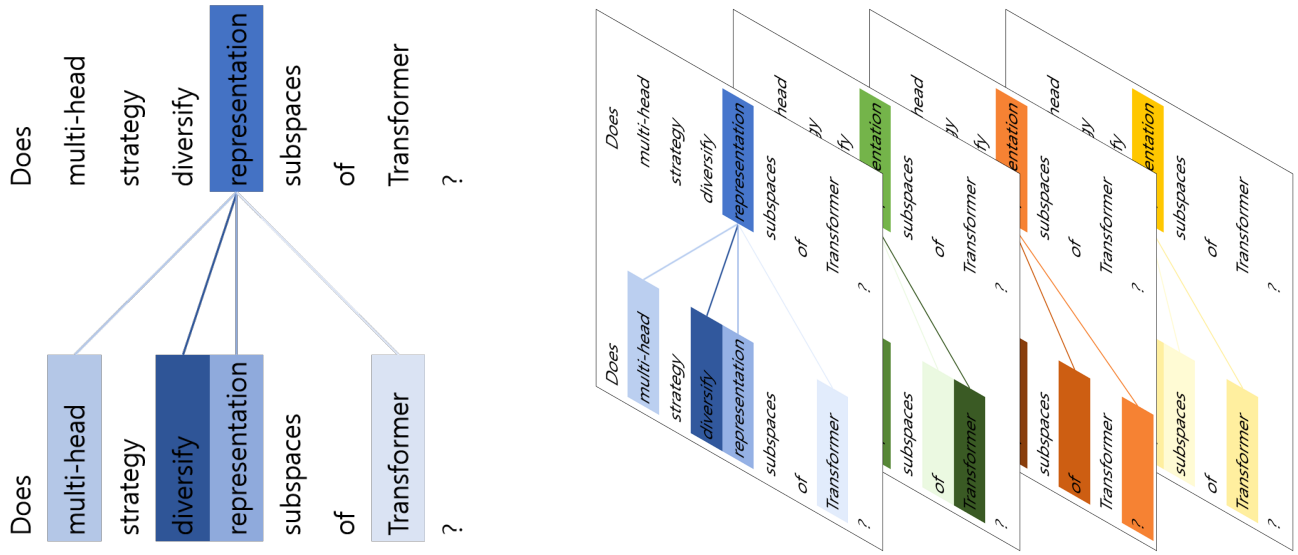
where L is a length of key and value matrix and d is a hidden dimension size. The single-head attention first computes attention weights by taking softmax function onto similarity scores between a query vector and key matrix, then finally operates multiplication with value matrix which can be considered as a pooling operation from a value matrix with the attention weights.

On the other hand, multi-head attention operates H-many single-head attentions in parallel with $q_i \in \mathbb{R}^{d_h}$, $K_i \in \mathbb{R}^{L\times d_h}$, $V_i \in \mathbb{R}^{L\times d_h}$, where q_i, K_i, V_i are projections of q, K, V onto smaller dimension d_h with weight matrices W_i^q, W_i^K, $W_i^V \in \mathbb{R}^{d\times d_h}$ respectively for each i-th head. The output of multi-head attention is calculated by concatenating all outputs of H-many heads followed by final linear projection;

$$X = [X_1, ..., X_H]W^O, \tag{2}$$

where X_i indicates an output of the i-th head and $W^O \in \mathbb{R}^{d\times d}$ is a weight matrix.

Figure 1 shows examples of self-attention weights in Transformer model. Given the query word "*representation*", the single-head attention module outputs attention weights for other words (a). On the other hand, each head in multi-head attention assigns different weights for other words as each head has its own weight matrix (b).



(**a**) Single-head attention (**b**) Multi-head attention

Figure 1. Visualization of attention weights in single-head attention and multi-head attention. Each head in multi-head attention assigns different weights to each word.

Although it has been believed that multi-head attention diversifies representation subspaces, measuring the similarity among deep representations of each head has been rarely studied. Measuring the inter-head similarity requires taking account of heads' response over the entire dataset. To do so, we adopt the following advanced tools for measuring similarity of representations in neural networks.

3.2. Singular Vector Canonical Correlation Analysis (SVCCA)

To measure the similarity between two deep representations, Raghu et al. [8] have amalgamated Canonical Correlation Analysis (CCA) with Singular Value Decomposition (SVD) into a novel method, Singular Vector Canonical Correlation Analysis (SVCCA). Raghu et al. [8] has claimed that SVCCA is invariant to affine transform, hence it can measure the similarity between unaligned deep representations.

SVCCA proceeds in two steps to seek correlation coefficients between two deep representations with N samples X_i and $X_j \in \mathbb{R}^{N \times d}$; (1) SVCCA performs SVD of each representation to pick out core representations, then (2) computes CCA of the core representations. Resulting SVCCA coefficients ρ_{ij} are computed as follows;

$$\rho_{ij} = \max_{a,b} corr(a^T U_i X_i, b^T U_j X_j), \tag{3}$$

where U_i and U_j are left orthogonal matrices computed from SVD of X_i and X_j respectively. SVCCA similarity between two deep representations using SVCCA is defined as a mean value over top SVCCA coefficients with a threshold such that covers all meaningful subspaces. In this paper, we measure inter-head similarity by averaging SVCCA similarity between two heads over all possible pairs of heads.

