2023 57th Asilomar Conference on Signals, Systems, and Computers | 979-8-3503-2574-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/IEEECONF59524.2023.10476895

SEQ2SEQ-SC: END-TO-END SEMANTIC COMMUNICATION SYSTEMS WITH PRE-TRAINED LANGUAGE MODEL

Ju-Hyung Lee[*] *Dong-Ho Lee*[*] *Eunsoo Sheen* *Thomas Choi* *Jay Pujara*

University of Southern California

ABSTRACT

In this work, we propose a realistic semantic network called seq2seq-SC, designed to be compatible with 5G NR and capable of working with generalized text datasets using a pre-trained language model. The goal is to achieve unprecedented communication efficiency by focusing on the meaning of messages in semantic communication. We employ a performance metric called semantic similarity, measured by BLEU for lexical similarity and SBERT for semantic similarity. Our findings demonstrate that seq2seq-SC outperforms previous models in extracting semantically meaningful information while maintaining superior performance. This study paves the way for continued advancements in semantic communication and its prospective incorporation with future wireless systems in 6G networks.

Index Terms— Semantic communication, natural language processing (NLP), link-level simulation.

I. INTRODUCTION

The recent rise of deep learning-based techniques to infer semantics (*i.e.,* the meaning of the message rather than the message itself) from texts, speeches, and videos, as well as the ever-increasing quality of service requirements for extremely data-hungry applications such as extended reality (XR), have motivated the use of semantic communication [1] for a new generation of wireless systems (6G). Focusing on semantics allows forgoing unnecessary data (*e.g.,* articles in a sentence or background in a portrait photo), which can increase communication efficiency.

While semantic communication may bring unprecedented benefits, many challenges remain to realize it for actual usage. First, it must be compatible with existing communication infrastructure; a "link-level" simulation is hence required to verify its realistic end-to-end (E2E) performance. Second, the semantic network has to be generalized to work with any dataset rather than a particular dataset. Third, since there is no universal performance metric for semantic communication yet, metrics such as semantic similarity must be refined to evaluate performance of the semantic network from the semantic point of view. Lastly, since classic communication cannot be completely replaced by semantic communication, the network must be able to deliver information both as-is or modified with high semantic similarity dependent upon the communication scenario.

Contributions. We revisit questions raised by DeepSC [2] regarding semantic communication:

Q 1: *How do we design the semantic and channel coding jointly?*

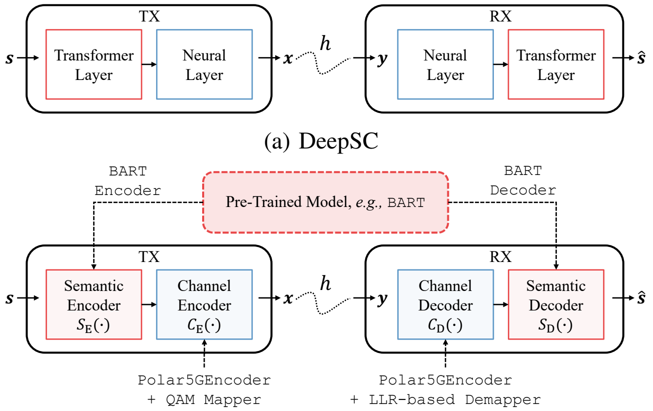
Q 2: *How do we measure semantic error (similarity) between transmitted and received sentences?*

Our main contributions, which address these questions, are summarized as follows:

- We employ E2E link-level simulation compliant to 5G NR (NVIDIA Sionna [3]), which contains features like Polar codes. Through such method, we validate semantic network performance in real-world settings and answer **Q 1**.
- We integrate the pre-trained encoder-decoder transformers with the E2E semantic communication systems, which efficiently extract the semantic (meaningful) information with reduced computation effort, dubbed seq2seq-SC. This network is "generalized", meaning it works with all (general) text corpus in comparison to Deep-SC which has limitations dependent upon a particular datasets.
- To answer **Q 2** and evaluate performance of a semantic network in a *semantic way*, we introduce a metric called semantic similarity. In order to make the network flexible with respect to the communication scenario, the network may either prioritize delivering a message as perfect as possible or on semantic similarity only.

