Speaking Rationally by Gestures: Information Theoretic Evidence from Multi-Modal Language Models

Anonymous ACL submission

Abstract

The multi-modality nature of human communication can be utilized to enhance the performance of computational language models. However, few studies have explored the non-verbal channels with finer theoretical lens. We use multi-modal language models trained against monologue video data to study how the non-verbal expression contributes to communication, by examining two aspects: first, whether incorporating gesture representations can improve the language model's performance (perplexity), and second, whether the gesture channel demonstrates the similar pattern of entropy rate constancy (ERC) found in verbal language, which is governed by Information Theory. We have positive results to support both assumptions. The conclusion is that speakers indeed use simple gestures to convey information that enhances verbal communication, and how this information is organized is a rational process.

1 Introduction

Communication is a multi-modal process, in which information from verbal and non-verbal modalities are mixed into one channel. It has already been revealed in empirical studies that speakers' expression in visual modality, including gestures, body poses, eye contacts and other types of non-verbal behaviors, play critical roles in face-to-face communication, as they add subtle information that is hard to convey in verbal language. However, it remains an untested idea to view these sparse and random non-verbal signals as a formal communication channel that transmits "serious" information, which has seldom been validated by computational studies. A key missing step is to explore whether the non-verbal information can be quantified.

The questions that are worth further investigation include (but are not limited to): How rich is the information contained in these non-verbal channels? What are their relationships to verbal information? Can we understand the meanings of different gestures, poses, and motions embedded in spontaneous language in a similar way to understanding word meanings? The goal of this study is to propose a simple but straight-forward framework to approach the above questions, under the guidance of Information Theory. Some preliminary, yet prospective results are presented.

2 Related Work

2.1 Gestures as non-verbal communication

There is vast literature on the connection between gesture and language in human communication. Gestures, defined as "the spontaneous hand movements produced in rhythm with speech" (Clough and Duff, 2020) naturally co-occur with spoken language. According to the thorough survey from (Clough and Duff, 2020), the *communication* function of gestures is one of the main focus of early studies. McNeill (1992) has classified gestures into two categories, representative and non-representative, in which the former has clearer semantic meanings (e.g., depicting objects and describing locations), while the latter refers to the brief, repetitive movements that has little substantive meanings.

2.2 Mixed-modal models in NLP and ML

The recent advances of deep neural network-based machine learning techniques provide new methods to understand the non-verbal components of human communication. Many existing works primarily focus on using multi-modal features as clues for a variety of inference tasks, including video content understanding and summarization (Li et al., 2020; Bertasius et al., 2021), as well as more specific ones such as predicting the shared attention among speakers (Fan et al., 2018) and semantic-aware action segmentation (Gavrilyuk et al., 2018; Xu et al., 2019). More recently, models that include multiple channels have been developed to character-

ize context-situated human interactions (Fan et al., 2021). Advances in representation learning have enabled researchers to study theoretical questions with the tools of multi-modal language models.

Neural sequential models are used for predicting the shared attention among speakers (Fan et al., 2018) and semantic-aware action segmentation (Gavrilyuk et al., 2018; Xu et al., 2019). More recently, models that include multiple channels have been developed to characterize visually embedded and context-situated language use (Fan et al., 2021; Li et al., 2019, 2021; He et al., 2022). Another line of work focuses on the predicting task in the opposite direction, that is, predicting/generating gesture motion from audio and language data (Ginosar et al., 2019; Yoon et al., 2020; Alexanderson et al., 2020). For short, advances in representation learning have enabled researchers to study theoretical questions complex models.

2.3 Insights from cognitive science studies

Gesture-based non-verbal communication has been proven to facilitate the formation of messages in cognitive science studies. This facilitation can come from multiple layers of visual and vocal signals can add semantic and pragmatic information in face-to-face communication. (Holler and Levinson, 2019). Visible gestures are more powerful form of communication than vocalization in dialogue object description tasks (Macuch Silva et al., 2020). In these studies, gestures from human subjects are usually encoded by the hands' spacial locations, which provide insights to the gesture extraction method used in this study. Also, their results strongly indicate the potentials of building more comprehensive computational language models by including simple non-verbal features. However, so far, few computational studies have attempted to directly model spontaneous language.

2.4 Information theories

Information theory (Shannon, 1948) has been broadly applied in computational linguistics as the theoretic background for the probabilistic models of language. This also provides philosophical explanations to a broad spectrum of linguistic phenomena. One example that interests researchers the most is the assumption/principle of *entropy rate constancy* (ERC). Under this assumption, human communication in any form (written, spoken, etc.) should optimize the rate of information transmission rate by keeping the overall entropy rate constant.

In natural language, *entropy* refers to the predictability of words (tokens, syllables) estimated with probabilistic language models. Genzel and Charniak (2002, 2003) first formulated a method to examine ERC for written language, by decomposing the entropy term into *local* and *global* entropy:

$$H(s|context) = H(s|L) - I(s, C|L) \quad (1)$$

in which s can be any symbol whose probability can be estimated, such as a word, punctuation, or sentence. C and L refer to the global and local contexts for s, among which C is purely conceptual and only L can be operationally defined. By ERC, the left term in eq. (1) should remain an invariant against the position of s. It results in an expectation that the first term on the right $H(s|L)$ should *increase* with the position of s, because the second term $I(s, C|L)$, i.e., the mutual information between s and itself global context should always decrease (see Genzel and Charniak (2003)'s paper for more examples). While they have confirmed the increase of local entropy in written language, Xu and Reitter (2016, 2018) also confirmed the pattern in spoken language, relating it to the success of task-oriented dialogues (Xu and Reitter, 2017).

