Spontaneous gestures encoded by hand positions can improve language models: An Information-Theoretic motivated study

Yang Xu
Department of Computer Science
San Diego State University
yang.xu@sdsu.edu

Yang Cheng
University of Southern California
ycheng04@usc.edu

Abstract

The multi-modality nature of human communication has been utilized to enhance the performance of language modeling-related tasks. Driven by the development of large-scale end-to-end learning techniques and the availability of multi-modal data, it becomes possible to represent non-verbal communication behaviors through joint-learning, and directly study their interaction with verbal communication. However, there are still gaps in existing studies to better address the underlying mechanism of how non-verbal expression contributes to the overall communication purpose. Therefore, we explore two questions using mixed-modal language models trained against monologue video data: first, whether incorporating gesture representations can improve the language model's performance (perplexity); second, whether spontaneous gestures demonstrate entropy rate constancy (ERC), which is an empirical pattern found in most verbal language data that supports the rational communication assumption from Information Theory. We have positive and interesting findings for both questions: speakers indeed use spontaneous gestures to convey "meaningful" information that enhances verbal communication, which can be captured with a simple spatial encoding scheme. More importantly, gestures are produced and organized rationally in a similar way as words, which optimizes communication efficiency.

1 Introduction

Human communication is a multi-modal process where both verbal and non-verbal information are expressed simultaneously. This is true in various forms of communication, one-way (speech) or two-way (conversation). It has been revealed in empirical studies that speakers' expression in the 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. It is becoming an emerging sub-area in computational linguistics. However, whether and to what degrees these sparse and random non-verbal signals can be treated as a formal communication channel that transmits "serious" information remains a seldom-validated question, especially with computational methods. We believe 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 straightforward framework to approach the above questions, under the guidance of Information Theory. Some preliminary, yet prospective results are presented. The code and data for this study is published in this repository https://github.com/innerfirexy/Life-lessons.

2 Related Work

2.1 Studies on gestures in communication

Early studies and theories

The functions of gestures in communication and the connection to verbal language have been extensively studied in behavioral science, psychology and cognitive sciences. McNeill (1992) has developed the Growth Point theory, which can be conceptualized as a "snapshot" of an utterance at its beginning stage psychologically. McNeill (1992)'s theory classifies gestures into two categories, representative ones, which have clearer semantic meanings (e.g., depicting objects and describing locations), and non-representative ones, which refer to the repetitive movements that have little substan-

Findings of the Association for Computational Linguistics: ACL 2023, pages 9409–9424
July 9-14, 2023

tive meanings. McNeill et al. (2008) further put forward a more fine-grained classification scheme for gestures: *iconic*, *metaphoric*, *deictic*, and *beats*, in which the iconic and metaphoric gestures are directly related to the concrete and abstract content in the verbal language. The psycholinguistics theories and studies indicate the feasibility of investigating the "meanings" of gestures with computational semantic approaches.

Lab-based experimental studies
The effect of gestures has been broadly studied in laboratory-based behavioral experiments. Holler and Levinson (2019) study the facilitation from multiple layers of visual and vocal signals can add semantic and pragmatic information in face-to-face communication. Similarly, Macuch Silva et al. (2020) find visible gestures are more powerful form of communication than vocalization in dialogue object description tasks. In these studies, gestures from human subjects are usually *manually coded* by observing the hands' spatial positions and motions to characterize naturalistic and meaningful movements. Trujillo et al. (2019) takes a step forward and develops a protocol for automatically extracting kinematic features from video data, which can be applied to quantitative and qualitative analysis of gestures. Their work provides insight to the hands position-based encoding method for gestures (discussed in section 4.2).

Computational studies
More recently, the communicative functions of gestures have been studied in different settings from human-human to human-agent interaction interactions. Synthesized gestures are integrated into virtual characters and robots to facilitate the dialogue fluidity and user experiences (Kopp, 2017). In such systems, the content and form of co-speech gestures are determined from the semantic meanings of utterances being produced (Hartmann et al., 2006), and/or from given communication goals and situations (Bergmann and Kopp, 2010). The success of these systems also indicates the possibility of understanding gestures in the wild by learning language models that include simple gestural features.