II. PRE-TRAINED MODEL FOR LANGUAGE

Contextualized embeddings from pre-training in a self-supervised manner (*e.g.,* masked language modeling) with transformers [4] are extremely effective in providing initial representations that can be refined to attain acceptable performance on numerous downstream tasks. Recent studies on text semantic communication exploit transformer architectures [4] to extract semantics at transmitter and recover the original information at receiver [2], [5]. However, such frameworks may have following challenges: (1) Training an E2E semantic communication pipeline requires a huge computational effort due to the many randomly initialized parameters of semantic encoder/decoder to be trained on; (2) Difficulty in handling out-of-vocabulary (OOV) since they only use a set of whitespace-separated tokens in the training data. In this work, we use a pre-trained encoder-decoder


TX
s
Transformer Layer
Neural Layer
x
h
y
RX
Neural Layer
Transformer Layer
ŝ
BART Encoder
Pre-Trained Model, e.g., BART
BART Decoder
TX
s
Semantic Encoder $S_E(\cdot)$
Channel Encoder $C_E(\cdot)$
x
h
y
RX
Channel Decoder $C_D(\cdot)$
Semantic Decoder $S_D(\cdot)$
ŝ
Polar5GEncoder + QAM Mapper
Polar5GEncoder + LLR-based Demapper

(a) DeepSC

(b) seq2seq-SC (Proposed)

Fig. 1: Architecture of semantic communication systems. Our proposed seq2seq-SC follows the way of link-level simulation, where the channel en/decoder are composed of 5G-NR compliant modules, reflecting the actual symbol (or bit) transmission. In contrast, DeepSC comprises all main modules with neural layers.

transformer [6] to initialize the parameters of semantic encoder/decoder so that the pipeline itself requires little or no computational effort, and use pre-trained tokenizer to effectively handle OOV so that our pipeline can be generalized to any other text.

III. SEQ2SEQ-SC: SEMANTIC COMMUNICATION SYSTEMS WITH PRE-TRAINED MODEL

III-A. Problem Description

Consider a sentence $\boldsymbol{s}$ that maps to symbol stream $\boldsymbol{x}$:

$$\boldsymbol{x} = C_{\mathbf{E}}\left(S_{\mathbf{E}}\left(\boldsymbol{s}\right)\right), \tag{1}$$

where $C_{\mathbf{E}}(\cdot)$ and $S_{\mathbf{E}}(\cdot)$ represent the channel encoder and the semantic encoder, respectively. This symbol stream passes through a physical channel, h, with flat fading and noise in the RF front end of a receiver (RX); which is expressed by the received signal, $\boldsymbol{y}$:

$$\boldsymbol{y} = h\boldsymbol{x} + \boldsymbol{n}, \tag{2}$$

Here, the encoded signal by transmitter (TX) propagates over the Rayleigh fading channel with $\mathcal{CN}\left(0,1\right)$; RX receives the attenuated signal with $\boldsymbol{n} \sim \mathcal{CN}\left(0, \sigma_n^2\right)$. Then, $\boldsymbol{y}$ is decoded at the RX to estimate the sentence $\hat{\boldsymbol{s}}$:

$$\hat{\boldsymbol{s}} = S_{\mathbf{D}}\left(C_{\mathbf{D}}\left(\boldsymbol{y}\right)\right), \tag{3}$$

where $S_{\mathbf{D}}(\cdot)$ and $C_{\mathbf{D}}(\cdot)$ represent the semantic decoder and the channel decoder, respectively.

III-B. Architecture

The architecture of the semantic communication system is illustrated in Fig. 1, where the TX consists of $C_{\mathbf{E}}(\cdot)$ and $S_{\mathbf{E}}(\cdot)$ and RX consists of $C_{\mathbf{D}}(\cdot)$ and $S_{\mathbf{D}}(\cdot)$. For the TX side, the symbol stream $\boldsymbol{s}$ is first encoded with a semantic encoder that reduces the size of information by removing information or unnecessary for extracting the necessary information. Then, it is encoded with a channel encoder, which adds redundancy to quantized source information for reliable detection and correction of bit errors caused by the noisy channel, resulting in $\boldsymbol{x}$. Inversely, at the RX side, a channel decoder first decodes the received signal; and then the semantic decoder decodes the signal and extracts the symbol $\hat{\boldsymbol{s}}$.