Now, the goal of this study is to extend the application scope of ERC to the non-verbal realm. More specifically, if the s in eq. (1) represents any symbol that carries information, for example, a gesture or pose, then the same *increase* pattern should be observed within a sequence of gestures. ERC can be interpreted as a "rational" strategy for the information sender (speaker) because it requires less predictable content (higher local entropy) to occur at a later position within the message, which maximizes the likelihood for the receiver (listener) to successfully decode information with the least effort. The question explored here is whether we "speak" rationally by gestures.

3 Questions and Hypotheses

We examine two hypotheses in this study:
Hypothesis 1: Incorporating non-verbal representations as input will improve the performance of language modeling task. To test Hypothesis 1, we extract non-verbal representations using the output from pose estimation, and then compose discrete tokens to represent the non-verbal information. The non-verbal tokens are inserted into word sequences

and form a hybrid type input data for training language models. The language models are modified to take non-verbal and verbal input sequences simultaneously and compute a fused internal representation. We expect the inclusion of non-verbal information will increase the performance of language models measured by perplexity.

Hypothesis 2: Non-verbal communication conforms to the principle of Entropy Rate Constancy. To test Hypothesis 2, we approximate the local entropy ($H(s|L)$) of non-verbal "tokens" using the perplexity scores obtained from neural sequential models, and correlate it with the utterances' relative positions within the monologue data. If we can find that $H(s|L)$ increases with utterance position, is similar to verbal language, then it supports the hypothesis.

4 Methods

4.1 Data collection and processing

The video data used are collected from 4 YouTube channels, i.e., 4 distinct speakers. There are 1 female and 3 male speakers, and the spoken language is English. All the videos are carefully selected based on the standards that each video must contain only one speaker who faces in front of the camera, and whose hands must be visible. The automatic generated captions in `.vtt` format are obtained for each video.

The pre-processing step is to extract the full-body landmark points of the speaker, in preparation for the next gesture representation step. For this task, we use BlazePose (Bazarevsky et al., 2020), which is a lightweight convolutional neural network-based pose estimation model provided in `MediaPipe`[1]. It outputs 33 pose landmarks of the human body detected in each frame.

4.2 Extract discrete gesture labels

The next step is to represent gestures so that they can be embedded into language data. There are various ways of creating *continuous* representations for gestures/poses, such as the pose embedding technique (Mori et al., 2015). However, it is difficult to obtain a set of gestures that are *universal* across speakers using such continuous representations. Thus, for the exploratory purpose of this study, we start with a simple method to create *discrete* gesture labels, by categorizing the hands positions into grids.

[1] https://google.github.io/mediapipe/

We divide the front space of a speaker into 3×3 areas, thus, for each frame we have 9 rectangular areas of the same size, indicated by integer numbers from 1 to 9. Each hand is assigned an integer based on which region it falls into. Because the speaker's body appear at different positions from frame to frame, we develop an ad-hoc algorithm to annotate gesture labels, based on the estimated central axis of body and shoulder width. The pseudo code is presented in appendix A.1.

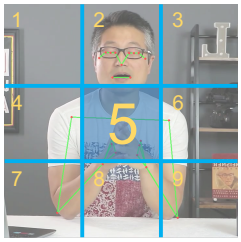
Next, we use the combination of both hands to create a unique gesture label for that frame. For example, as shown in fig. 1b, the speaker's left and right hands fall into region 9 and 8, so the gesture label is `<72>`. Because there are 9 possible positions for each hand, the total number of gesture labels is $9 \times 9 = 81$. For convenience, we use one integer ID (instead of the merged ID connected by a hyphen) to denote each of these 81 gestures: `<1>`, `<2>`, ..., `<81>`. Note that 81 is the maximum number, and the actual count of unique gesture labels depends on the data.

4.3 Prepare gesture sequences

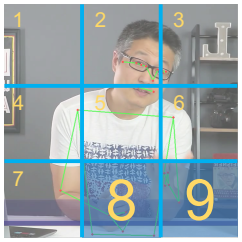
After obtaining the discrete gesture labels for all video frames, we prepare the gesture sequences based on the time stampped text transcript for each video. We use the automatically generated text transcript in `.vtt` format, which contains the `<START>` and `<END>` time stamps for each word (token) in the subtitle. See the following example:

```
<00:00:00.510><c> let's</c>
<00:00:00.780><c> talk</c>
<00:00:01.020><c> about</c>
<00:00:01.140><c> saving</c>
<00:00:01.650><c> some</c>
<00:00:01.860><c> time</c>
```

in which each word is annotated by a pair of `<c></c>` tag, and the `<START>` time stamp is appended to the head. We treat the start time for one word as the ending time for the previous word. In this example, the token *let's* elapses from 0.780 to 1.020 in seconds. Multiplying the time stamps with frame rate of 24 FPS (different videos have slightly different FPS), it tells that the word elapses from the 19th frame to the 24th. Then, for each frame within the range of $[19, 24]$, we extract a gesture label using the method described in Section 4.2, resulting in a sequence of gesture labels, $[g_{19}, g_{20}, \ldots, g_{24}]$. This sequence represents a con-



(a) Both hands in region 5 → label <25>.



(b) Right hand in region 9, left in 8 → label <72>.

Figure 1: Create discrete gesture labels based on landmark positions of both hands.

tinuous change of gestures during the articulation of the corresponding word, which in most cases, consists of identical gesture labels. Therefore, we select the median label g^m as a more compact representation.

For an utterance consisting of N word tokens, $\{w_1, w_2, \ldots, w_N\}$, we obtain the median gesture label for each token, $\{g_1, g_2, \ldots, g_N\}$. Despite the down sampling effect of using the median label, there is still large amount of repetition in the resulted g^m sequence, which brings a sparsity issue for later sequential modeling tasks. For example, in the first row of table 1, the median gesture label is the same <24> for the first 6 tokens, which means that the speaker did not move his/her hands during that period of time. It makes sense that we treat these repeated gesture labels just as one label. By merging the 6 repeats of <24> and 2 repeats of <36>, we get a compressed gesture sequence, {<36>, <24>}, which means the speaker has made two distinct gestures during the utterance. For each median gesture sequence of length N, we obtain its compressed version $\{\hat{g}_1, \hat{g}_2, \ldots, \hat{g}_{N'}\}$, where $N' \leq N$. See table 1 for more examples.