3.3. Centered Kernel Alignment (CKA)

Kornblith et al. [9] have introduced Centered Kernel Alignment (CKA) as a similarity measure between deep representations. The authors have pointed out a limitation of SVCCA that it is invariant to invertible linear transformation when dimension size exceeds the number of data, whereas CKA shows robustness regardless of a small number of data N.

CKA is calculated by normalizing an index of Hilbert-Schmidt Independence Criterion (HSIC) [27] in order to keep invariance to isotropic scaling. For a pair of heads

$X_i = (x_{i1}, x_{i2}..., x_{iN})^T$ and $X_j = (x_{j1}, x_{j2}..., x_{jN})^T$, we can define two matrices $K_{ikl} = \kappa(x_{ik}, x_{il})$ and $K_{jkl} = \kappa(x_{jk}, x_{jl})$ where κ is kernel function. Then HSIC between two heads is computed as follows;

$$\text{HSIC}(K_i, K_j) = \frac{1}{(N-1)^2} tr(K_i C K_j C), \tag{4}$$

where C is a centering matrix $C_N = I_N - \frac{1}{N}\mathbf{1}\mathbf{1}^T$, where $\mathbf{1}$ is a vector of ones. CKA of a pair of heads is computed by normalizing HSIC [28,29];

$$\text{CKA}(K_i, K_j) = \frac{\text{HSIC}(K_i, K_j)}{\sqrt{\text{HSIC}(K_i, K_i)\text{HSIC}(K_j, K_j)}}. \tag{5}$$

Finally, we define inter-head similarity using CKA as an average value over CKA of every possible pair of heads;

$$\text{CKA}_{\text{multi}} = \frac{1}{\#\text{ of } pairs} \sum_{i<j} \text{CKA}(K_i, K_j). \tag{6}$$

In this paper, CKA similarity is used as not only a tool for analyzing inter-head diversity as well as SVCCA statistics but also an augmented loss to control inter-head diversity.

4. Methods for Controlling Inter-Head Diversity

In this section, we inspect three methods for multi-head attention to finely control inter-head diversity in training. Our three methods are architecture-agnostic, task-agnostic, and able to fine-tune so that they can be easily applied to any existing models with multi-head attention.

4.1. HSIC Regularizer

Because Kornblith et al. [9] have demonstrated that CKA robustly performs even with a small number of samples, we exploit it directly as an augmented loss term to enforce inter-head representations to be diverse. While SVCCA similarity is inappropriate for a regularizer term to be used in training because it requires many samples, CKA can properly operate within samples randomly drawn from a mini-batch. Since CKA is fully differentiable function and its gradient can be properly back-propagated through neural networks, we can directly use CKA as an additional loss term in training. As directly optimizing CKA loss, we expect representational subspaces of multi-head attention to be diverse.

To prevent high computational cost in training, we only compute HSIC term (Equation (4)) as an augmented loss. Our HSIC regularizer term L_{hsic} is computed by average of HSIC values with every possible pair of heads as follows;

$$L_{\text{hsic}} = \lambda_{hsic} \cdot \frac{1}{\#\text{ of } pairs} \sum_{i<j} \text{HSIC}(X_i, X_j), \tag{7}$$

where X_i is a representation of the i-th head. HSIC is zero when two variables are independent, hence we expect that HSIC regularizer increases inter-head diversity by minimizing L_{hsic} in training.

4.2. Orthogonality Regularizer

We also revisit the orthogonality loss [7] which adds disagreement term on between heads' representations. The disagreement term can be interpreted as a weak orthogonal constraint term since it is computed by cosine similarity between V_{li} and V_{lj}, where V_{li} is the l-th vector in the i-th head. Therefore, the disagreement term orthogonalizes an orientation through minimizing the cosine similarity. We apply the disagreement term to

q, K, and V in our model, assuming that it can give variation to inter-head diversity with SVCCA and CKA.

In line with orthogonality regularization, Bansal et al. [25] have suggested Spectral Restricted Isometry Property (SRIP) regularization as a stricter orthogonal constraint. SRIP regularizer minimizes a spectral norm of orthogonality to its target matrix more strictly because the spectral norm requires all singular values of its target matrix to be close to 1. Thus, by utilizing both SRIP regularizer and the disagreement regularizer, we suggest an orthogonality regularizer for multi-head attention as a tool for controlling inter-head diversity. Our orthogonality term L_{ortho} is computed as follows. We first build V_{all} by collecting every l-th vector of *value* matrix V in every i-th head, $V_{all} = [V_0, ..., V_i, ..., V_H]$. Then, we take SRIP of V_{all};

$$L_{\text{ortho}} = \lambda_{ortho} \cdot \sigma(V_{all}^T V_{all} - I), \tag{8}$$

where $\sigma(W)$ is the spectral norm of W.

