How Much Effort is Needed to Transmit the Tense: Probing Sentence Transformer Representations for Linguistic Properties

Urban Knupleš
Institute for Natural Language Processing
University of Stuttgart
Stuttgart, Germany
urban.knuples@ims.uni-stuttgart.de

Abstract

Transformer models dominate the NLP field, excelling in various tasks, yet comprehending the information they capture remains a challenge. Probing tasks offer a method to unveil the type of information that is encoded in these models. The goal of this project is to study how well sentence representations encode information about linguistic properties. In line with prior research, the study aims to probe sentence transformers in comparison to a BiLSTM baseline across different simple linguistic tasks. We probe with a conventional probing classifier and a more informative MDL probe. We find that most linguistic information is encoded in the lower and middle layers of the sentence transformers that encode more information compared to the BiLSTM on 8 of 10 probing tasks. While that is noteworthy, our examination of the final representations of the sentence transformer models showes that they encode less linguistic information compared to the BiLSTM. Additionally, our findings indicate that the conventional probing classifier falls short of the MDL probe in scrutinizing the encoding of simple linguistic information. This observation prompts a reconsideration of probing tools used to analyze properties in sentence representations. Our analysis is *publicly* available and reproducible[1].

1 Introduction

Transformer-based models have become popular in the field of natural language processing (NLP), often surpassing other neural architectures in various downstream tasks (Lin et al., 2022). The currently most used paradigm involves utilizing large pre-trained language models (PLMs), such as BERT (Devlin et al., 2019), which are fine-tuned on a downstream task. In this approach, the learned representations are further trained to transfer to a new task or domain. However, the majority of PLMs used represent black-box models that are difficult to interpret in terms of the type of information they learn from the large amount of data they were trained on.

Due to the capabilities of such models, research has shown that the representations also capture information about linguistic properties themselves (Belinkov et al., 2017). A common approach to investigating what information is encoded is through the use of *probing methods* (Belinkov et al., 2017; Conneau et al., 2018; Hewitt and Liang, 2019; Hewitt and Manning, 2019; Fayyaz et al., 2021). One such method of probing the representations of the PLMs for specific properties is by constructing *probing classifiers* (Conneau et al., 2018). These are supervised models that are trained on encodings from black-box architectures as input and predict linguistic properties using classification tasks. When these tasks are specifically designed for probing, they are often referred to as *probing tasks*.

Most probing classifiers are simple linear classifiers or multi-layer perceptrons, which has raised a debate about whether these methods capture information in the representation or learn the tasks themselves (Hewitt and Liang, 2019; Belinkov, 2022). Hence, researchers have proposed different methods for probing that address these limitations. One such method is the *Minimum Description Length (MLD) probe* by Voita and Titov (2020), which takes an information-theoretic perspective on probing the representations of PLMs.

An increasingly used version of transformer-based PLMs are a group of models called sentence transformers (STs) (Reimers and Gurevych, 2019). Fine-tuned networks based on pre-trained language models such as BERT (Devlin et al., 2019) use siamese and triplet network architectures to produce semantically meaningful sentence representations during training. They have achieved state-of-the-art performance on downstream tasks such as semantic search (Santander-Cruz et al., 2022)

[1] https://github.tik.uni-stuttgart.de/st180633/probing-hierarchical-sentences

or been utilized as a tool for estimating party (dis)similarities from texts (Ceron et al., 2022). However, compared to other variants of the BERT model (Rogers et al., 2020), there is less focus on analyzing their representations. A notable contribution in that direction is the work by Nikolaev and Padó (2023). Given the performance of these models tasks that require a better encoding of similarities between sequences of texts, the question arises: what linguistic information is encoded in the representations of STs?

In this study, we first replicate the 10 probing tasks from Conneau et al. (2018) using the originally proposed probing classifier, and extend the probing tasks using the MDL probe from Voita and Titov (2020). We probe the sentence representations of a BiLSTM and two ST models for surface-level, syntactic, and subtle semantic information. The BiLSTM is included for reproducibility in our study. Our results show that the BiLSTM model outperforms the other models across almost all probing tasks, with comparable results to the original results reported by Conneau et al. (2018).

To further analyze the information captured in the representations, we reduce the dimensions of the sentence representations produced by our different PLMs using the dimensionality-reduction method PCA. This allows to investigate whether the probing methods learned to decode the information encoded in the representations or merely learned to capture spurious correlations. While the probing results on the dimensionally-reduced representations show lower probing scores, they still produce results with similar trends to those in the first experiment. Some results further corroborate our assumptions about the probing classifier capturing spurious correlations in one of the probing tasks. A notable insight is that the use of the MDL probe led to more comparable results, indicating that it is a more suitable probing method for analyzing sentence representations for linguistic properties.

Finally, we take a closer look at the change in representation of linguistic properties across the layers of the two ST models, to investigate whether a better encoding is present in other layers of the network. The results show that the ST models outperform the BiLSTM when taking the representations of the best layer for each task, indicating that more linguistic information is encoded in the deeper layer of the network. The experiments also reveal that for probing tasks where the sentences were modified, most information is captured in the upper-middle layers of the network, while for other surface-level, syntactic, and semantic tasks, the best results were obtained from the beginner layers of the network.