4.4 Language models that incorporate gesture inputs

We implement two neural network-based models for the language modeling tasks, using LSTM (Hochreiter and Schmidhuber, 1997) and Transformer (Vaswani et al., 2017) encoders. The models are tailored for handling two types of input: single-modal (words or gestures alone) and mixed-modal (words + gestures).

Single-modal LM task

The single-modal model takes as input a sequence of either word (w) or gesture (median g or compressed $\hat{g}$) tokens and convert them to the embedding space. Then the token embeddings are fed to the LSTM/Transformer encoders to compute a dense representation for tokens at each time step of the sequence. Finally, the dense representation at the current time step t is used to predict the token at the next time step $t+1$ using a softmax output. The model architecture is shown in fig. 2.

The learning object here is the same as a typical sequential language modeling task, i.e., to minimize the negative log probability:

$$NLL = -\sum_{k=1}^{K} \log P(t_k | t_1, t_2, \ldots, t_{k-1}) \quad (2)$$

in which $t_1, \ldots, t_{k-1}$ is all the tokens (gesture or word) before t_k within the same utterance. An exponential conversion of eq. (2) leads to the local entropy term, $H(g|L) = \exp(NLL)$, which is the target variable of our interest. Detailed model hyper-parameters and training procedures are included in appendix A.2.

Mixed-modal LM task

The mixed-modal model takes the word sequence $S_w(u) = \{w_i\}$ and gesture sequence $S_g(u) = \{g_i\}$ of the same utterance u simultaneously as input. A pair of sequences, S_w (words) and S_g (gestures) are the input, which is then fed into a modality fusion module, where the embedding representation for words and gestures at each time step, i.e., w_i and g_i, are fused by *sum*, *concat*, or a *bilinear* fusion component. Finally, the resulting mixed embeddings are encoded by the LSTM/Transformer encoder for the next-word prediction task. The purpose of this model is to verify Hypothesis 1, for which we expect the perplexity scores of a mixed-modal model to be lower than that of a single-modal one. It is also our interest to explore the optimal modality fusion method. The model's architecture is shown in fig. 2b. Detailed hyper-parameters will be presented in the Appendix.

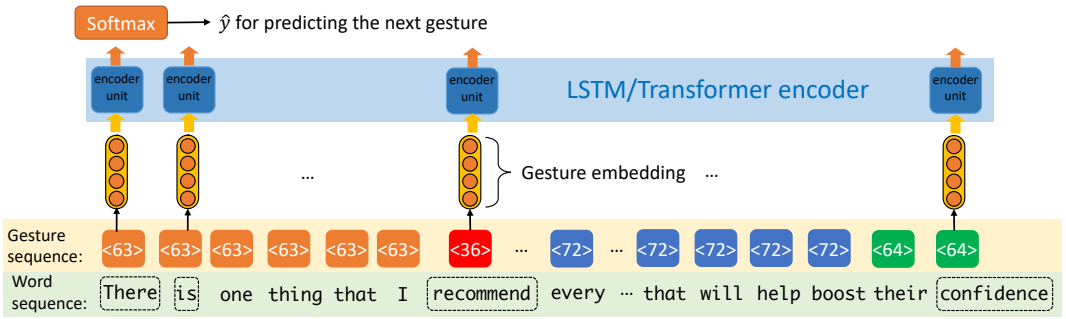
5 Results

5.1 Summary of data

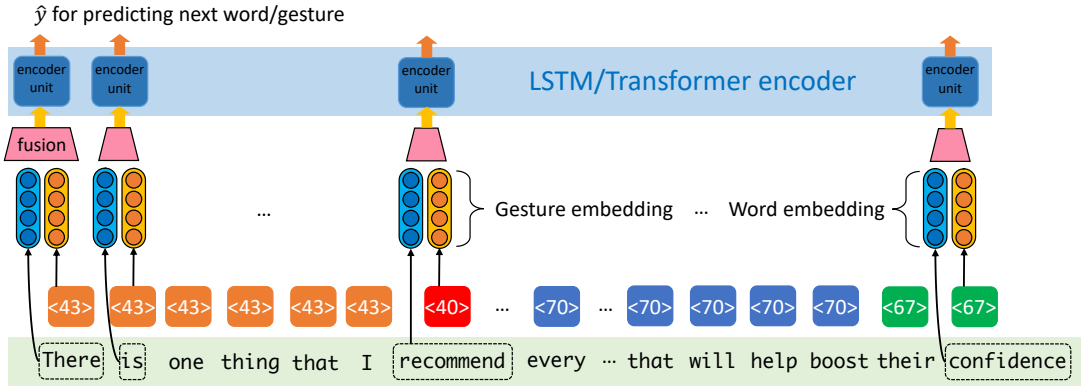
53 videos of total length 10 hours and 39 minutes are collected. The average length of each video is 723.7 seconds (*SD* = 438.1). 17.9K lines of automatically generated subtitles consisting of 121.5K word tokens are obtained. We have extracted 81 distinct gesture labels, and the total count of the

Word tokens in utterance	Median gesture of each token $\{g\}$	Compressed gesture sequence $\{\hat{g}\}$
going to give you a flatter look glossy	<24> <24> <24> <24> <24> <24> <36> <36> ($N = 8$)	<24> <36> ($N' = 2$)
now this is really your preference	<40> <72> <64> <64> <40> <40> ($N = 6$)	<40> <72> <64> <40> ($N' = 4$)
I think most of us can get on board	<63> <63> <63> <63> <63> <63> <63> <63> <63> ($N = 9$)	<63> ($N' = 1$)

Table 1: Examples of gesture sequences. Integers wrapped by "<>" are gesture labels.