Surprisingly, although Li et al. [7] has claimed that the disagreement regularizer encourages inter-head diversity, we find it slightly decreases inter-head diversity measured with SVCCA and CKA. However, instead of encouraging inter-head diversity, we observe that the orthogonality regularizer increases top-few SVCCA coefficients that can be regarded as core representations. We report detailed results and discussion in Section 6.

4.3. Drophead

We also inspect Drophead [12] as the very naive but effective method to control the diversity. Zhou et al. [12] have introduced Drophead as a regularizing method in order to reduce overfitting similar to Dropout [13]. Zhou et al. have introduced Drophead as a method that drops an entire attention head during training and shown that the Drophead improves the model's robustness and performance with carefully scheduled dropout rate. Unlike Zhou et al., we mainly focus on how Drophead controls and diversifies the inter-head similarity. We use more naive Drophead method that randomly zero-out each head in training with a dropout rate γ, a real value ranged from 0.0 to 1.0. Our Drophead only requires a scalar hyperparameter γ while a model can keep its architecture identical. Also, our Drophead can be applied to training without additional computational cost.

Drophead reduces the *effective* number of heads by randomly dropping it out in training, hence it operates similarly to *decreasing number of heads* in training and decreases inter-head diversity. Simultaneously, applying Drophead can benefit the advantages of Dropout as well as Zhou et al. have shown. In our experiments, Drophead is applied independently to Dropout.

5. Inter-Head Similarity Analysis

In this section, we investigate how SVCCA and CKA values change with respect to the number of heads. By analyzing the diversity of representation subspaces, we show that how SVCCA and CKA reflect the dynamics of inter-head similarity in terms of the numbers of heads.

5.1. Experimental Details for Similarity Analysis

- **Data and Setups:** We choose De→En IWSLT17 machine translation task [14] for our analysis in this section. Training set consists of 223,162 sentences, development set consists of 8130 sentences, and test set consists of 1116 sentences. To tokenize the corpus, we use Byte Pair Encoding [30] with a vocabulary size of 16,384. We use Transformer [1] architectures with various numbers of heads and hidden dimension sizes for comparison. For all models, we use 6 layers for encoder's self-attention, decoder's self-attention, and encoder-decoder attention modules.
- **Performances of trained models:** Table 1 shows BLEU scores of models with various hidden dimension sizes and numbers of heads. As represented in Table 1, increasing hidden size d results in higher BLEU performances with a fixed number of heads,

although increasing the number of heads does not always assure higher performance with fixed hidden size.

Table 1. BLEU scores comparison with various hidden size d and *number of head* H on IWSLT17 De→En corpus.

Hidden Size d	***Number of Heads*** H				
	1	**2**	**4**	**8**	**16**
64	26.33	27.47			
128	31.30	32.75	31.71		
256	32.63	33.23	33.42	33.62	
512	33.33	33.42	33.90	33.67	32.69

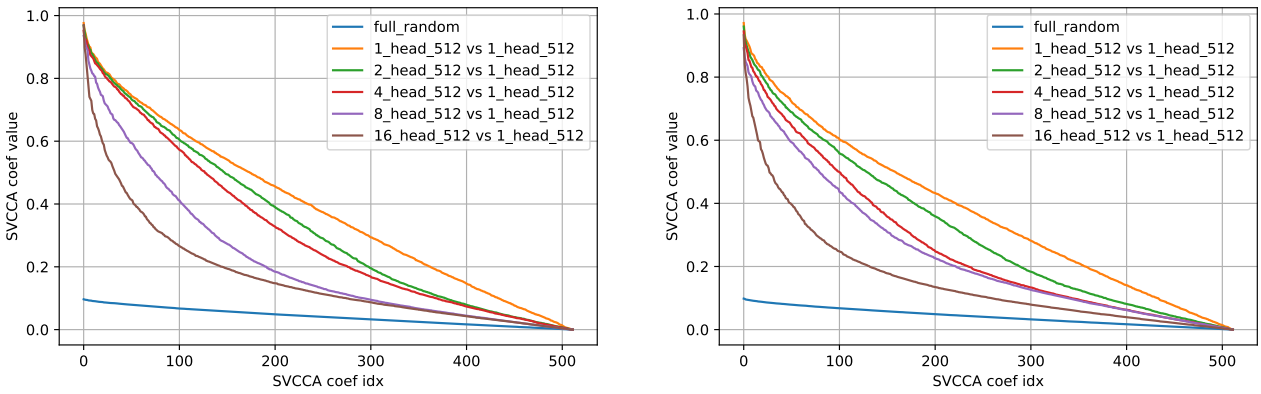
5.2. Applying SVCCA and CKA

In order to verify whether the multi-head strategy affects models' representation subspaces, we examine SVCCA statistics between representations of heads in each model. To utilize SVCCA and CKA, we collect responses $X = [X^1, ..., X^N]$ of each head at the last layers of three modules (encoder's self-attention, decoder's self-attention, and encoder-decoder attention) from development dataset consisting of *num_sentence* sentences, so that we have $N = num\_sentence \times token\_per\_sentence$ many d-dimensional vectors. We compare nine models with a number of heads $h = \{2, 4, 8, 16\}$ and hidden size $d = \{64, 128, 256, 512\}$ in order to examine how those architectural parameters change inter-head diversity. We report our results of the last layer of the encoder-decoder attention module only, yet we find the same tendency through every layer of every module.