To summarize, the works reviewed above have paved the road for studying gestures in a more "data-driven" style, that is, using data collected from more naturalistic contexts and more automatic methods for encoding gestures.

2.2 Multi-modal techniques in machine learning and NLP research

The recent advances in 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 characterize 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 in complex models.

2.3 The theoretical basis of informative communication

To what degrees do non-verbal actions contribute to informative communication? Other than the empirical works reviewed in section 2.1, the same question can also be explored from the perspective of abstract theories. (Sandler, 2018) draws evidence from sign languages to show that the actions of hands and other body parts reflect the *compositional* nature of linguistic components (their methods are further discussed in section 4.2). Their work reveals that the use of bodily articulators maps the way a verbal language origins and evolves. Although the spontaneous gestures of our interest here are different from a strictly defined sign language, Sandler (2018)'s work inspires us that more

similar properties can be found between verbal and non-verbal languages at a higher level of abstraction. Information Theory (Shannon, 1948) is the next lens that we use.

Information Theory is broadly applied as the theoretical background for the probabilistic models of language. It also provides philosophical explanations for a broad spectrum of linguistic phenomena. One interesting example 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 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, which is confirmed in Genzel and Charniak (2003)'s work. 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).

The term $H(s|L)$ can be estimated with various methods. Genzel and Charniak (2002, 2003) used the average negative log-probability of all n-grams in a sentence to estimate $H(s|L)$, and the probabilities are returned from an n-gram language model. Some more recent works have used transformer-based neural language models to examine ERC in dialogue (Giulianelli et al., 2021, 2022) and in broader data modalities with various operationalizations (Meister et al., 2021).

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 tasks. 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 of 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, 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 prepara-

tion for the next gesture representation step. For this task, we use the BlazePose (Bazarevsky et al., 2020) model, which is a lightweight convolutional neural network-based pose estimator provided in `MediaPipe`[1]. It outputs the (x, y) coordinates of 33 pose landmarks that characterize the key points of the body pose, including {*nose*, *left-eye*, . . . , } (see (Xu et al., 2020) for full description). Here each coordinate (x, y) is a pair of fraction values within $[0, 1]$ describing the key point's relative position on a frame, whose zero point is at the upper left corner. In fact, the pose estimator returns a 3-D coordinate (x, y, z) for each point, where the third dimension z is the depth. We discard this z component based on our observation that most speakers do not show hand movement in that direction.

4.2 Encode gestures based on hands' positions

The next step is to obtain *representation* for gestures so that they can be studied using language models in a similar way as word embeddings. After having surveyed extensively on previous studies about methods of encoding gestures, we decide to develop an encoding scheme that categorizes gestures into discrete *token* based on the positions of hands, which are inspired by the work of (Trujillo et al., 2019; Sandler, 2018). To briefly summarize their work, Trujillo et al. (2019) measure the vertical amplitude feature of the right dominant hand in relation to a participant's body (upper-left of fig. 1); Sandler (2018) use the relative positions of dominant and non-dominant hands between torso and face as the evidence for the hierarchical organization of body language (lower-left of fig. 1).

The workflow of our method is in three steps. The first two steps are to identify the **focus area** of the speaker's upper body, which a square area whose size *almost* equals the height of the upper body. We come up with this empirical setup based on the observation that this square area covers the vast majority of possible hand positions in our data. The third step is to encode the gesture based the relative positions of hands within the focus area.

1) Compute the horizontal center of the body x_{center} by averaging the x coordinates of *nose*, left & right *shoulders*, and left & right *hips*.

2) Find the vertical boundaries of the body area. First, compute the vertical distance between the nose and the mid-point of two eyes, $\delta = |y_{\text{nose}} - y_{\text{eyes}}|$. Then the top bound (forehead) is calculated by: $y_{\text{min}} = y_{\text{eyes}} - 2\delta$. This is according to the common knowledge about proportions of the human head (Artyfactory, 2022). The bottom bound y_{max} is the mean y coordinates of both *hips* because the speakers are in a sitting pose and only their upper bodies are visible. Lastly, obtain the size of focus area by $y_{\text{max}} - y_{\text{min}}$.