(a) Single-modal (gesture) model



(b) Mixed-modal (word + gesture) model

Figure 2: Architecture of the LSTM/Transformer-based language models for handling single- (a) and mixed-modal (b) input sequences.

median gesture label is the same as that of the word tokens (121.5K). The compressed gesture labels has a smaller total count of 26.12 K.

The top 5 most frequent gesture labels are <63>, <56>, <64>, <72> and <36>, whose rankings are slightly different between the median and compressed labels. We find that the frequency distribution of gesture labels roughly follows the Zipf's law, as shown in the frequency vs. rank plots in fig. 3, which is a common distribution pattern in natural language data (Zipf, 2013; Piantadosi, 2014).

Gesture label <63> is the dominant gesture throughout the data. It is gestural position where the speaker's right hand (from his/her perspective) is in region 7, and left hand region 9. A detailed analysis The positional and semantic meanings of these labels is provided in section 5.4.

5.2 Examining Hypothesis 1: Mixed vs. single modal comparison

The plots of validation loss against training epochs are shown in fig. 4. We use the prefixes ***s-*** and ***m-*** to indicate the **single**-modal and **mixed**-modal models, respectively, that is, s- models take pure word sequences as input, while m- models take word+gesture sequences as input. It can be clearly seen that the *m*-LSTM has lower validation loss than *s*-LSTM, and same trend is found between *m*-Transformer and *s*-Transformer. It supports *Hypothesis 1*: gestures indeed contain useful information that can improve the language model's performance. The Transformer-based models have overall lower perplexity than LSTM-based ones, which is expected as a Transformer encoder has

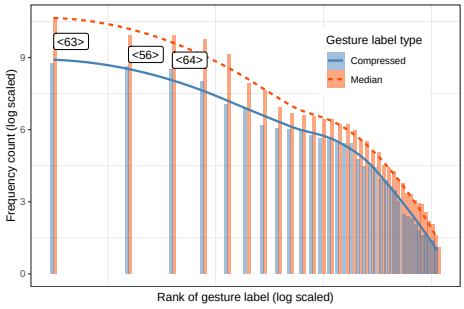


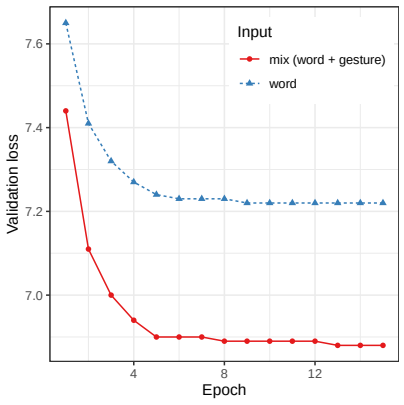
Figure 3: Frequency count against the rank gesture labels in logarithm transformed scales. Top three most frequent gesture labels annotated.

more parameters to facilitate the sequence prediction task. But meanwhile, the validation loss for training Transformer models does not decrease as significantly (see the less smooth curves in fig. 4b) as LSTM models, which probably indicates some overfitting issue. This can be fixed by collecting more training data.

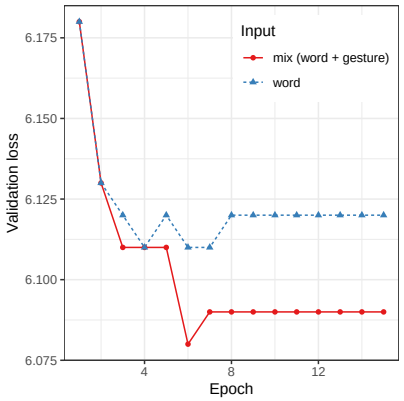
We also compare three different feature fusion method in training the m-LSTM/Transformer models. The corresponding validation losses are shown in fig. 5. It is found that *sum* and *concat* result in significantly lower loss for *m*-LSTM, but the difference is not that observable in *m*-Transformer, because in the latter loss shortly converges after training starts. Thus, we can conclude that *sum* and *concat* have similar performance in language modeling tasks. We will further verify this principle on more data (especially on Transformer model) in future studies.

5.3 Examine Hypothesis 2: Local entropy increases with utterance position

To examine *Hypothesis 2*, we plot the local entropy of each gesture sequence (median and compressed, respectively) against the corresponding utterance's position in fig. 6, which shows a visible increasing trend. We also use linear models to verify the correlations between local entropy and utterance position, that is, local entropy as dependent variable and utterance position as predictor (no random effect is considered due to limited data size). It is confirmed that utterance position is a significant predictor of local entropy with positive β coefficients. For raw gestures, the *beta*s are smaller: $\beta_{\text{LSTM}} = 1.6 \times 10^{-3}$ ($p < .05$), $\beta_{\text{Trm}} = 2.3 \times 10^{-3}$ ($p < .01$); for compressed gestures: $\beta_{\text{LSTM}} = 0.097$, $\beta_{\text{Trm}} = 0.093$ ($p < .001$). Therefore, the increase of local entropy is statistically significant. It supports our hypothesis.



(a) LSTM model



(b) Transformer model

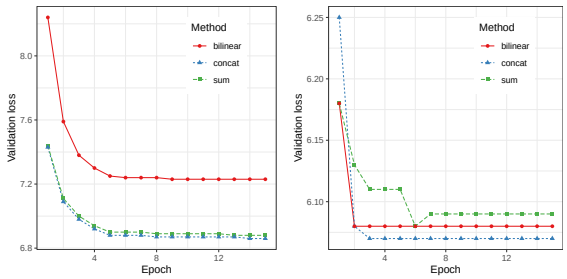
Figure 4: Validation loss against training epochs for comparing the mixed-modal and single-modal language models.