5.3. Analysis on Inter-Model Similarity

We first examine SVCCA statistics of representations of five models versus representation of a single-headed model. We compare five models with varying numbers of heads ($H = 1, 2, 4, 8$, and 16) and fixed hidden size d as 512.

As shown in Table 2, SVCCA similarities between multi-headed models and a single-headed model, we can see that the response of a model is getting more dissimilar to a single-headed model as the number of heads increases. SVCCA coefficient curves also show similar results in Figure 2. SVCCA coefficients drop more rapidly with large number of heads in every layer. These results indicate that multi-head strategy can induce a model to find some representations uncorrelated to a single-headed model while its core representations remain, as shown as top few SVCCA coefficients are high.



(**a**) Encoder's self-attention (**b**) Encoder-Decoder attention

Figure 2. Singular Vector Canonical Correlation Analysis (SVCCA) coefficient curves versus a single headed model.

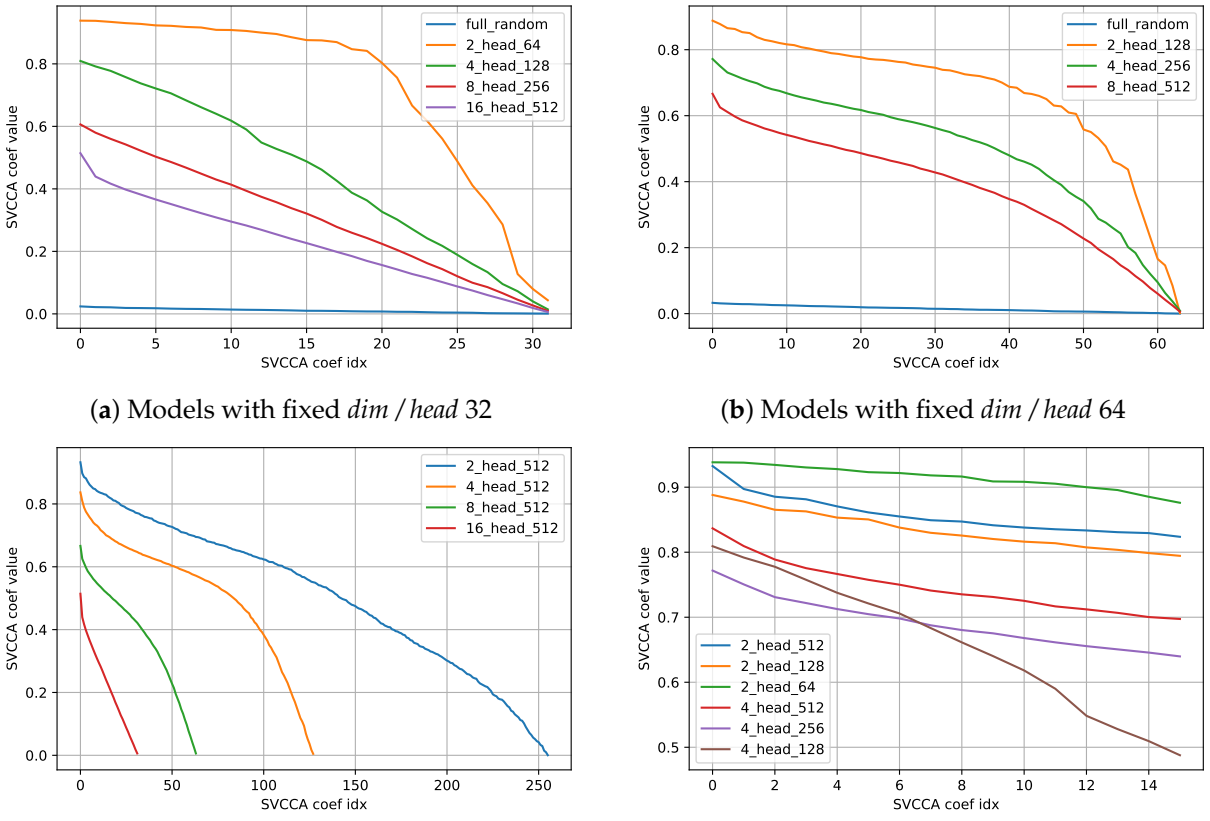
Table 2. SVCCA similarities versus a single headed model.

Modules	*Number of Heads H*				
	1	**2**	**4**	**8**	**16**
enc self	0.448	0.407	0.374	0.272	0.202
dec self	0.474	0.4	0.353	0.328	0.237
enc-dec	0.429	0.378	0.319	0.289	0.189

5.4. Does Multi-Head Strategy Diversify a Model's Representation Subspaces?

Table 3 shows inter-head similarity using SVCCA and CKA of each model. Both inter-head similarity measures using SVCCA and CKA show a persistent tendency that the inter-head similarity of each model decreases as the number of heads increases. On the other hand, we observe that increasing hidden dimension size d does not meaningfully affect the inter-head similarity with a fixed number of heads.