2 Related work

Probing is a method used to investigate what linguistic properties are encoded in sentence embeddings generated by fixed-length sentence encoders trained on different pre-training tasks. Conneau et al. (2018) defined probing tasks as simple classification tasks that focus on surface-level, syntactic, and semantic properties. The goal is to determine if a particular information is encoded in the sentence embeddings by measuring the success of a probing classifier using accuracy and similarity metrics. Their results suggest that models generally learn and encode a wide range of linguistic properties, such as the maximal depth of the parse tree of the sentence, and that different types of embeddings capture linguistic properties to different degrees. While the work investigated the encoding of linguistic properties in LSTM and CNN networks, it is yet unknown how well STs encode such linguistic properties.

Hewitt and Manning (2019) have conducted a related study that investigates the syntactic knowledge of LSTM and transformer networks using a structural probe. The probe was trained to extract syntax trees from the representations generated by both architectures. The authors found evidence that such knowledge is encoded and can be read in the generated representations of both models. However, the study only conducted experiments on word-level representations and not sentence-level representations, despite finding linguistic information modern deep neural networks based on LSTMs or transformer models.

In their work, Fayyaz et al. (2021) analyzed the representations of three transformer-based language models with different pretraining objectives: masked language modeling, permuted language modeling, and replaced token detection. These models were evaluated on various core NLP tasks such as Named Entity Recognition and Coreference Resolution. Instead of using traditional probing methods evaluated on accuracy, they employed the information-theoretic probing method proposed by Voita and Titov (2020), which has been found to be more reliable in extracting properties from the model's representations (Belinkov, 2022). The analysis revealed that variations in pretraining objectives and architectures result in different encoding of linguistic information in word representations. However, an analysis conducted with such a probing method has yet to be extended to transformer-based models trained on a sentence representation objective.

3 Methodology

To investigate the extent to which STs capture linguistic properties in sentence representations, we define several probing tasks. These tasks involve identifying properties by framing them as classification tasks. In this formulation, a classifier is trained on top of the model's sentence embeddings to predict a targeted word or property of a sentence.

For instance, consider the legality of word order within a sentence. In a binary classification setting, the classifier aims to predict whether a sentence is correct or incorrect due to the inversion of two words. If the trained classifier can accurately predict the legality of the word order in a sentence, it suggests that the pre-trained encoder has embedded information about word order that can be read by the trained classifier.

3.1 Probing Tasks

We utilize the probing tasks and datasets by Conneau et al. (2018) to investigate linguistic knowledge distributed across 10 linguistic tasks. The tasks with added examples are shown in Table 1. The tasks are divided into three categories: *surface-level tasks*, which account for the surface properties of the encoded sentence; *syntactic tasks*, which test the ability of sentence embeddings to encode syntactic properties; and subtle *semantic tasks*, which rely on syntactic structure and with an additional understanding of what a sentence denotes. The probing dataset consists of pre-processed sentences extracted from the Toronto Book Corpus (Paperno et al., 2016). Each sentence is between 5 and 28 words long and has been processed to include part-of-speech, constituency, and dependency parsing information. The dataset consists of task-specific subsets, each with 120k sentences split into training, validation, and test sets in a 80/10/10 ratio. Furthermore, the subsets are balanced for each of the target classes.

The following is a brief description of each probing task:

1. **Sentence length** (SentLen): The objective is to estimate the length of a sentence based on the number of words it contains. Sentences are categorized into six fixed-width bins according to their length, forming a six-class classification task.

2. **Word content** (WC): Evaluates whether the encoder embeds information about the original words in the sentence. The dataset comprises sentences that include just one out of 1,000 medium-frequency words chosen from the source corpus vocabulary. The goal is to predict which of the 1,000 words a sentence contains, formulating a challenging 1k-way classification task.

3. **Tree depth** (TreeDepth): Examines whether the representations encode the hierarchical structure of sentences and if the classifier can accurately predict the depth of a sentence's syntax tree. The sentences in the dataset have tree depth values from 5 to 12, creating an eight-class classification task.

4. **Top constituent** (TopConst): Sentences are classified based on the sequence of top constituents immediately below the sentence (S) node. The task is formulated as a 20-way classification problem, with 19 classes making out the most frequent sequences and one for all the other constructions.

5. **Bigram shift** (BiShift): For this task, half of the sentences are modified by reversing the order of randomly selected bigrams in the sentence. The challenge is to predict whether a sentence contains a reversal, making it a binary classification task.

6. **Tense prediction** (Tense): Evaluates if the classifier can determine the tense of the main verb, comparing past and present tenses. This forms a binary classification task.

7. **Subject number** (SubjNum): Examines whether the representations store information about the number of the subject in the main clause. The dataset only contains sentences with one subject, and the classifier is tasked with predicting whether the subject is singular or plural.