Semantic En/Decoder. For semantic encoder and decoder, we consider a variant of a standard encoder-decoder transformer architecture consisting of two layer stacks in which the encoder is given an input sequence while the decoder generates a new output sequence [4]. Both the encoder and decoder are a stack of m transformer blocks, consisting of a self-attention layer and a fully-connected layer with residual connections, but the self-attention mechanism is different. The encoder uses a form of fully-visible self-attention mechanism that allows the model to attend to any entry of the input while the decoder uses a form of auto-regressive self-attention which only allows the model to attend to past outputs. These architectures can be pre-trained on a large scale corpus by corrupting documents and computing the cross entropy loss between the decoder's output and the original document to learn the model generalizable knowledge [6]. Here, we load such pre-trained checkpoints (BART [6]) and use the weights of encoder and decoder to initialize the weights of $S_{\mathbf{E}}(\cdot)$ and $S_{\mathbf{D}}(\cdot)$, respectively. Also, the pre-trained embedding $\mathcal{E}$ and the pre-trained tokenizer $\mathcal{T}$ are used and shared across $S_{\mathbf{E}}(\cdot)$ and $S_{\mathbf{D}}(\cdot)$. Once the sentence $\boldsymbol{s}$ is given to $S_{\mathbf{E}}(\cdot)$, $\mathcal{T}$ tokenizes $\boldsymbol{s}$ into tokens $\boldsymbol{s}_t = [s_{t_1}, s_{t_2}, ...s_{t_n}]$ and maps each token to embedding $\boldsymbol{s}_e = [s_{e_1}, s_{e_2}, ...s_{e_n}]$ by $\mathcal{E}$. Then, $S_{\mathbf{E}}(\cdot)$ encodes $\boldsymbol{s}$ into hidden states $\boldsymbol{r} = [r_1, r_2, ...r_n]$ based on $\boldsymbol{s}_e$ and passes it to channel encoder $C_{\mathbf{E}}(\cdot)$. Next, hidden states $\boldsymbol{r}'$, which are recovered from the channel decoder $C_{\mathbf{D}}(\cdot)$, are given to $S_{\mathbf{D}}(\cdot)$. Then, $S_{\mathbf{D}}(\boldsymbol{r}')$ defines the conditional distribution $p_{\theta_{S_{\mathbf{D}}}}\left(\hat{\boldsymbol{s}}_i \mid \hat{\boldsymbol{s}}_{0:i-1}, \boldsymbol{r}'\right)$ and auto-regressively samples words from the distribution for each index.

Channel En/Decoder. Polar codes, a form of linear block error correction codes, are one of the channel coding schemes in 5G-NR, where low complexity decoding is available [7]. We consider such practical channel coding modules, PolarEncoder and PolarDecoder as our channel encoder and decoder. In order to verify our semantic communication systems more practically, other modules (such as modulator and demodulator) are also chosen based on compatibility with 5G-NR. Details are in Table. I.

Regarding semantic communication, there are two main goals: (1) the minimization of semantic error, which corresponds to the maximization of semantic capacity (similarity); (2) reduce the number of symbols to be transmitted (*i.e.,* compression). In this paper, we only focus on minimizing semantic errors by maximizing semantic similarity, which is

Table I: Type (or parameter) for channel en/decoder and network scenario.

Type (or parameter)	Value
Channel en/decoder	Polar coding
# of information bits	512
# of codeword bits	1024
Coderate	0.5
Mapper/Demapper constellation	16-QAM
# of bit per symbol	4
Demapping method	Log-likelihood ratios
Channel	AWGN, Rayleigh fading

elaborated on in the following subsection.

III-C. Evaluation

In traditional communications, the performance of information transmission is evaluated by how accurately bits of 0 or 1 are transmitted (*e.g.,* bit error rate (BER)) or how well the symbol, which is a set of bits, is transmitted (*e.g.,* symbol error rate (SER)). In contrast, semantic communication focuses on *meaningful information*. Our view is aligned with the latter angle. To measure both lexical and semantic similarity, we use BLEU for lexical similarity, and SBERT for semantic similarity. Note that the higher the SBERT and BLEU scores (in-between $0 \sim 1$), the better.

BLEU. BLEU [8], originally proposed for machine translation, computes the average of the n-gram precision scores by counting the number of matches between n-grams of the input and the n-grams of the output, in a position independently [2], [5].