In addition to Table 3, we plot SVCCA coefficient curves of inter-head similarity in Figure 3. With various number of heads $H = \{2, 4, 8, 16\}$ and fixed $dim/head = \{32, 64\}$ ((a) and (b) in Figure 3), we observe that increasing number of heads make SVCCA coefficients smaller, indicating that inter-head diversity also increases. We also observe the same tendency with fixed dim ((c) in Figure 3), while we cannot find any consistency of inter-head similarity with fixed number of heads ((d) in Figure 3). Besides, we observed an interesting feature of SVCCA similarity curves that well-trained models have steep slopes on top-few SVCCA coefficients. We later discuss the steepness of top-few SVCCA coefficients in Section 6. Our analysis of inter-head similarity measured by SVCCA and CKA statistically support the hypothesis that multi-head attention diversifies deep representations.



(**a**) Models with fixed *dim* / *head* 32

(**b**) Models with fixed *dim* / *head* 64

(**c**) Models with fixed *dim* 512

(**d**) Top 16 SVCCA coefficients with $H = \{2, 4\}$

Figure 3. SVCCA coefficient curves of inter-head similarity.

Table 3. Inter-head similarity with various numbers of heads and hidden dimension.

Models	*Dim/Head*	SVCCA	CKA
2_H_64_d	32	0.793	0.553
2_H_128_d	64	0.712	0.488
2_H_512_d	256	0.559	0.344
4_H_128_d	32	0.504	0.277
4_H_256_d	64	0.541	0.309
4_H_512_d	128	0.560	0.277
8_H_256_d	32	0.346	0.143
8_H_512_d	64	0.419	0.197
16_H_512_d	32	0.252	0.117

6. Experiments on Controlling Inter-Head Similarity Methods

To examine how our methods affect multi-head attention, we analyze inter-head similarity statistics on De→En machine translation task with IWSLT17 corpus. We also report our results through extensive experiments on machine translation and language modeling tasks to empirically verify that our methods can make a model achieve higher performance than its baseline model.

6.1. Experimental Details

- **Data and Setups:** We test our proposed methods on machine translation tasks with De-En WMT17 corpus [31], Ru-En UN corpus, and Zh-En UN corpus [15]. For WMT17 and UN corpus, we sample 2.5 M sentences randomly from each training set for training and use the whole development/test sets, similar to the setup of Voita et al. [6]. Each corpus has development set consisting of 16,573 and 4000 sentences respectively and test set consisting of 3004 and 4000 sentences respectively. We also test our methods on a language modeling task with the Penn Treebank corpus [32]. We follow the rest of details as mentioned in Section 5
- **Model architectures:** We set a baseline model as an encoder-decoder Transformer with 6 layers, 512 hidden size, and 8 heads for every machine translation task. For language modeling, we use only the decoder part of Transformer only with 2 layers, 256 hidden sizes, and 4 heads. For each model named with ORTHO and HSIC, we add each regularization term L_{ortho} and L_{hsic} to Transformer's default loss term. We choose the value of hyperparameters *Drophead rate*, λ_{ortho} and $lambda_{hsic}$ by grid search on De→En IWSLT17 task; *Drophead rate*$= 0.1$, $\lambda_{ortho} = 1.0$, and $\lambda_{hsic} = 10^{-7}$. We apply the same values for other models.

6.2. Analysis on Controlling Inter-Head Diversity

We report the performances of our suggested methods in Table 4 and the controlled inter-head similarity with our suggested methods in Table 5. We also plot SVCCA coefficient curves in Figure 4.

With Drophead, all models show increased inter-head similarity compared to the baseline. As γ increases to 0.0 to 0.5, inter-head similarity indeed increases to 0.397 to 0.709, indicating that Drophead affects inter-head similarity by reducing the number of *effective* heads as desired. We observe this clear tendency by comparing SVCCA coefficient curves (a) in Figure 4 to (b) in Figure 3. The curve of 8_H_512_d with $\gamma = 0.3$ is very similar to that of 4_H_256_d, and as the rate increases $\gamma = 0.5$, the curve becomes similar to that of the model with fewer heads 2_H_128_d.

In addition, as opposed to the expectation of Li et al. [7] have expected, we find that the orthogonality loss does not diversify inter-head similarity. For +ORTHO and +HSIC, every model shows average disagreement score [7] as 0.999, which implies that two vectors from different heads are orthogonal. However, instead of diversifying, the

orthogonality loss slightly increases inter-head similarity measured in both SVCCA (from 0.397 to 0.420) and CKA (from 0.199 to 0.366). Nevertheless, the model only with the orthogonality loss performs better than a baseline as it records 34.03 BLEU score (+ORTHO ONLY in Table 4). We suspect that the performance improvements are caused by steep rises of top-few SVCCA coefficients. The affects of the orthogonality loss on top-few SVCCA coefficients are depicted in (b) and (d) in Figure 4 (as comparing curves of *baseline, ortho 0.1, ortho 1.0, and ortho 10.0*). The orthogonality regularizer makes the heads similar to each other in a prime direction while sustaining other directions diverse, hence it makes the model robust to both general features and rare features.