Group	Task	Example	Label
Surface-level	SentLen	*The riders halted and looked back at her .*	1
	WC	*The boys nodded happily .*	happily
Syntactic	TreeDepth	*It even seemed to them that Mary was smiling at them .*	10
	TopConst	*He stepped inside the museum and started poking around .*	NP VP
	BShift	*Then it her hit .*	Yes
Semantic	Tense	*Tears pricked my eyes .*	PAST
	SubjNum	*The crows circled above me .*	NNS
	ObjNum	*Peter set down his fork .*	NN
	SOMO	*Her eyes bore down into mine and I just nodded in theatre .*	Yes
	CoordInv	*He grabbed my ankle and I started to crawl away .*	Yes

Table 1: Datapoint samples for each of the probing tasks used in the experiments.

8. **Object number** (ObjNum): The task is similar to the SubjNum task, but it involves predicting the number of the direct object in the main clause. The classifier's goal is to identify whether the direct object is singular or plural.

9. **(Semantic) odd man out** (SOMO): In this task, half of the sentences are modified by replacing a random noun or verb o with another noun or verb r that has comparable bigram frequencies in the corpus. The goal of the classifier is to predict the plausibility of a sentence based on whether it has been modified or not, resulting in a binary classification task.

10. **Coordination inversion** (CoordInv): This task tests whether the encoder can identify if a sentence has an inverted order of its two coordinate clauses. The dataset only contains sentences with two top-level conjuncts. The task is a binary classification that requires identifying whether a sentence is intact or modified. As described by Conneau et al. (2018), it requires the encoder to understand broad discourse and pragmatic factors.

3.2 Probing Methods

To analyze and decode linguistic information from sentence representations encoded by STs, we use two probing methods: a probing classifier (Conneau et al., 2018) and Minimum Description Length (MLD) probing (Voita and Titov, 2020).

3.2.1 Probing Classifier

A probing method is a multilayer classifier, which is trained to predict a linguistic property from the input of a sentence representation. Following Conneau et al. (2018), our classifier consists of a single hidden layer using a sigmoid activation function that projects the input representation into a 256-dimensional vector. The classifier's input dimension is the same as the encoder's fixed-sized sentence representation, and the output dimension is equal to the number of classes in the trained task. The classifier is trained using a training set and continues to train until the loss on the validation set stops improving. Once trained, the classifier is evaluated using a held-out test set, and classification accuracy results are reported. The high accuracy scores achieved on the evaluation task demonstrate the classifier's ability to decode linguistic knowledge from the encoder's representations.

3.2.2 MDL Probing

The MDL probe is an information-theoretic approach that measures the performance of the probing model and evaluates the amount of effort needed to achieve the prediction of a property. The idea behind the approach is to transform the process of predicting a linguistic property from the encoder's sentence representations into training a probe that can effectively transmit the property from the said representations. By doing so, it becomes possible to estimate the minimum length required to transmit the property. Meaning the better the representations encode a property, the better the probing model compresses and transmits said property. This approach is more informative and robust compared to conventional probing classifiers.

We use the *online coding* method to estimate MDL[2]. This is done by dividing the dataset $D =$

[2] Another method proposed by Voita and Titov (2020) is *variational coding*, which computes complexity using a Bayesian model. However, we chose to use online coding because both methods exhibit similar trends in code length,

$(x_i, y_i)_{i=1}^{N}$ into timesteps $1 = t_0 < t_1 < ... < t_S < N$. Here, x_i refers to the sentence representation, y_i is the linguistic property that we want the model to transmit, S refers to the number of chunks of the dataset, and N is the total number of samples in the dataset. A probing model is trained on the samples $(1, ..., t_i)$ and used to predict the linguistic properties of the next samples $(ti + 1, ..., t_{i+1})$. This process continues until the entire dataset has been processed[3]. MDL is calculated as the sum of the cross-entropy loss of the classifier over the data (data codelength) for each timestep, and the uniform encoding of the first block.

$$\text{MDL} = L^{\text{online}}(y_{1:n}|x_{1:n}) = t_1 \log_2 K \\ - \sum_{i=1}^{S-1} \log_2 p_{\theta_i}(y_{t_i+1:t_{i+1}}|x_{t_i+1:t_{i+1}})$$

where C is the number of target classes. Following Fayyaz et al. (2021), we reformulate MDL as a compression evaluation metric. In this formulation, MDL is scaled in relation to the maximum possible codelength. We assume that each representation has a label with a probability of $1/K$ and is transmitted without training. Compression is calculated as:

$$\textit{compression} = \frac{N \log_2(C)}{\text{MDL}}$$

A model that performs well will report a higher compression score, since fewer bits in the codelength need to be transmitted by the model. In our experiments, we chose to report compression instead of codelength. This is because both accuracy and compression should be maximized, while minimum description length (MDL) should be minimized. Reporting compression allows for easier intuition and comparison.