5.4 Analysis of typical gesture

We examine the top four frequent gesture labels <63>, <56>, <72> and <36>, and show some selected screenshots in fig. 7 (More examples for each gesture are included in appendix A.3).

For <63>, <56> and <72>, the positions of both hands are at the mid-lower position in front of the body. Gesture <63> has two hands evenly distant from the center, while <56> captures a movement to the right and <72> to the left. Gesture <36> has the right hand at the same height as the speaker's neck, and the left hand hanging down, which is a typical one-hand gesture in conversation. A technical detail is that in most screenshots of <36> the left hands are invisible, but the pose estimation algorithm can still infer their positions with accuracies above 95% (see the report from Mediapipe[2]), which is also why they are included

[2] https://google.github.io/mediapipe/



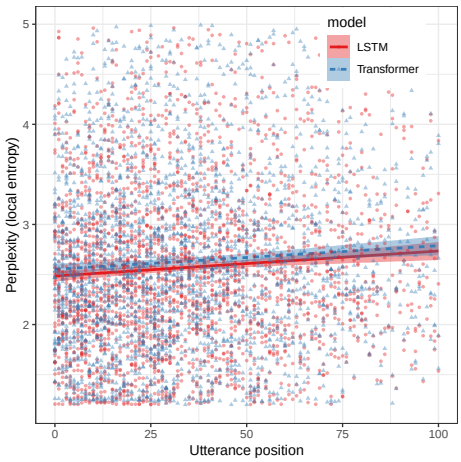
(a) *m*-LSTM model (b) *m*-Transformer model

Figure 5: Validation loss against training epochs for comparing the three feature fusion methods in mixed-modal models: sum, concat, and bilinear.

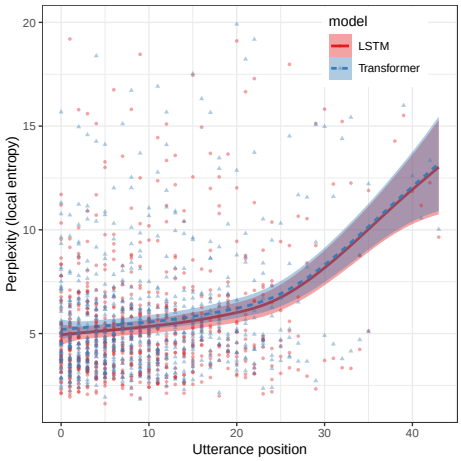
in our analysis. In general, the selected four gestures can represent commonly seen patterns in daily communication.

Based on the results from section 5.2 that including gesture features can improve the performance of language models, we conjecture that there could exist a correlation between gestures and certain semantic representations, i.e., a speaker may use certain type of gestures to convey certain meanings. We verify this guess by examining the embedding vectors of word tokens that co-locate with four selected gestures: <0>, <56>, <64>, and <72>. Among them, <0> is the label that indicates "no gesture", i.e., no body key point detected in the frame, which means the speaker is a *narrator* hidden behind the camera. It is included because of its high frequency in our data. The other three labels are among the top four most frequent gestures (as shown before). The other two frequent gestures, <36> and <63> are excluded because <63> is overwhelmingly frequent, which could result in in-balanced samples across gestures, and <36> is scarcely distributed, which makes it difficult to find sentences solely containing it. Next, we pick sentences that contain one distinct gesture, and then obtain the corresponding sentence vectors from a pre-trained BERT model [3]. The last hidden layer of 768-d for each word is collected, and the mean of all word vectors is used as the sentence vector.

We run *t*-SNE (Van der Maaten and Hinton, 2008) on all sentence vectors and show the result in fig. 8. It can be seen that the sentence vectors from different gestures cannot be visually distinguished. To further examine the possibility that the sentence vectors of certain gestures may differ from the others, we calculate the inner-group pair-wise distances (norm-2 Euclidean distance) for each gesture, and the outer-group pair-wise distances between all gestures. From the results shown in table 2, we can see that for gesture <0> and <72>, their inner-group distances are smaller than the outer-group ones, which suggest that their corresponding sentences are distributed in a semantic sub-space that is more distinguishable from other gestures. Since <0> is for **no-gesture** (narrator mode), its particularly small inner-distance value indicates that speakers do have different preferences in planning semantic content depending on whether they are in front of the camera or not. <72> is the most significant example of *actual* gestures whose inner-distance is smaller, which indicates that it is probably a gesture that co-occur



(a) Gesture (raw) sequence



(b) Gesture (compressed) sequence

Figure 6: Local entropy of gesture sequences increases with utterance position. Dots are actual data points. Lines are smoothed curves using generalized additive models (GAM). Shaded areas are 95% bootstrap confidence intervals.

solutions/hands.html

[3] https://huggingface.co/bert-base-uncased

Gesture	<0>	<56>	<64>	<72>
<0>	**8.09 (1.78)**	8.43 (1.04)	8.47 (0.89)	8.36 (1.04)
<56>	8.43 (1.04)	8.60 (2.03)	8.54 (0.89)	8.47 (1.04)
<64>	8.47 (0.89)	8.54 (0.89)	8.51 (1.65)	8.45 (0.96)
<72>	8.36 (1.04)	8.47 (1.04)	8.45 (0.96)	**8.25 (1.85)**

Table 2: Pair-wise inner-group average norm-2 Euclidean distances (diagonal cells) and outer-group average distances (other cells) between sentence vector of corresponding gestures. Standard deviations shown in parentheses.