SBERT. Even though the lexical similarity between the input and output is low, the semantic similarity can be high. For example, *"child"* and *"children"* are semantically related, but the BLEU computed lexical similarity is zero. To compute such semantic similarity, we can represent sentences into embeddings using an embedding model $\boldsymbol{M}$ and compute the cosine similarity between them.

$$\operatorname{match}(\hat{\mathbf{s}}, \mathbf{s}) = \frac{\boldsymbol{M}(\mathbf{s}) \cdot \boldsymbol{M}(\hat{\mathbf{s}})^T}{\|\boldsymbol{M}(\mathbf{s})\| \, \|\boldsymbol{M}(\hat{\mathbf{s}})\|} \tag{4}$$

Existing semantic communication studies use BERT [9] as $\boldsymbol{M}$ to encode sentences into embeddings and compute the cosine similarity [2], [5]. However, the sentence embeddings from such pre-trained models without fine-tuning on semantic textual similarity task poorly capture semantic meaning of sentences due to anisotropic embedding space [10]. Here, we use SBERT [11], which is fine-tuned on semantic textual similarity tasks, to encode the sentence embedding.

III-D. Training

Our framework is trained by cross-entropy loss $\mathcal{L}_{\mathrm{CE}}$ which minimizes the discrepancy between predicted sentence $\hat{\boldsymbol{s}}$ and its original correct input sentence $\boldsymbol{s}$:

$$\begin{aligned} \mathcal{L}_{\mathrm{CE}}(\hat{\mathbf{s}}, \mathbf{s}) = -\sum\nolimits_{i=1} p\left(\mathbf{s}_{t_i}\right) \log\left(p\left(\hat{\mathbf{s}}_{t_i}\right)\right) + \\ \left(1 - p\left(\mathbf{s}_{t_i}\right)\right) \log\left(1 - p\left(\hat{\mathbf{s}}_{t_i}\right)\right) \end{aligned} \tag{5}$$

where $p\left(\mathbf{s}_{t_i}\right)$ is the true distribution, where the probability of the correct token for i-th index is 1 while other tokens are 0, and $p\left(\hat{\mathbf{s}}_{t_i}\right)$ is a predicted probability distribution over possible tokens for the i-th index. The framework is trained with batch size 4 and learning rate 5e-5.

IV. EXPERIMENTS

IV-A. Datasets

Following the conventional semantic communication works [2], [5], we use the European Parliament dataset [12], which consists of 2M sentences, for model training. Here, we let the input $\boldsymbol{s}$ and the output $\hat{\boldsymbol{s}}$ be the same for each sentence (*i.e.,* $\boldsymbol{s} \rightarrow \boldsymbol{s}$). Furthermore, to check whether the input $\boldsymbol{s}$ can be transferred to the receiver in a modified version with the same semantic meaning, we use 270K pairs of entailment relationship (*e.g., "A soccer game with multiple males playing."* ↔ *"Some men are playing a sport."*) in natural language inference data [13], [14]. Here, the output $\hat{\boldsymbol{s}}$ is different from $\boldsymbol{s}$ but the semantic meaning of $\hat{\boldsymbol{s}}$ and $\boldsymbol{s}$ are the same (*i.e.,* $\boldsymbol{s} \rightarrow \boldsymbol{s'}$). To evaluate the model, we randomly sample 1K sentences from image-caption dataset Flickr [15], which are not presented in the training data, to check the generalizability and superiority of our framework.