3.3 Sentence Embedding Models

When analyzing the sentence representations of PLMs, we utilized two STs and a Bi-directional LSTM. All models are pre-trained and produce a fixed-sized vector encoding the representation of an input sentence. We chose the BiLSTM as a point of comparison with the results from Conneau et al. (2018), as we are reproducing the probing tasks from their work.

BiLSTM: For our baseline model, we use a pre-trained BiLSTM published by Conneau et al. (2017)[4] that at the time of publishing reported state-of-the-art sentence embeddings. The BiLSTM is a recurrent neural network that generates a sentence representation by concatenating the hidden representations from both a forward LSTM (that reads and encoded a sentence from left to right) and a backward LSTM (concurrently from right to left). The model has approximately 38 million parameters. When given a sentence as input, the model produces a 4096-dimensional representation vector by performing a max pooling step over the word embeddings that are produced at each step of the model. It was trained in a supervised approach on the Stanford Natural Language Inference (SNLI) dataset (Bowman et al., 2015) on fastText word embeddings (Mikolov et al., 2018), where model encodes two sentences and feeds them to a classifier that classifies them as either contradictory, neutral or entailed.

The model's architecture and training were subsequently reproduced by Conneau et al. (2018) in investigating the encoding of linguistic properties. We thus use this as a proxy for comparing their reported results.

SBERT: We use two STs models in our work: PARAPHRASE-MPNET-BASE-V2[5] (S-MPNet) and PARPAHRASE-MINILM-L12-V2[6] (S-MiniLM). Both models are fine-tuned from a base model, with S-MPNet being based on MPNET-BASE (Song et al., 2020) and S-MiniLM being based on MINILM-L12-H384-UNCASED (MiniLM) (Wang et al., 2020). Both are finetuned using a contrastive learning objective from Reimers and Gurevych (2019), where given a sentence pair, the model predicts which randomly sampled sentence from the dataset was actually paired with it. S-MiniLM is a distilled instance of a larger UNILM-V2 (Bao et al., 2020) model. In model distillation, the smaller model acts as a student that learns to mimic the encoding representations

and it is easier to implement.

[3] Following Voita and Titov (2020), we use timesteps that correspond to 0.1%, 0.2%, 0.4%, 0.8%, 1.6%, 3.2%, 6.25%, 12.5%, 25%, 50%, and 100% of the dataset for all our experiments.

[4] https://github.com/facebookresearch/InferSent

[5] https://huggingface.co/sentence-transformers/paraphrase-mpnet-base-v2

[6] https://huggingface.co/sentence-transformers/paraphrase-MiniLM-L12-v2

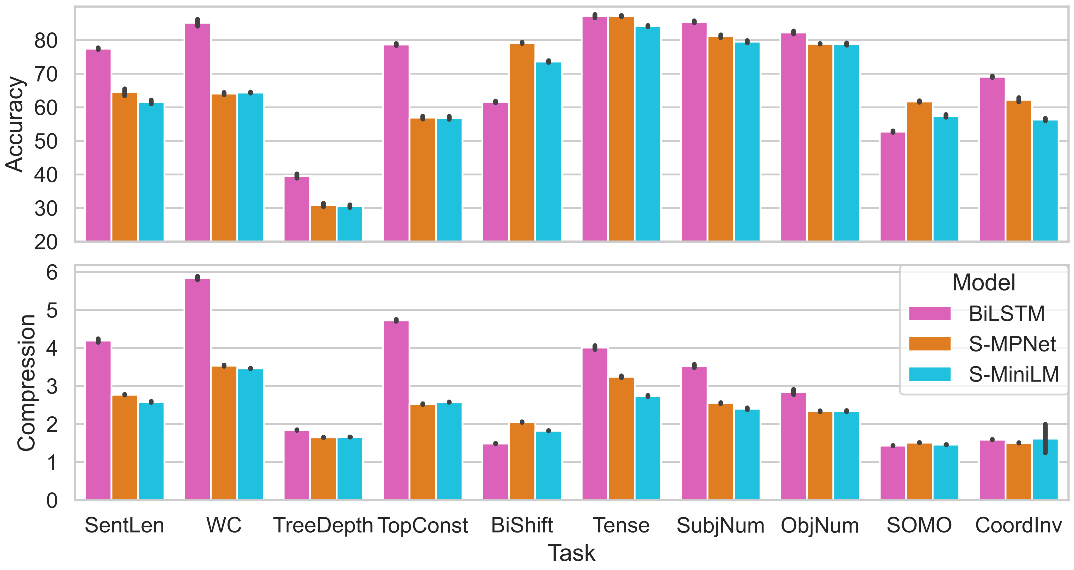

Figure 1: Comparison of probing classification (above) and MDL probing (below) for probing the encoding of linguistic properties in BiLSTM, S-MPNet, and S-MiniLM sentence representations.

produced by the larger teacher model when given the same input. This effectively transfers the knowledge previously learned by the teacher model to the smaller model (Wang et al., 2020). Both models were fine-tuned on the same paraphrasing datasets, including the SNLI dataset. Sentences with an entailment label were considered similar and trained to produce a smaller distance in the vector space.