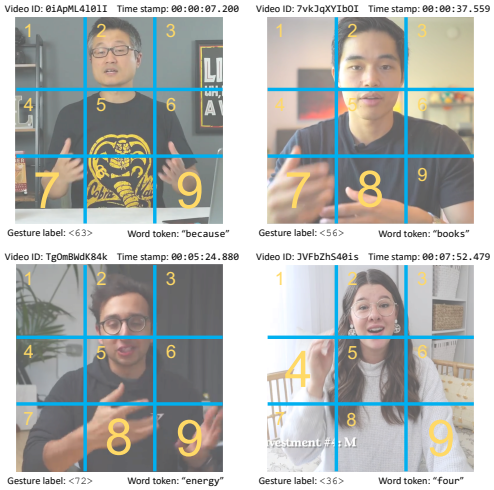


Figure 7: Selected screenshots for the top 4 frequent gestures.

with distinct meanings in sentences. This needs be further examined in future studies using more data.

To sum, we found preliminary positive evidence for associating gestures with distinct semantic meanings. However, the analysis above is limited in following aspects: First the sentences that contain gesture <0> are all from one single video, which means the findings lacks generality. Second, pre-trained embeddings are used instead of fine-tuned parameters, which can result in inaccurate description of the semantic space. We believe these limits can be overcome in our future plan.

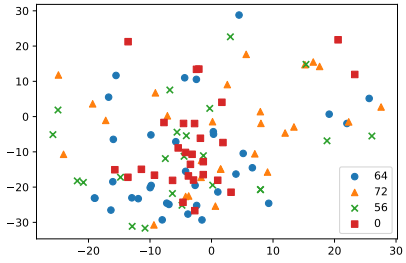


Figure 8: $t-$SNE plot of the BERT vectors obtained from utterances that include gesture tokens <0>, <56>, <64> and <72>.

6 Conclusions

Our main conclusions are two-fold: First, incorporating gestural features will significantly improve the performance of language modeling task, even when gestures are represented with a simplistic method. Second, the way gestures are used as a complementary non-verbal communication side-channel follows the principle of entropy rate constancy (ERC) in Information Theory. It means that the information encoded in hand gestures, albeit subtle, is actually organized in a *rational* way that enhances the decoding/understanding of information from a receiver's perspective. This is the first work done, to the best of our knowledge, to extend the scope of ERC to non-verbal communication.

The conclusions are based on empirical results from multi-modal language models trained on monologue speech videos with gesture information represented by discrete tokens. There are two explanations for what causes the observed entropy increasing pattern: First, more rare gestures (higher entropy) near the later stage of communication; Second, the entropy for the same gesture also increases during the communication. While the latter indicates a more sophisticated and interesting theory about gesture usage, both explanations requires further investigation.

This work is exploratory but the evidence is promising, as only a small data-set is used and a simplistic gesture representation method is applied. For future work, we plan to work with a larger and more diverse dataset with a higher variety in genres (public speech, etc.) and examine more advanced representation methods, such as continuous embedding and clustering. Another direction to pursue is to interpret the semantic meanings of gestures and other non-verbal features by examining their semantic distance from utterances in vector space. More specifically, non-parametric clustering algorithms can be useful to identify distinct dynamic actions, which provides a different way to extract non-verbal representations.

References

Simon Alexanderson, Gustav Eje Henter, Taras Kucherenko, and Jonas Beskow. 2020. Style-controllable speech-driven gesture synthesis using normalising flows. In *Computer Graphics Forum*, volume 39, pages 487–496. Wiley Online Library.

Valentin Bazarevsky, Ivan Grishchenko, Karthik Raveendran, Tyler Zhu, Fan Zhang, and Matthias Grundmann. 2020. Blazepose: On-device real-time body pose tracking. *arXiv preprint arXiv:2006.10204*.

Gedas Bertasius, Heng Wang, and Lorenzo Torresani. 2021. Is space-time attention all you need for video understanding? *arXiv preprint arXiv:2102.05095*.

Sharice Clough and Melissa C Duff. 2020. The role of gesture in communication and cognition: Implications for understanding and treating neurogenic communication disorders. *Frontiers in Human Neuroscience*, page 323.

Lifeng Fan, Yixin Chen, Ping Wei, Wenguan Wang, and Song-Chun Zhu. 2018. Inferring shared attention in social scene videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6460–6468.

Lifeng Fan, Shuwen Qiu, Zilong Zheng, Tao Gao, Song-Chun Zhu, and Yixin Zhu. 2021. Learning triadic belief dynamics in nonverbal communication from videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7312–7321.

Kirill Gavrilyuk, Amir Ghodrati, Zhenyang Li, and Cees GM Snoek. 2018. Actor and action video segmentation from a sentence. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5958–5966.

Dmitriy Genzel and Eugene Charniak. 2002. Entropy rate constancy in text. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 199–206, Philadelphia, PA.

Dmitriy Genzel and Eugene Charniak. 2003. Variation of entropy and parse trees of sentences as a function of the sentence number. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 65–72, Sapporo, Japan.

Shiry Ginosar, Amir Bar, Gefen Kohavi, Caroline Chan, Andrew Owens, and Jitendra Malik. 2019. Learning individual styles of conversational gesture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3497–3506.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Judith Holler and Stephen C Levinson. 2019. Multimodal language processing in human communication. *Trends in Cognitive Sciences*, 23(8):639–652.

Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. 2020. Hero: Hierarchical encoder for video+ language omni-representation pre-training. *arXiv preprint arXiv:2005.00200*.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*.

Minghao Li, Tengchao Lv, Lei Cui, Yijuan Lu, Dinei Florencio, Cha Zhang, Zhoujun Li, and Furu Wei. 2021. Trocr: Transformer-based optical character recognition with pre-trained models. *arXiv preprint arXiv:2109.10282*.

Vinicius Macuch Silva, Judith Holler, Asli Ozyurek, and Seán G Roberts. 2020. Multimodality and the origin of a novel communication system in face-to-face interaction. *Royal Society open science*, 7(1):182056.

David McNeill. 1992. Hand and mind1. *Advances in Visual Semiotics*, page 351.