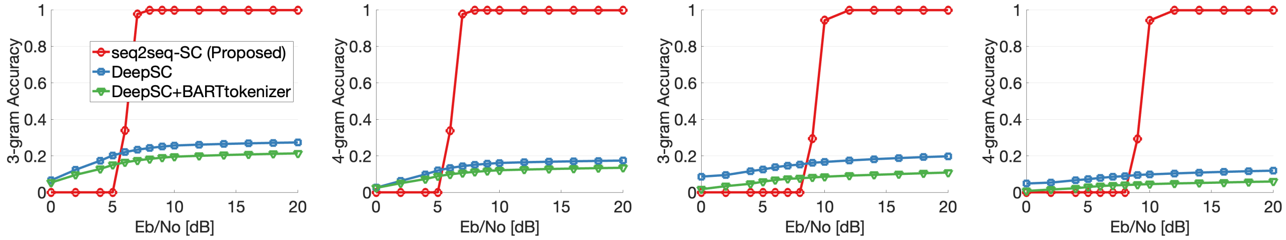
IV-B. Baselines

We compare our model with the following models: (1) **DeepSC** [2] consists of semantic en/decoder including multiple transformer en/decoder layers, respectively, while each channel en/decoder uses dense layers; that is, deep neural network (DNN)-based E2E physical layer communication systems. Framework tokenizes sentences in the training data and creates a set of tokens that can assign an embedding to each token in the training data; (2) **DeepSC+BARTtokenizer** is designed for a more fair comparison with our framework; it replaces the tokenizer (1) with a pre-trained BART tokenizer [6] to process tokens that are not in the training data; (3) **seq2seq-SC** is our main model consisting of semantic en/decoder initialized with pre-trained model checkpoint (`BART-base`), pre-trained BART tokenizer [6], and the channel en/decoder with Polar coding. Other modules in TX and RX (*e.g.,* mapper/demapper) are NR-5G compatible and configured based on the link level simulator `NVIDIA Sionna` [3].

IV-C. Experimental Results

Comparison Study: seq2seq-SC vs DeepSC. Fig. 2 shows the BLEU scores of DeepSC and our proposed seq2seq-SC for 1K sentences sampled from Flickr [15] not used for training. DeepSC achieves only about 10 to 20% 3,4-gram accuracy in the unseen corpus while our proposed seq2seq-SC outperforms it, achieving a near-perfect lexical similarity for $E_b/N_o \geq 6$ [dB]. This highlights the generalizability and superiority of our proposed framework.

It is worth noting that, for $E_b/N_o < 6$ [dB], however, the accuracy drops remarkably, as the channel begins to cause an error to the input of the semantic decoder even



(a) 3-gram, additive white Gaussian noise (AWGN)

(b) 4-gram, AWGN

(c) 3-gram, Rayleigh fading

(d) 4-gram, Rayleigh fading

Fig. 2: BLEU score over energy per bit to noise spectral density ratio (E_b/N_o) [dB] (AWGN, Rayleigh fading)

.

Transmit $\boldsymbol{s}$	An older dog and a younger one playing with a toy.		
Train	**Receive $\hat{\boldsymbol{s}}$**	**BLEU**	**SBERT**
$\boldsymbol{s} \to \boldsymbol{s}$	An older dog and a younger one playing with a toy.	1.0	1.0
$\boldsymbol{s} \to \boldsymbol{s'}$	Two dogs are playing with a toy.	0.210	0.820

Table II: Received output examples of seq2seq-SC trained by different dataset (*i.e.,* $\boldsymbol{s} \to \boldsymbol{s}$, $\boldsymbol{s} \to \boldsymbol{s'}$) for $E_b/N_o = 10\,$[dB]. BLEU and SBERT score show the lexical and semantic similarity between the transmitted input $\boldsymbol{s}$ and the received output $\hat{\boldsymbol{s}}$

Train	**Lexical Similarity**	**Semantic Similarity**
	BLEU [8]	SBERT [11]
$\boldsymbol{s} \to \boldsymbol{s}$	0.993	0.999
$\boldsymbol{s} \to \boldsymbol{s'}$	0.173	0.764

Table III: Lexical and semantic similarities of seq2seq-SC, trained by different training data, $E_b/N_o = 10\,$[dB].

after the error correction in the channel decoder. Here, the polar coding, which is our considered forward error correction (FEC) method in the channel en/decoder, achieves BER $> 10^{-3}$ for $E_b/N_o \simeq 5\,$[dB]. The out/input of semantic en/decoder is a tensor whose elements are FP32 single-precision floating points of 32 bits. The tensor, the token the semantic en/decoder exchange with, is particularly vulnerable to bit-wise errors (*e.g.,* bit flip); for instance, 1.0 can become ∞ even with a single flip of the second bit; that explains such a drastic accuracy drop. In this experiment, we utilized $\tanh(\cdot)$ function to map the elements of the tensor into $[-1, 1]$, but the accuracy in a low-signal-to-noise ratio (SNR) channel can be improved by introducing a better method to handle bit flip in tensors, and it is the subject of our future work.