S-MPNet contains approximately 110 million parameters and produces 768-dimensional sentence representations. In contrast, S-MiniLM is considerably smaller with approximately 34 million parameters and produces a 384-dimensional sentence representation. Both models take the mean of the final hidden layers of each input token in the sentence.

To enable meaningful comparisons between sentence representations from different architectures while using limited computational capacity and time constraint, our selection criteria required that the models be pre-trained or fine-tuned on comparable tasks. We also included S-MiniLM as a point of comparison with BiLSTM. Both models have a similar number of parameters.

4 Probing Pre-trained Sentence Representations

Approach. In order to effectively train and evaluate our probing methods, we extract sentence representations from each model using the 120k samples for each task defined in our probing tasks. This results in a total of 3.6 million sentence embeddings. For our MDL probing model, we use the probing classifier that was introduced in section 3.2.1. We train the classifiers on a batch size of 32 and a dropout rate of 0.3 for both probing methods, and we implement early stopping during training. Specifically, the classifier is trained on the training set until the cross-entropy loss of the development test stops improving after 10 iterations. To ensure accuracy and consistency in our results, we report the outcomes for each probing method on the average of 5 random seed runs. It is worth noting that, unlike Conneau et al. (2018), we do not perform additional hyperparameter tuning. This is because MDL has been shown to report stable results across settings and does not require any additional search for probing settings (Voita and Titov, 2020). Overall, our approach allows us to effectively train and evaluate the performance of our probing methods while maintaining consistency and accuracy in our results.

Results. The accuracy results of the probing classifier and compression results of the MDL probe are depicted in Figure 1. Table 2 lists the average accuracies for each task, including the performance reported by Conneau et al. (2018). Overall, the trends in both accuracy and compression scores are similar across tasks for all models.

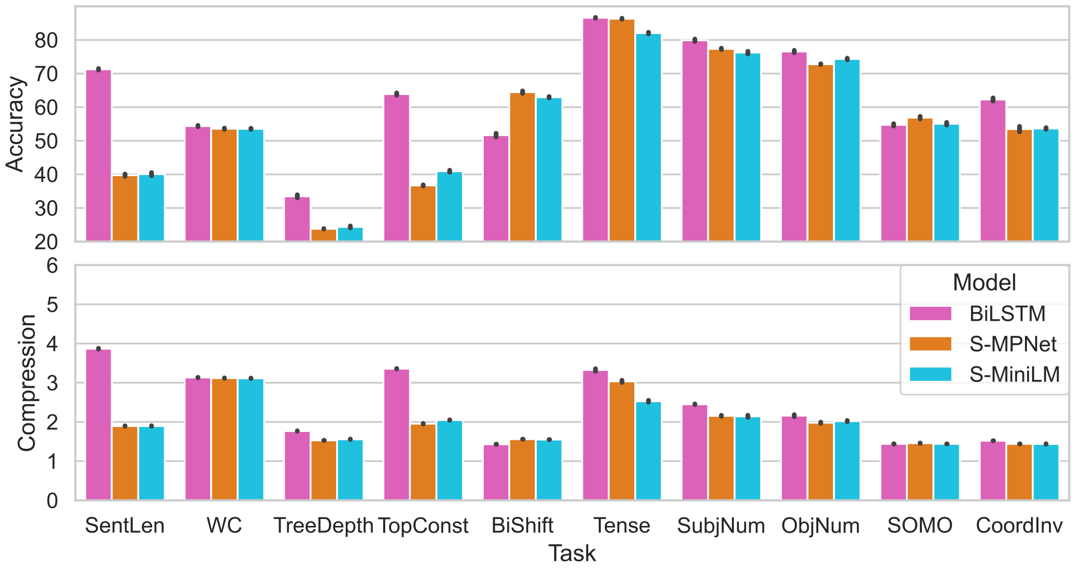

Figure 2: Comparison of probing classification (above) and MDL probing (below) for probing the encoding of linguistic properties in BiLSTM, S-MPNet, and S-MiniLM sentence representations that were reduced to a 100-dimensional space using PCA.

Task	Conneau et al.	BiLSTM	S-MPNet	S-MiniLM
SentLen	71.7	**77.5**	64.5	61.5
WC	**87.3**	85.2	64.1	64.4
TreeDepth	**41.6**	39.5	30.9	30.5
TopConst	70.5	**78.7**	56.9	56.9
BiShift	65.1	61.6	**79.2**	73.6
Tense	86.7	**87.1**	**87.1**	84.2
SubjNum	80.7	**85.5**	81.2	79.5
ObjNum	80.3	**82.3**	78.9	78.8
SOMO	**62.1**	52.7	61.7	57.4
CoordInv	66.8	**69.1**	62.2	56.3

Table 2: Results of the probing methods on the sentence representation models showing average accuracy. For comparison, we report the accuracy results obtained by Conneau et al. (2018) using a BiLSTM with max pooling.