Greg Mori, Caroline Pantofaru, Nisarg Kothari, Thomas Leung, George Toderici, Alexander Toshev, and Weilong Yang. 2015. Pose embeddings: A deep architecture for learning to match human poses. *arXiv preprint arXiv:1507.00302*.

Steven T Piantadosi. 2014. Zipf's word frequency law in natural language: A critical review and future directions. *Psychonomic Bulletin and Review*, 21(5):1112–1130.

Claude Elwood Shannon. 1948. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

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

Fei Xu, Kenny Davila, Srirangaraj Setlur, and Venu Govindaraju. 2019. Content extraction from lecture video via speaker action classification based on pose information. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1047–1054. IEEE.

Yang Xu and David Reitter. 2016. Entropy converges between dialogue participants: explanations from an information-theoretic perspective. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 537–546, Berlin, Germany.

Yang Xu and David Reitter. 2017. Spectral analysis of information density in dialogue predicts collaborative task performance. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 623–633, Vancouver, Canada. Association for Computational Linguistics.

Yang Xu and David Reitter. 2018. Information density converges in dialogue: Towards an information-theoretic model. *Cognition*, 170:147–163.

Youngwoo Yoon, Bok Cha, Joo-Haeng Lee, Minsu Jang, Jaeyeon Lee, Jaehong Kim, and Geehyuk Lee. 2020. Speech gesture generation from the trimodal context of text, audio, and speaker identity. *ACM Transactions on Graphics (TOG)*, 39(6):1–16.

George Kingsley Zipf. 2013. *The Psycho-Biology of Language: An Introduction to Dynamic Philology*. Routledge.

A Appendix

A.1 Algorithm for labeling gestures

The algorithm for labeling gestures based on the hand positions is described by the pseudo code below:

Algorithm 1 Algorithm for hand position-based labeling

Require: $0 < r = \frac{H}{W} < 1, \varepsilon = 0.001, N = 3$
Ensure: $\texttt{label} \in \{1, 2, \ldots, 81\}$

1: $\texttt{l\_shd\_x} \leftarrow x$ coord of left shoulder
2: $\texttt{r\_shd\_x} \leftarrow x$ coord of right shoulder
3: $\texttt{l\_hip\_x} \leftarrow x$ coord of left hip
4: $\texttt{r\_hip\_x} \leftarrow x$ coord of right hip
5: $\texttt{nose\_x} \leftarrow x$ coord of nose
6: $x_c = (\texttt{nose\_x} + \frac{\texttt{l\_shd\_x}+\texttt{r\_shd\_x}}{2} + \frac{\texttt{l\_hip\_x}+\texttt{r\_hip\_x}}{2})/3$
7: $x_{\text{left}} = x_c - 0.5 \cdot r + \varepsilon$
8: $x_{\text{right}} = x_c + 0.5 \cdot r - \varepsilon$
9: $\texttt{w} = x_{\text{right}} - x_{\text{left}}$
10: $y_{\text{bot}} = \varepsilon$
11: $y_{\text{top}} = 1 - \varepsilon$
12: $\texttt{h} = y_{\text{top}} - y_{\text{bot}}$
13: $\texttt{l\_hnd\_x} \leftarrow x$ coord of left hand
14: $\texttt{r\_hnd\_x} \leftarrow x$ coord of right hand
15: $\texttt{l\_hnd\_y} \leftarrow y$ coord of left hand
16: $\texttt{r\_hnd\_y} \leftarrow y$ coord of right hand
17: $\texttt{l\_col} = \frac{\lfloor \min(\max(\texttt{l\_hnd\_x} - x_{\text{left}}, 0), \texttt{w}) \rfloor}{r} \cdot N + 1$
18: $\texttt{r\_col} = \frac{\lfloor \min(\max(\texttt{r\_hnd\_x} - x_{\text{left}}, 0), \texttt{w}) \rfloor}{r} \cdot N + 1$
19: $\texttt{l\_row} = \frac{\lfloor \min(\max(\texttt{l\_hnd\_y} - y_{\text{bot}}, 0), \texttt{h}) \rfloor}{r} \cdot N + 1$
20: $\texttt{r\_row} = \frac{\lfloor \min(\max(\texttt{r\_hnd\_y} - x_{\text{bot}}, 0), \texttt{h}) \rfloor}{r} \cdot N + 1$
21: $\texttt{l\_label} = \lfloor (\texttt{l\_row} - 1) \cdot N + \texttt{l\_col} \rfloor$
22: $\texttt{r\_label} = \lfloor (\texttt{r\_row} - 1) \cdot N + \texttt{r\_col} \rfloor$
23: $\texttt{label} = (\texttt{l\_label} - 1) \cdot N^2 + \texttt{r\_label}$

The algorithm takes an image frame of size $H \times W$ (pixels) as input ($H = 720,\ W = 1280$ for most videos). $r = H/W$ is the ration of frame height over width, and thus its value is fixed as $r = 720/1280 = 0.5625$ in our data. All x and y coordinates returned by the body key points detector (Mediapipe) are relative values within the range of $[0, 1]$. We have also observed that a $H \times H$ square region centered around the central axis of body can consistently cover the speaker's hands, so that is why we use r as the relative width to define the left and right boundaries of the $N \times N$ split areas (line 7 and 8). The resulting label for left hand $\texttt{l\_label} \in \{1, \ldots, N\}$, and label for right hand $\texttt{r\_label} \in \{1, \ldots, N\}$. According to line 23, the final label combing information from both hands $\texttt{label} \in \{1, 2, \ldots, N^2\}$, which contains 81 distinct labels when $N = 3$.

The code for the labeling algorithm will be published in a public repository under the MIT license.