Lastly, we observe that HSIC regularizer directly enforces each head to be diverse as shown in both Table 5 and (c) in Figure 4. While the other two methods increase inter-head similarity, HSIC regularizer is the only method to diversify inter-head similarity without modifying a model's architecture. Although *increasing number of heads H* also diversify inter-head similarity, it has a critical downside that architectural modification must be accompanied.

Table 4. BLEU evaluation with controlled inter-head similarity on En-De IWSLT17 corpus.

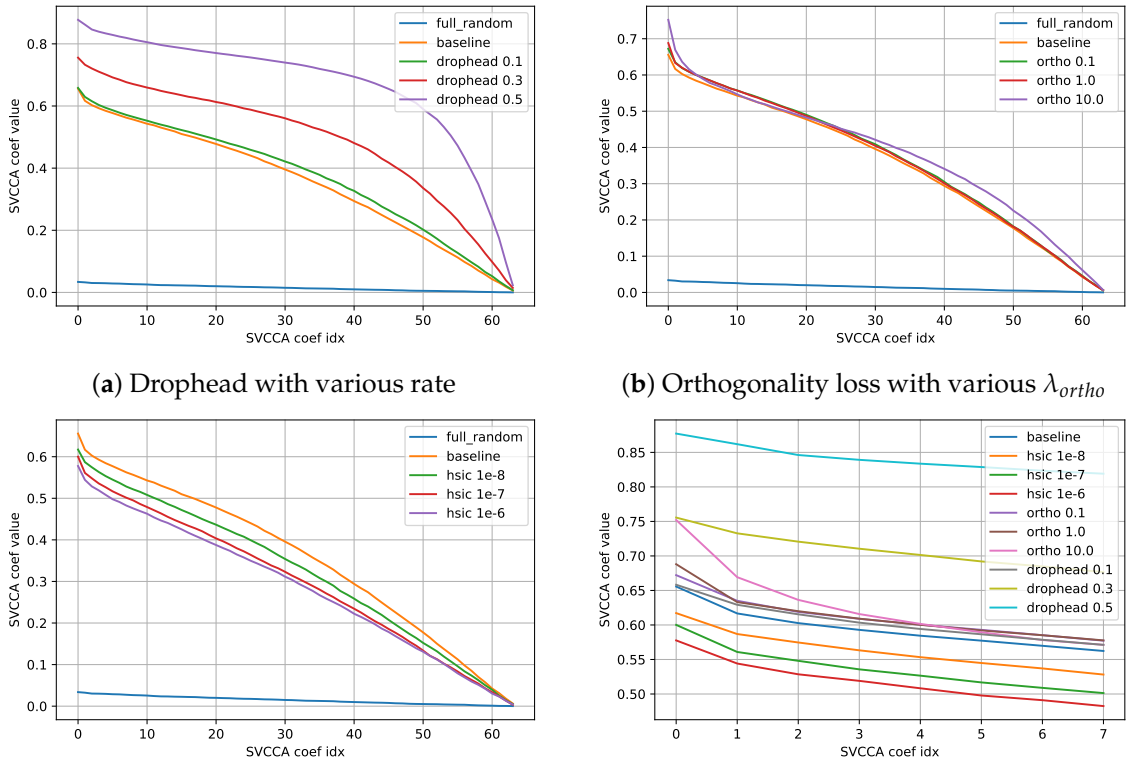
Models	Language Pairs	
	De→En	En→De
Baseline Transformer	33.67	29.76
+ DROPHEAD ONLY	34.26	30.13
+ ORTHO ONLY	34.03	30.27
+ HSIC ONLY	34.43	30.32
+ALL	**34.53**	**30.38**

Table 5. Controlled inter-head similarity with suggested methods.

Models		SVCCA	CKA
Baseline Transformer		**0.397**	**0.199**
+DROPHEAD	*0.1*	0.415	0.207
+DROPHEAD	*0.3*	0.534	0.317
+DROPHEAD	*0.5*	0.709	0.527
+ORTHO	*0.1*	0.408	0.208
+ORTHO	*1.0*	0.407	0.223
+ORTHO	*10.0*	0.420	0.366
+HSIC	10^{-8}	0.364	0.182
+HSIC	10^{-7}	0.338	0.158
+HSIC	10^{-6}	0.325	0.125

6.3. Quantitative Evaluation

We report BLEU scores on every language pairs in Tables 4 and 6. These results support our hypothesis that a multi-head attention model can extend its own capability by controlling inter-head diversity with our suggested methods. Models with all three suggested methods applied (+ALL) show the best performances on every language pair.



(**a**) Drophead with various rate

(**b**) Orthogonality loss with various λ_{ortho}

(**c**) HSIC loss with various λ_{hsic}

(**d**) Top 8 SVCCA coefficients

Figure 4. SVCCA coefficient curves of inter-head similarity with controlling methods.

Table 6. BLEU evaluation on various language pairs with controlled inter-head similarity on WMT17 corpus and UN corpus.