3) Divide the focus area into 3×3 regions, i.e., nine regions with indices $\{1, 2, \ldots, 9\}$. Index each hand with an integer based on which region it is in, and then encode the gesture into an integer number, using the combination of both hands' indices. The encoding formula is:

$$g(L, R) = (L - 1) \cdot 3^2 + R \qquad (2)$$

in which L and R are the region index for the left and right hand, respectively. This formula maps any combination of (L, R) to a distinct integer number g, which we call **gesture token**.

As shown in the example of fig. 1, 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 tokens is $9 \times 9 = 81$. For the convenience of the modeling step later, we use one integer index (instead of a string connected by hyphen) to denote each of these 81 gestures: `<1>`, `<2>`, ..., `<81>`. The pseudo code is presented in appendix A.1. Some notes: why not string but integer. We understand that encoding (L, R) into an integer number is not as straight-forward to interpret the gesture as another method of simply representing it with a string, such as "L-R" to indicate left hand in region *L* and right hand in region *R*. But using an integer index has the advantage that the gesture tokens can be directly supplied to the language models, just like word indices.

4.3 Prepare gesture sequences

Having gestures encoded, we prepare the gesture sequences using the time stamped 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>
```

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

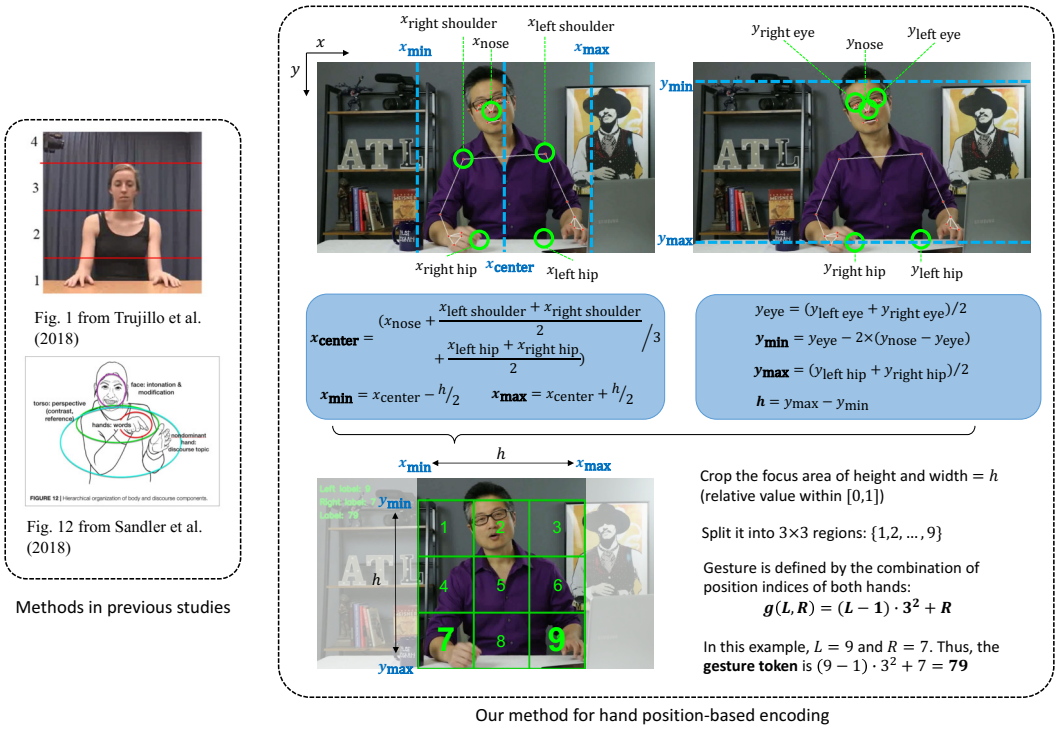


Figure 1: The method for encoding gestures based on hand positions.

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 (seconds). Multiplying the time stamps with the frame rate of 24 (FPS), which means the frame range is from the 19th to 24th. Then, for each frame within this range, we extract a gesture token using the method described in Section 4.2, resulting in a sequence of gesture tokens, $\{g_{19}, g_{20}, \ldots, g_{24}\}$. This sequence represents a continuous change of gestures during the articulation of the word, which in most cases, consists of identical tokens. Thus, we select the majority token g^m within the sequence as the final representation.

Applying the above process to an utterance consisting of N words, $\{w_1, w_2, \ldots, w_N\}$, we can obtain N majority gesture tokens, $\{g_1, g_2, \ldots, g_N\}$. Despite the down sampling effect of using majority sampling, there is still a large amount of repetition in the resulted gesture sequence, which could cause sparsity issues for the modeling tasks. For instance, in the first row of table 1, the gesture token 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. We deal with this issue by "compressing" the repeated gesture tokens. For the same example in table 1, merging the 6 repeats of <49> and 2 repeats of <76> results in a compressed gesture sequence, {<49>, <76>}, which indicates that the speaker has made two distinct gestures during the utterance. Throughout the rest of the paper, we call the original gesture sequence that come with repeats the *raw* sequence, and the one with repeats merged the *compressed* sequence. For each raw gesture sequence of length N, its compressed version $\{\hat{g}_1, \hat{g}_2, \ldots, \hat{g}_{N'}\}$ usually has smaller length $N' \leq N$.