The distilled S-MiniLM model and the larger, non-distilled S-MPNet model achieve similar scores on most tasks, but there are some instances where the former scores slightly lower than the latter. For instance, while the accuracy results for all models on the *Tense* prediction task are similar, a closer look reveals that the amount of compressed information obtained with the MDL probe differs between the models. The BiLSTM model outperforms the others in terms of compression scores, suggesting that it does a better job encoding tense information. A different result is observed in the semantic *SOMO* and *CoordInv* tasks, where the accuracy scores differ. S-MPNet reports the best results in the former, while BiLSTM performs better in the latter. When comparing compression scores, however, all three models encode a similar amount of information. In contrast to predicting tense, the MDL compression scores for the *TreeDepth* and *CoordInv* tasks are similar across all models, while the BiLSTM model reports better accuracy scores. Overall, both probing methods report a better encoding of surface-level, syntactic, and subtle semantic information in the BiLSTM model. The results raise the questions about the reliability and credibility of the probing methods and experiment reports, which we address in the next section.

It is worth noting that the classification probes trained on the embeddings by the model used by Conneau et al. (2018) achieve very similar performances to the probing classifier trained on our BiLSTM embeddings, which suggests that our results are reliable.

4.1 Sentence Representations of Equal Dimensions

Approach. When interpreting the results obtained from the probing experiments, it is important to consider that it is difficult to confirm whether the model stores the specific information that is being probed. The probing classifiers may rely on correlated information or find interactions between other features that help them solve the task, which would not give any insights about the information being sought. Therefore, it is crucial

Task	BiLSTM	S-MPNet	S-MiniLM
SentLen	**71.30**	39.70	40.10
WC	**54.40**	53.60	53.50
TreeDepth	**33.50**	23.80	24.40
TopConst	**63.90**	36.70	40.90
BiShift	51.70	**64.50**	62.90
Tense	**86.60**	86.30	82.00
SubjNum	**79.90**	77.30	76.20
ObjNum	**76.60**	72.80	74.30
SOMO	54.70	**56.90**	55.10
CoordInv	**62.30**	53.50	53.60

Table 3: Results of the average accuracy scores from the probing methods conducted on sentence representations that were reduced to a 100-dimensional space using PCA.

to use methods that can mitigate this risk and provide a better understanding of the information being encoded in the models. We conducted an additional experiment using principal component analysis (PCA) (Pearson, 1901). PCA is a widely used method that can reduce the dimensionality of embeddings while retaining most of the information. It uses singular value decomposition to identify the most important patterns in the data and project them onto a lower-dimensional space. By doing so, PCA can reduce the risk of the probing classifier relying on spurious correlations and provide a better understanding of the information encoded in the models. In our experiments, we reduce the dimensions of the embeddings to 100, using the scikit-learn PCA implementation (Pedregosa et al., 2011)[7]. We then rerun the same probing methods and tasks as in the previous set of experiments to evaluate the performance of the models on the reduced-dimensional embeddings. Comparing probing results on original and reduced embeddings can potentially help us understand how models encode information and how it is affected by dimensionality reduction. This may provide a more accurate understanding of the information encoded in the models and reduce the risk of probing classifiers relying on spurious correlations.

Results. Table 3 and Figure 2 show the scores achieved by both probing methods on the dimensionality-reduced sentence representations. Overall, the accuracy and compression scores decrease across all tasks for both SBERT and BiLSTM methods, which is expected. Notably, the compression scores for all methods, except *WC* on BiLSTM, remain similar to the previous experiment with the original sentence representations. This suggests that our models capture linguistic information in the sentence representations. However, the *WC* scores may be due to the classifiers relying on correlated information instead of decoding linguistic properties, since the BiLSTM representations were reduced from a dimensionality size of 4096 to 100. On the other hand, the accuracy scores for all tasks significantly drop, with some tasks such as *BiShift* dropping by almost 10%. This highlights the usefulness of employing MDL probes to draw conclusions on properties encoded in sentence representations, and suggests that methods that rely on accuracy may yield inconsistent outcomes. Therefore, the results and conclusions drawn from such probes, as presented by Conneau et al. (2018), should be reconsidered.

5 Layer-wise Analysis of Sentence Transformers

Approach. Building on the results of the previous experiments, we investigate whether there is a better encoding of linguistic properties in other layers of the S-MPNet and S-MiniLM models. To produce sentence representations, we follow the same pooling strategy used by both models for the last hidden layer. However, we now produce sentence representations from *each* layer $n \in [0; 12]$, since both models contain the same number of layers. Layer 0 represents the initial embedding layer of the models.

According to insights from Voita and Titov (2020), the use of accuracy as a metric can result in inconsistent layer rankings, complicating quality assessments. We also observed similar inconsistencies in our probing experiments on the dimensionality-reduced representations. Therefore, in our probing method, we rely solely on the compression results from the MDL probe. This strategy helps us avoid potential misinterpretations that may occur when comparing layers based on accuracy scores.