Models	Language Pairs					
	De→En	**En→De**	**Ru→En**	**En→Ru**	**Zh→En**	**En→Zh**
Baseline Transformer	31.47	25.69	52.68	44.62	52.21	47.08
+ DROPHEAD ONLY	31.69	25.73	52.90	44.82	52.60	47.09
+ ORTHO ONLY	31.63	25.86	52.92	44.86	52.55	47.28
+ HSIC ONLY	31.57	25.89	52.89	44.82	52.54	47.16
+ALL	**31.76**	**25.91**	**53.02**	**45.23**	**52.69**	**47.33**

We also verify the effect of our suggested methods on language modeling task in order to show that our methods can be applied to tasks other than machine translation. Table 7 shows perplexity score on language modeling task with PTB corpus. As well as on the encoder-decoder Transformer, our methods applied to the decoder-only Transformer also increases its performance on the language modeling task. Applying +HSIC ONLY shows the best performance, even better than applying all methods. Nevertheless, all of our methods clearly improve the perplexity of the decoder-only Transformer. The experimental results show that our methods can easily be applied to various model architectures that use multi-head attention. Note that our suggested methods and our analyses in Section 5 do not relate to the size of the model (i.e., the hidden size or the number of layers). We strongly believe that our methods can be applied to larger language models such as BERT [2] or XLM-R [33], because they also exploit the multi-head attention as the same way as the Transformer model in our experiments.

Table 7. Perplexity with controlled inter-head similarity on PTB language modeling.

Models	Perplexity
Baseline Transformer	120.38
+ DROPHEAD ONLY	102.72
+ ORTHO ONLY	102.62
+ HSIC ONLY	**101.89**
+ALL	102.07

7. Conclusions

In this paper, we analyze the inter-head similarity of multi-head attention using SVCCA and CKA to unveil representation of each heads' subspaces. We show an empirical proof that multi-head attention diversifies its representations as the number of heads increases. Based on our observation, we hypothesize that there is an optimal degree of inter-head diversity that fully utilizes a model's capability. Then, we introduce three methods to control the degree of inter-head diversity; (1) HSIC regularizer, (2) the orthogonality regularizer revisited, and (3) Drophead method. The three methods are all able to fine-tune the inter-head diversity without architectural change. We show that HSIC regularizer diversifies the inter-head diversity and Drophead works the other way, whereas the orthogonality regularizer gathers the core representations of multi-head attention. Finally, we empirically show that controlling inter-head diversity can make the model utilize its own capability better resulting in higher performances on various machine translation and language modeling tasks. Our methods to control inter-head diversity can be easily applied to every model that uses multi-head attention including Transformer, BERT, and XLNet.

Author Contributions: Conceptualization, H.Y. and K.J.; methodology, H.Y.; software, H.Y. and T.K.; validation, H.Y., T.K., and K.J.; formal analysis, H.Y.; investigation, H.Y. and T.K.; resources, H.Y.; data curation, H.Y. and T.K.; writing—original draft preparation, H.Y.; writing—review and editing, H.Y., T.K., and K.J.; visualization, H.Y.; supervision, K.J. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Samsung Electronics. This work was also supported by the BK21 FOUR program of the Education and Research Program for Future ICT Pioneers, Seoul National University in 2021. This work was also supported by the Automation and Systems Research Institute (ASRI), Seoul National University.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: None.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
2. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. 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), Minneapolis, MN, USA, 3–7 June 2019; pp. 4171–4186.
3. Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R.R.; Le, Q.V. Xlnet: Generalized autoregressive pretraining for language understanding. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; pp. 5754–5764.
4. Zhang, X.; Wei, F.; Zhou, M. HIBERT: Document Level Pre-training of Hierarchical Bidirectional Transformers for Document Summarization. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; pp. 5059–5069.