4.4 Incorporate gesture inputs to LMs

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 converts 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

Word tokens	Raw gesture tokens $\{g\}$	Compressed gesture sequence $\{\hat{g}\}$
going to give you a flatter look glossy	<49> <49> <49> <49> <49> <49> <76> <76> ($N = 8$)	<49> <76> ($N' = 2$)
now this is really your preference	<44> <80> <71> <71> <44> <44> ($N = 6$)	<44> <80> <71> <44> ($N' = 4$)
I think most of us can get on board	<79> <79> <79> <79> <79> <79> <79> <79> <79> ($N = 9$)	<79> ($N' = 1$)

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

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:

$$NLL = -\sum_{k=1}^{K} \log P(t_k | t_1, t_2, \ldots, t_{k-1}) \quad (3)$$

in which $t_1, \ldots, t_{k-1}$ is all the tokens (gesture or word) before t_k within the same utterance. We directly use this *NLL* value as the estimated local entropy, i.e., $H(g|L) \triangleq NLL$, which is the target variable of our interest. Detailed model hyperparameters 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 hyperparameters will be presented in the Appendix.

5 Results

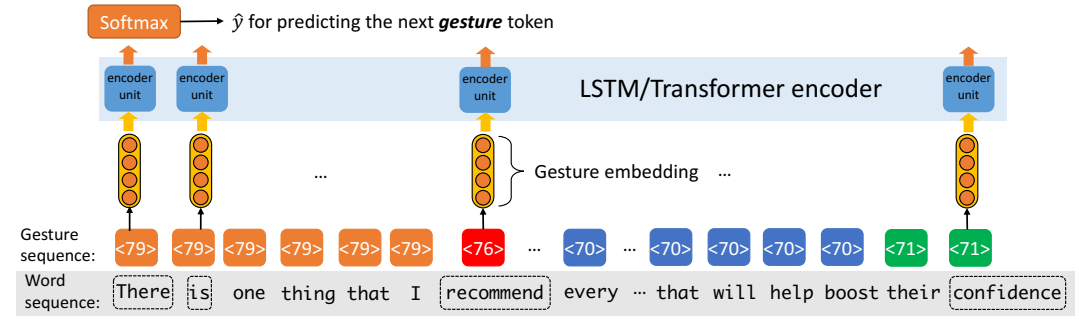
5.1 Statistics

62 videos of a total length of 10 hours and 39 minutes are collected. The average length of each video is 723.7 seconds (*SD* = 438.1). The data and preprocessing scripts will be open-sourced. 17.9K lines of subtitles consisting of 121.5K words are collected. We have extracted 81 distinct gesture tokens, whose total number is 121.5K in the raw sequence data (equals the total number of words). Within the compressed sequence data, the total number of gesture tokens is reduced to 26.12 K.

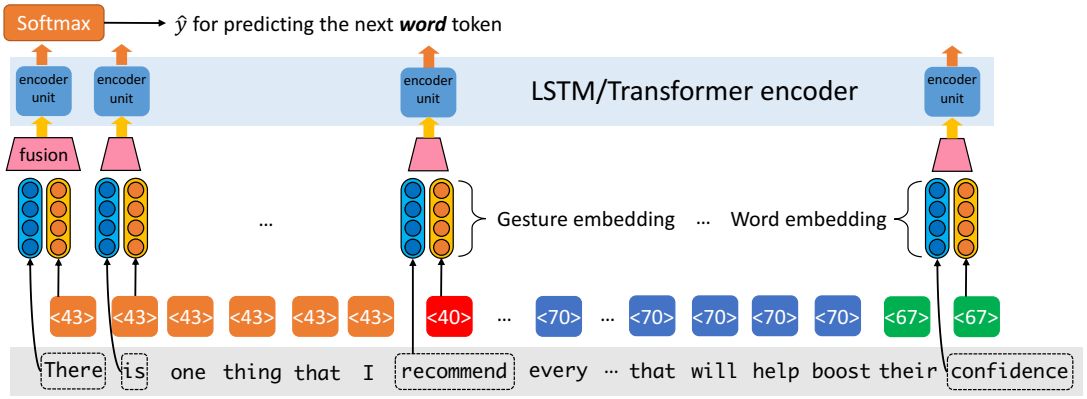
The top five most frequent gesture tokens (according to the raw, uncompressed data) are <79>, <71>, <70>, <80> and <76>. Their frequency counts, proportions, and the average entropy values are shown in section 5.1. It can be seen that <79> is the dominant gesture token, where the speaker's right hand falls in region 7 and left hand in region 9. The entropy value increases as the frequency rank drops, which roughly follows the Zipf's law (see the frequency vs. rank plots in fig. 3). Because Zipf's law is a common distribution for word tokens (Zipf, 2013; Piantadosi, 2014), it is a side evidence showing that gestures encode semantic information in a similar way as words. A detailed analysis of the gestures' positional and semantic meanings is provided in section 5.4.

Token	Freq.	Prop.	Entropy
<79>	42367	0.349	2.97
<71>	20540	0.169	6.06