Results. Figure 3 shows the results of information compression across layers and tasks. Higher compression scores indicate better task encoding. Across most layers, S-MPNet attains the highest compression scores on most tasks,

[7] https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html

with the highest scores concentrated in the deeper layers. In comparison, S-MiniLM performs better overall on the surface-level *SentLen* task. For tasks where sentences are modified, namely *BiShift*, *SOMO*, and *CoordInv*, the most information is captured at the 5th to 10th layer of the networks. For all other 7 tasks, most linguistic knowledge is encoded in the first 5 layers of the network, with the embedding layer encoding most information about *WC*. Interestingly, a shared behavior among most tasks and both the S-MPNet and S-MiniLM models is the decrease, and at points, encoding the least amount of linguistic information towards the final layer. This could be attributed to their contrastive learning objective.

These experiments also show a different aspect in terms of the results from the previous probing experiments. When choosing the highest compression score in comparison to the probing results of the BiLSTM model, the S-MPNet model achieves the overall best results on *WC*, *BiShift*, *ObjNum*, and *SOMO*, while the S-MiniLM model performs best on *SentLen*, *TreeDepth*, *TopConst*, and *CoordInv* tasks. These results provide evidence that there is an increase in linguistic information captured in the deeper layers of both networks.

6 Discussion and Conclusion

This study aims to investigate and improve understanding of sentence transformers, specifically by analyzing the degree of linguistic knowledge encoded in the sentence representations they produce. We use a suite of probing tasks proposed by Conneau et al. (2018) to probe for surface-level, syntactic, and subtle semantic information in the representations. To probe the representations, we use two probing methods: a conventional probing classifier, and an MDL probing method, which has recently proven to provide more reliable and informative results when compared with conventional probes (Belinkov, 2022). To the best of our knowledge, this is the first time this suite of probing tasks and methods has been used to analyze such state-of-the-art pre-trained sentence transformer models.

We replicated the study by Conneau et al. (2018) using a BiLSTM pre-trained for NLI tasks and two sentence transformers specifically trained for NLI tasks. One is a standard pre-trained sentence transformer and the other a smaller distilled transformer. We discover that most linguistic information is encoded in the lower and middle layers of the sen-

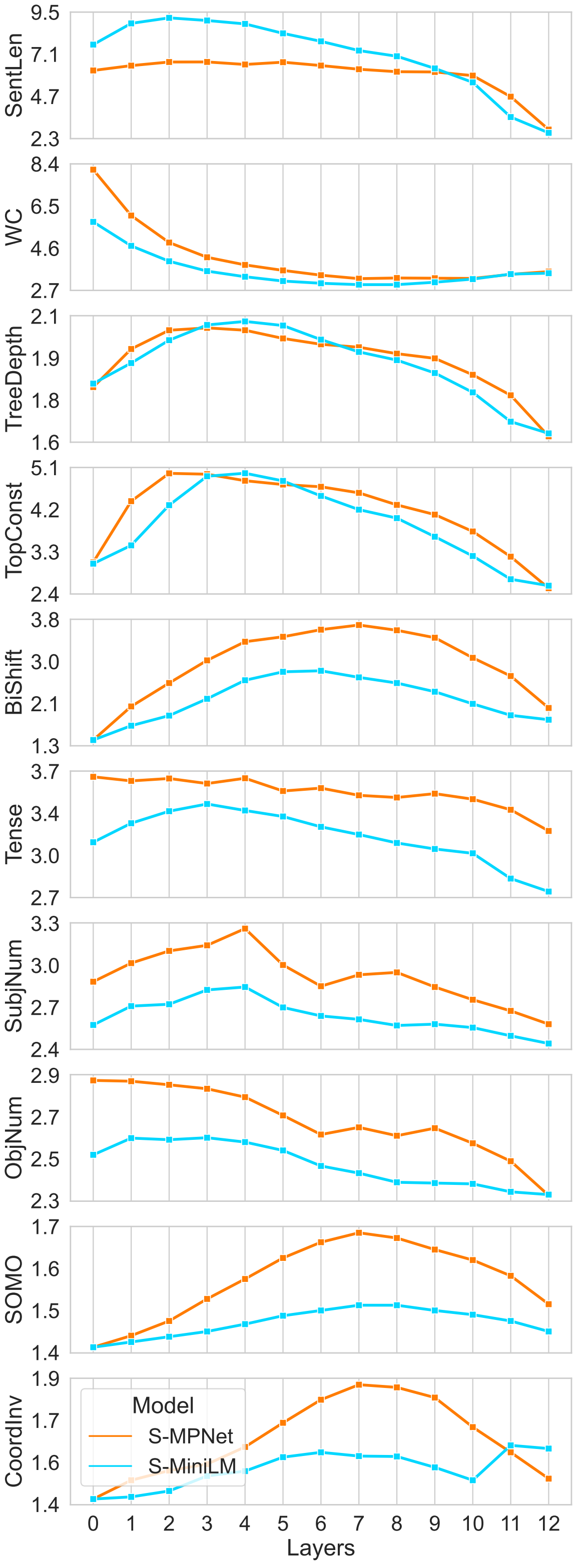

Figure 3: MDL probing compression of S-MPNet and S-MiniLM across layers.

tence transformers, which outperform the BiLSTM on 8 of 10 probing tasks. However, we find that the final sentence representations produced by sentence transformers, typically used for downstream tasks, encode less linguistic information than the BiLSTM.