5. Clark, K.; Khandelwal, U.; Levy, O.; Manning, C.D. What Does BERT Look at? An Analysis of BERT's Attention. In Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, Florence, Italy, 29 July–2 August 2019; pp. 276–286.
6. Voita, E.; Talbot, D.; Moiseev, F.; Sennrich, R.; Titov, I. Analyzing Multi-Head Self-Attention: Specialized Heads Do the Heavy Lifting, the Rest Can Be Pruned. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics,Florence, Italy, 28 July–2 August 2019; pp. 5797–5808.
7. Li, J.; Tu, Z.; Yang, B.; Lyu, M.R.; Zhang, T. Multi-Head Attention with Disagreement Regularization. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 2897–2903.
8. Raghu, M.; Gilmer, J.; Yosinski, J.; Sohl-Dickstein, J. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 6076–6085.
9. Kornblith, S.; Norouzi, M.; Lee, H.; Hinton, G. Similarity of Neural Network Representations Revisited. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; pp. 3519–3529.
10. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv* **2016**, arXiv:1603.04467.
11. Bau, A.; Belinkov, Y.; Sajjad, H.; Durrani, N.; Dalvi, F.; Glass, J. Identifying and controlling important neurons in neural machine translation. *arXiv* **2018**, arXiv:1811.01157.
12. Zhou, W.; Ge, T.; Xu, K.; Wei, F.; Zhou, M. Scheduled DropHead: A Regularization Method for Transformer Models. *arXiv* **2020**, arXiv:2004.13342.
13. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
14. Cettolo, M.; Federico, M.; Bentivogli, L.; Jan, N.; Sebastian, S.; Katsuitho, S.; Koichiro, Y.; Christian, F. Overview of the iwslt 2017 evaluation campaign. In Proceedings of the International Workshop on Spoken Language Translation, Okyo, Japan, 14–15 December 2017; pp. 2–14.
15. Ziemski, M.; Junczys-Dowmunt, M.; Pouliquen, B. The united nations parallel corpus v1.0. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), Portorož, Slovenia, 23–28 May 2016; pp. 3530–3534.
16. Tang, G.; Sennrich, R.; Nivre, J. An Analysis of Attention Mechanisms: The Case of Word Sense Disambiguation in Neural Machine Translation. *arXiv* **2018**, arXiv:1810.07595.
17. Raganato, A.; Tiedemann, J. An analysis of encoder representations in transformer-based machine translation. In Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, Brussels, Belgium, 31 October–4 November 2018; pp. 287–297.
18. Michel, P.; Levy, O.; Neubig, G. Are Sixteen Heads Really Better than One? In *Advances in Neural Information Processing Systems*; Vancouver, Canada, 8-14 December, 2019; pp. 14014–14024.
19. Li, Y.; Yosinski, J.; Clune, J.; Lipson, H.; Hopcroft, J.E. Convergent learning: Do different neural networks learn the same representations? In Proceedings of the FE@ NIPS, Montreal, QC, Canada, 7–12 December 2015; pp. 196–212.
20. Morcos, A.; Raghu, M.; Bengio, S. Insights on representational similarity in neural networks with canonical correlation. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 3–8 December 2018; pp. 5727–5736.
21. Maheswaranathan, N.; Williams, A.; Golub, M.; Ganguli, S.; Sussillo, D. Universality and individuality in neural dynamics across large populations of recurrent networks. *Adv. Neural Inf. Process. Syst.* **2019**, *2019*, 15603–15615.
22. Kudugunta, S.R.; Bapna, A.; Caswell, I.; Arivazhagan, N.; Firat, O. Investigating multilingual nmt representations at scale. *arXiv* **2019**, arXiv:1909.02197.
23. Rodríguez, P.; Gonzalez, J.; Cucurull, G.; Gonfaus, J.M.; Roca, X. Regularizing cnns with locally constrained decorrelations. *arXiv* **2016**, arXiv:1611.01967.
24. Xie, D.; Xiong, J.; Pu, S. All you need is beyond a good init: Exploring better solution for training extremely deep convolutional neural networks with orthonormality and modulation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6176–6185.
25. Bansal, N.; Chen, X.; Wang, Z. Can we gain more from orthogonality regularizations in training deep CNNs? In Proceedings of the 32nd International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 3–8 December 2018; pp. 4266–4276.
26. Gu, S.; Hou, Y.; Zhang, L.; Zhang, Y. Regularizing Deep Neural Networks with an Ensemble-based Decorrelation Method. In Proceedings of the IJCAI, Stockholm, Swede, 13–19 July 2018; pp. 2177–2183.
27. Gretton, A.; Bousquet, O.; Smola, A.; Schölkopf, B. Measuring statistical dependence with Hilbert-Schmidt norms. In *International Conference on Algorithmic Learning Theory*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 63–77.
28. Cortes, C.; Mohri, M.; Rostamizadeh, A. Algorithms for learning kernels based on centered alignment. *J. Mach. Learn. Res.* **2012**, 13, 795–828.
29. Cristianini, N.; Shawe-Taylor, J.; Elisseeff, A.; Kandola, J.S. On kernel-target alignment. *Adv. Neural Inf. Process. Syst.* **2002**, *14*, 367–373.

30. Sennrich, R.; Haddow, B.; Birch, A. Neural Machine Translation of Rare Words with Subword Units. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Berlin, Germany, 7–12 August 2016; pp. 1715–1725.
31. Ondrej, B.; Chatterjee, R.; Christian, F.; Yvette, G.; Barry, H.; Matthias, H.; Philipp, K.; Qun, L.; Varvara, L.; Christof, M.; et al. Findings of the 2017 conference on machine translation (wmt17). In Proceedings of the Second Conference onMachine Translation, Copenhage, Denmark, 7–8 September 2017; pp. 169–214.
32. Marcus, M.; Santorini, B.; Marcinkiewicz, M.A. Building a Large Annotated Corpus of English: The Penn Treebank; University of Pennsylvania: Philadelphia, PA, USA,1993.
33. Conneau, A.; Khandelwal, K.; Goyal, N.; Chaudhary, V.; Wenzek, G.; Guzmán, F.; Grave, E.; Ott, M.; Zettlemoyer, L.; Stoyanov, V. Unsupervised cross-lingual representation learning at scale. *arXiv* **2019**, arXiv:1911.02116.