<70>	20354	0.167	6.25
<80>	9264	0.076	13.99
<76>	2762	0.023	51.58

Table 2: The frequency count, proportion, and entropy values of the top five frequent gesture tokens.



(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.

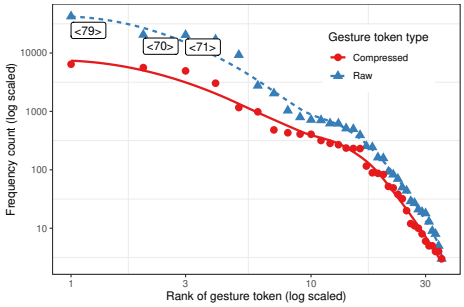


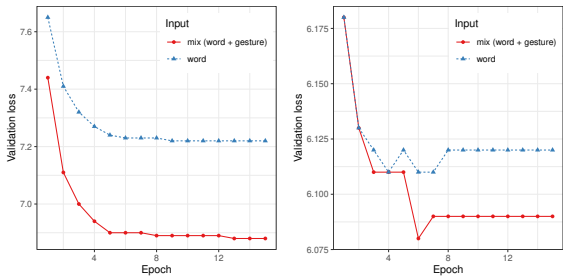
Figure 3: Frequency count against the rank of gesture tokens in logarithm transformed scales. The top three frequent ones are annotated.

5.2 Examining Hypothesis 1: Mixed vs. single modal comparison

The plots of validation cross-entropy 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.

Note that an exponential conversion of the cross-entropy loss (i.e., the *NLL* in eq. (3)) leads to another quantity *perplexity*, which is more commonly used to evaluate the performance of language models. The Transformer-based models have overall lower perplexity than LSTM-based ones, which is expected as a Transformer encoder has 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.

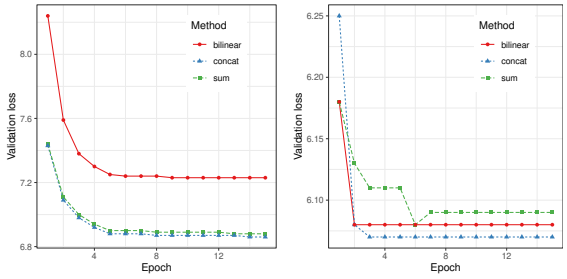
We also compare three different feature fusion method in training the m-LSTM/Transformer models, and found that *sum* and *concat* have better performance (lower loss) in language modeling tasks. The corresponding validation losses for three feature fusion methods, *sum*, *concat*, and *bilinear*, are shown in fig. 5. It can be seen that *sum* and *concat* result in a significantly lower loss for *m*-LSTM, but



(a) LSTM model (b) Transformer model

Figure 4: The cross-entropy loss on the validation set against training epochs for mixed- and single-modal models.

the difference is not that observable in *m*- Transformer because in the latter loss shortly converges after training starts.

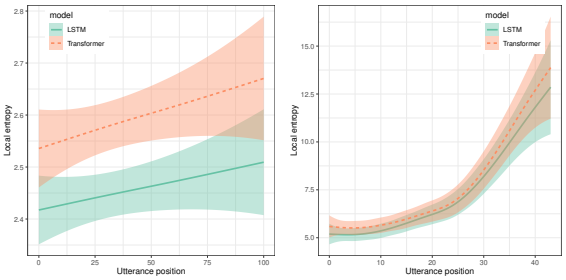


(a) *m*-LSTM model (b) *m*-Transformer model

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

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) Raw gesture (b) Compressed gesture

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). 95% bootstrap CIs presented.

5.4 Analysis of typical gestures

We examine the top five frequent gesture tokens <79>, <71> <70>, <80> and <76>, and show some selected screenshots in fig. 7 (See appendix A.3 for more examples). For <79>, <70> and <80>, the positions of both hands are at the mid-lower position in front of the body. Gesture <79> has two hands evenly distant from the center, while <70> captures a movement to the right and <80> to the left. Gesture <76> 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. One technical detail is that in most screenshots of <76> the left hands are invisible, but the pose estimation algorithm can still infer their positions with accuracies above 95% (see the report from Mediapipe), which is also why they are included 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 colocate with three selected gestures: <70>, <71>, and <80>. Two other frequent gestures, <79> and <76> are excluded from the analysis because: <79> is overwhelmingly frequent, which could result in in-balanced samples across gestures; <76> is scarcely distributed, which makes it difficult to find sentences solely containing it. Next, we pick sentences that contain one distinct gesture, and then