Interestingly, we observe that the smaller distilled sentence transformer achieve similar results to the larger, standard sentence transformer, showing that distillation captures a similar amount of information compared to a larger, non-distilled sentence transformer. We achieved results similar to those of Conneau et al. (2018) using our BiLSTM. However, during the process, we noticed some interesting patterns. When we reduced the dimensions of the representations to comparable sizes using PCA, the conventional probing classifier showed a tendency to learn spurious correlations from the representations. Despite this, the MDL probe reported similar results to our initial findings. These findings showcase the usefulness of the MDL probe in drawing conclusions on properties encoded in the representations, while conclusions from a conventional probing classifier should be reconsidered.

A direction for future work would be to explore changes in the amount of encoded linguistic information on a larger variety of sentence transformers, while also comparing the results to the vanilla transformers from which the sentence transformers were fine-tuned. This would show how much the contrastive learning objective of the sentence transformers impacts the encoding of such information. Considering the limited number of probing methods used in the study, future work could compare the usefulness of other probing methods (Belinkov, 2022) on simple and easily interpretable probing tasks.

7 Limitations

Our analysis is subject to several limitations that need to be taken into account:

1. The study analyzes sentence embeddings from only three models: a standard sentence transformer, a distilled sentence transformer, and a BiLSTM. Comparing different combinations of model architectures could lead to different outcomes in our analysis. Therefore, our results may not be applicable to other sentence transformer models or vanilla transformers. Furthermore, all of our pre-trained models were trained on a natural language inference objective. Although we controlled for similarity in domain, models with different objectives could yield different results.

2. There is a significant difference in the size of sentence representation vectors produced by the two models. The BiLSTM generates 4096-dimensional vectors, while the sentence transformer model S-MPNet generates 768-dimensional vectors and the S-MiniLM model only generates 384-dimensional vectors. During the dimensionality reduction of the sentence representations using PCA, a change in the results of the probing classifications was observed, with the accuracies of the BiLSTM being significantly lower compared to the sentence transformer model. Although we attempted to control the outcome of our analysis by using the more robust MDL probing method, which reports compression scores as an evaluation metric, a comparison of the two architectures with similarly sized sentence representations could yield different results in the analysis.

8 Ethics Statement

In our study, we have strictly relied on publicly available datasets and models. The datasets neither contain any identifiable personal information nor any sensitive data that could potentially infringe on an individual's privacy. In our analysis we did not identify any immediate ethical concerns in our analysis.

References

Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, Nan Yang, Xiaodong Liu, Yu Wang, Songhao Piao, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2020. Unilmv2: Pseudo-masked language models for unified language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219.

Yonatan Belinkov, Lluís Màrquez, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2017. Evaluating layers of representation in neural machine translation on part-of-speech and semantic tagging tasks. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume*

1: Long Papers), pages 1–10, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Tanise Ceron, Nico Blokker, and Sebastian Padó. 2022. Optimizing text representations to capture (dis)similarity between political parties. In *Proceedings of the 26th Conference on Computational Natural Language Learning (CoNLL)*, pages 325–338, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

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

Alexis Conneau, Germán Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single vector: Probing sentence embeddings for linguistic properties. *CoRR*, abs/1805.01070.

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.

Mohsen Fayyaz, Ehsan Aghazadeh, Ali Modarressi, Hosein Mohebbi, and Mohammad Taher Pilehvar. 2021. Not all models localize linguistic knowledge in the same place: A layer-wise probing on BERToids' representations. In *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 375–388, Punta Cana, Dominican Republic. Association for Computational Linguistics.

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. 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 2733–2743, Hong Kong, China. Association for Computational Linguistics.

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.

Tianyang Lin, Yuxin Wang, Xiangyang Liu, and Xipeng Qiu. 2022. A survey of transformers. *AI Open*, 3:111–132.

Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2018. Advances in pre-training distributed word representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Dmitry Nikolaev and Sebastian Padó. 2023. Representation biases in sentence transformers. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3701–3716, Dubrovnik, Croatia. Association for Computational Linguistics.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1525–1534, Berlin, Germany. Association for Computational Linguistics.

Karl Pearson. 1901. Liii. on lines and planes of closest fit to systems of points in space. *Philosophical Magazine Series 1*, 2:559–572.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Yamanki Santander-Cruz, Sebastián Salazar-Colores, Wilfrido Jacobo Paredes-García, Humberto Guendulain-Arenas, and Saúl Tovar-Arriaga. 2022. Semantic feature extraction using sbert for dementia detection. *Brain Sciences*, 12(2):270.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA. Curran Associates Inc.

Elena Voita and Ivan Titov. 2020. Information-theoretic probing with minimum description length. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 183–196, Online. Association for Computational Linguistics.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in Neural Information Processing Systems*, 33:5776–5788.