A.2 Hyper-parameters and training procedures

For the LSTM-based encoder, embedding size is 300, hidden size is 200, number of layers is 2; a fully connected layer is used as the decoder connecting the encoder output and the softmax; dropout layers of probability 0.2 are applied to the outputs of both the encoder and decoder. For the Transformer-based encoder, model size is 20, hidden size is 100, number of layers is 2; same fully connected linear decoder is used; dropout layers of probability 0.5 are used at the position encoding, and each transformer encoder layer. To enable the one-direction (left to right) modeling effect, a mask matrix (of 0 and 1s) in an upper-triangular shape is used together with each input sequence.

Model parameters are randomly initialized. Training is done within 40 epochs, with batch size of 20, at and initial learning rate $lr = 0.05$. SGD optimizer with default momentum is used for training the LSTM model; Adam optimizer is used for training the Transformer model. Data are split to 80% for training and 20% for testing. After each training epoch, evaluation is done over the test set, and the model with lowest perplexity scores is saved as the best one.

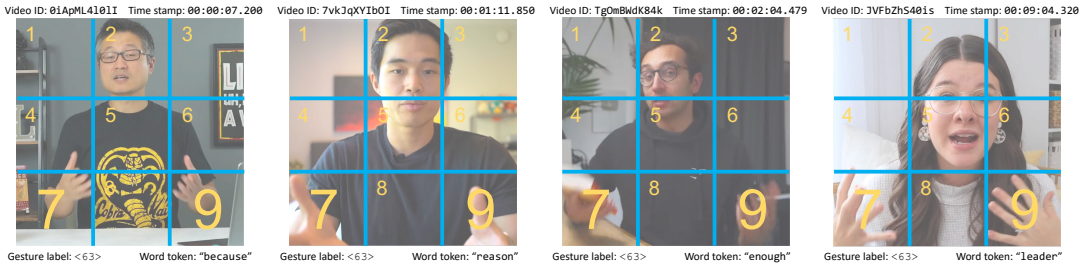
Models are implemented with PyTorch. `torch.nn.CrossEntropyLoss` module is used as the loss function. The mathematical meaning of the output from this function is the negative logarithm likelihood (*NLL* in eq. (2)), and thus we compute the exponential values of the output to get the local entropy scores. The entropy scores used in the plot and statistical analysis are obtained from both train and test sets. Models are trained on 4 Nvidia 1080Ti GPU cards. The total GPU hours needed is about 2 hours.

The code for training, testing the language models will be published in a public repository under the MIT license. The binary files of trained model will also be provided via URLs included in the repository. The intended use of the trained language models are for scientific research about general patterns in human non-verbal communication, but not for identification of individual speakers, nor

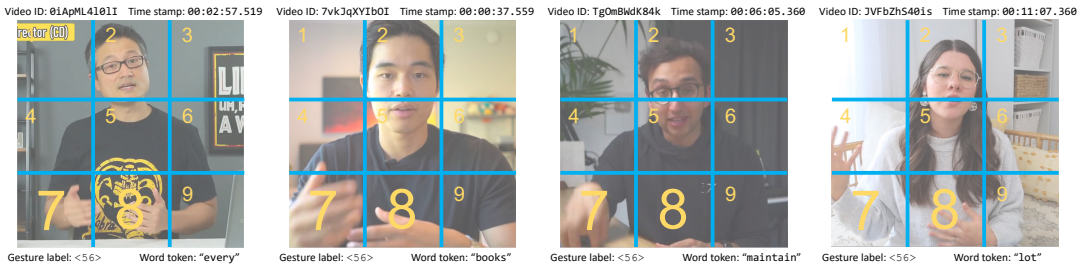
for other commercial use.

A.3 Screenshots for frequent gestures

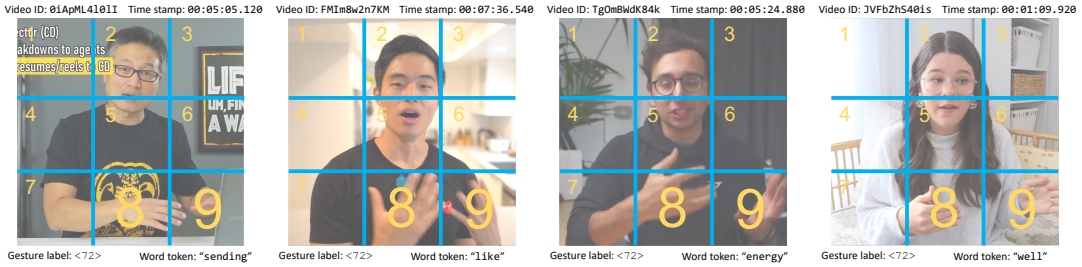
Some typical screenshots for the top 4 frequent gestures from all four speakers are shown in fig. 9. We can find similar appearance of same gestures across different speakers.



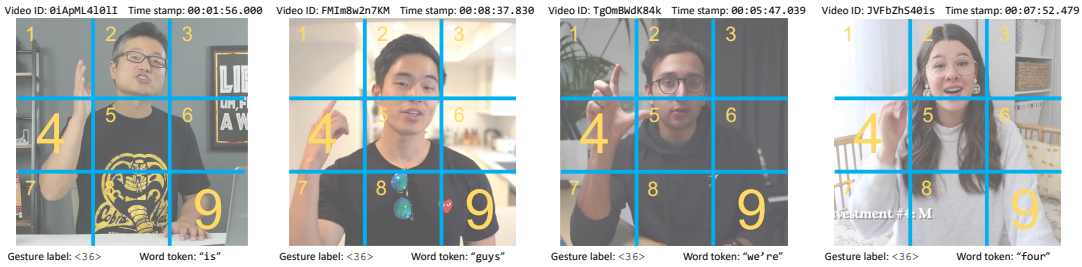
(a) Gesture label <63>



(b) Gesture label <56>



(c) Gesture label <72>



(d) Gesture label <36>

Figure 9: Typical screenshots for gesture labels <63>, <56>, <72> and <36>.