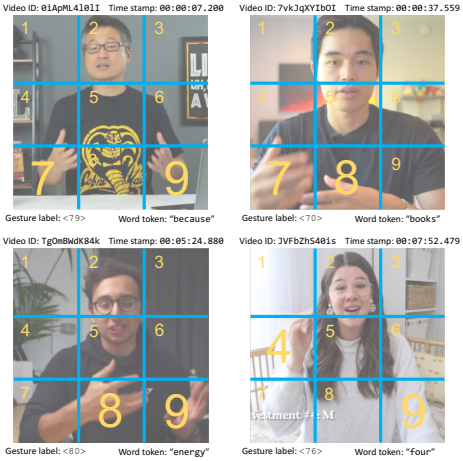


Figure 7: Examples of four frequent gestures: <79> (Upper left), <70> (Upper right), <80> (Lower left) and <76> (Lower right).

obtain the corresponding sentence vectors from a pre-trained BERT model. The last hidden layer of 768-d for each word is used to compute the mean sentence vector.

Gesture	<70>	<71>	<80>
<70>	**.291 (.007)**	.298 (.007)	.298 (.008)
<71>	.298 (.007)	.304 (.009)	.305 (.008)
<80>	.298 (.008)	.305 (.008)	.305 (.008)

Table 3: Pair-wise inner-group average cosine distances (diagonal cells) and outer-group average distances between sentence vector of corresponding gestures. Standard deviations shown in parentheses.

We calculate the inner-group pair-wise cosine distances for each gesture, and the outer-group pair-wise distances between all gestures. From the results shown in table 3, we can see that for gesture <70>, its inner-group distance (.291) is smaller than the outer-group ones (.298 and .298), with which t-tests yield $p < .001$ results. It suggests that its corresponding sentences are distributed in a semantic sub-space farther away from others, and <70> is probably a gesture that co-occurs with some particular meanings. This needs to be further examined in future studies with 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 data come from a limited population, which means the findings about gesture semantics may lack generality. Second, pre-trained embeddings are used instead of fine-tuned ones, which can result in inaccurate description of the semantic space. We believe these limits can be overcome in our future studies.

6 Conclusions

Our main conclusions are two-fold: First, incorporating gestural features will significantly improve the performance of language modeling tasks, 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 pattern of increasing entropy: 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 require further investigation.

This work is exploratory, but the evidence is promising, as only a small dataset 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.

Acknowledgements

This work is supported by National Science Foundation of the United States (CRII-HCC: 2105192). We sincerely thank all the reviewers for their efforts in pointing out the mistakes in the paper and their insightful advice for future improvement.

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.

Artyfactory. 2022. The proportions of the head. https://www.artyfactory.com/portraits/pencil-portraits/proportions-of-a-head.html. Accessed: 2023-01-14.

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*.