Semantic Similarity. As aforementioned, the interoperability of delivering information both as-is or modified with high semantic similarity, dependent upon the communication scenario, is important. Table III corroborates the interoperability of the proposed seq2seq-SC. Since there is no universal performance metric for semantic communication yet, here, the semantic similarity is evaluated by several metrics from different semantic points of view. When the framework is trained to output the same sequence as the input (*i.e.,* $\boldsymbol{s} \to \boldsymbol{s}$), both the lexical and semantic similarities show a near-perfect score for $E_b/N_o = 10\,$[dB]. However, when the framework is trained to output a sentence that is semantically similar to the input (*i.e.,* $\boldsymbol{s} \to \boldsymbol{s'}$), the lexical similarity drops significantly while the semantic similarity remains similar, meaning it is a lexically different sentence but is semantically alike to the original sentence. It shows that our proposed system empowered by pre-trained `BART` model successfully decodes the original information and extracts semantic (meaningful) information upon their scenario, achieving high lexical and semantic similarities, as demonstrated in Table II. Such interoperability underlines the potential of refined pre-trained models (weights) that could be utilized as a new type of source compression technique, which can be discussed in our future work.

V. Conclusions

In this paper, we have shown that seq2seq-SC, which uses a pre-trained model, not only helps with computation time during training, but also with generalization of adapting to new texts unforeseen during training. This network proved its superiority over Deep-SC in terms of semantic similarity, through link-level simulations which closely resemble actual 5G systems. While we have focused on text datasets, such pre-trained model in the future can be expanded to other datasets including speeches, videos, etc., where benefits of semantic communication in terms of communication efficiency becomes even more substantial.

VI. REFERENCES

[1] Guangming Shi, Yong Xiao, Yingyu Li, and Xuemei Xie, “From semantic communication to semantic-aware networking: model, architecture, and open problems,” *IEEE Commun. Mag.*, vol. 59, no. 8, pp. 44–50, 2021.

[2] Huiqiang Xie, Zhijin Qin, Geoffrey Ye Li, and Biing-Hwang Juang, “Deep learning enabled semantic communication systems,” *IEEE Trans. on Signal Process.*, vol. 69, pp. 2663–2675, 2021.

[3] Jakob Hoydis, Sebastian Cammerer, Fayçal Ait Aoudia, Avinash Vem, Nikolaus Binder, Guillermo Marcus, and Alexander Keller, “Sionna: An open-source library for next-generation physical layer research,” *arXiv preprint arXiv:2203.11854*, 2022.

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.

[5] Han Hu, Xingwu Zhu, Fuhui Zhou, Wei Wu, Rose Qingyang Hu, and Hongbo Zhu, “One-to-many semantic communication systems: Design, implementation, performance evaluation,” *IEEE Commun. Lett.*, 2022.

[6] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer, “BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension,” in *Proc. the Annual Meeting of the Association for Computational Linguistics*, Online, July 2020, pp. 7871–7880.

[7] 3GPP TS 38.212 v17.3.0, “NR; multiplexing and channel coding,” Sep. 2022.

[8] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu, “Bleu: a method for automatic evaluation of machine translation,” in *Proc. the Annual Meeting of the Association for Computational Linguistics*, Philadelphia, Pennsylvania, USA, July 2002, pp. 311–318.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. Conf. the Association for Computational Linguistics: Human Language Technologies*, Minneapolis, Minnesota, June 2019, pp. 4171–4186.

[10] Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li, “On the sentence embeddings from pre-trained language models,” in *Proc. the 2020 Conf. on Empirical Methods in Natural Language Processing (EMNLP)*, Online, Nov. 2020, pp. 9119–9130.

[11] Nils Reimers and Iryna Gurevych, “Sentence-BERT: Sentence embeddings using Siamese BERT-networks,” in *Proc. the 2019 Conf. on Empirical Methods in Natural Language Processing and the 9th Int. Joint Conf. on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China, Nov. 2019, pp. 3982–3992.

[12] Philipp Koehn, “Europarl: A parallel corpus for statistical machine translation,” in *Proc. Machine Translation Summit X: Papers*, Phuket, Thailand, Sept. 13-15 2005, pp. 79–86.

[13] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning, “A large annotated corpus for learning natural language inference,” in *Proc. the Conf. on Empirical Methods in Natural Language Processing (EMNLP)*, 2015.

[14] Adina Williams, Nikita Nangia, and Samuel Bowman, “A broad-coverage challenge corpus for sentence understanding through inference,” in *Proc. the Conf. the Association for Computational Linguistics: Human Language Technologies*, 2018, pp. 1112–1122.

[15] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier, “From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions,” *Trans. of the Association for Computational Linguistics*, vol. 2, pp. 67–78, 2014.