Kirsten Bergmann and Stefan Kopp. 2010. Modeling the production of coverbal iconic gestures by learning bayesian decision networks. *Applied Artificial Intelligence*, 24(6):530–551.

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

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.

Mario Giulianelli, Arabella Sinclair, and Raquel Fernández. 2021. Is information density uniform in task-oriented dialogues? In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8271–8283.

Mario Giulianelli, Arabella Sinclair, and Raquel Fernández. 2022. Construction repetition reduces information rate in dialogue. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing*, pages 665–682.

Björn Hartmann, Maurizio Mancini, and Catherine Pelachaud. 2006. Implementing expressive gesture synthesis for embodied conversational agents. In *International Gesture Workshop*, pages 188–199. Springer.

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.

Stefan Kopp. 2017. Computational gesture research: Studying the functions of gesture in human-agent interaction. In R Breckinridge Church, Martha W Alibali, and Spencer D Kelly, editors, *Why Gesture?: How the hands function in speaking, thinking and communicating*, volume 7, chapter 12, pages 267–284. John Benjamins Publishing Company.

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 mind. *Advances in Visual Semiotics*, page 351.

David McNeill, Susan D Duncan, Jonathan Cole, Shaun Gallagher, and Bennett Bertenthal. 2008. Growth points from the very beginning. *Interaction Studies*, 9(1):117–132.

Clara Meister, Tiago Pimentel, Patrick Haller, Lena Jäger, Ryan Cotterell, and Roger Levy. 2021. Revisiting the uniform information density hypothesis. *arXiv preprint arXiv:2109.11635*.

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.

Wendy Sandler. 2018. The body as evidence for the nature of language. *Frontiers in Psychology*, 9:1782.

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

James P Trujillo, Julija Vaitonyte, Irina Simanova, and Asli Özyürek. 2019. Toward the markerless and automatic analysis of kinematic features: A toolkit for gesture and movement research. *Behavior Research Methods*, 51(2):769–777.

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.

Hongyi Xu, Eduard Gabriel Bazavan, Andrei Zanfir, William T Freeman, Rahul Sukthankar, and Cristian Sminchisescu. 2020. Ghum & ghuml: Generative 3d human shape and articulated pose models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6184–6193.

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 position-based gesture encoding

The algorithm for encoding gestures based on the hand positions is described by the following pseudo code:

Algorithm 1 Hand position-based gesture encoding

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: $\texttt{l\_index} = \lfloor (\texttt{l\_row} - 1) \cdot N + \texttt{l\_col} \rfloor$
22: $\texttt{r\_index} = \lfloor (\texttt{r\_row} - 1) \cdot N + \texttt{r\_col} \rfloor$
23: $\texttt{token} = (\texttt{l\_index} - 1) \cdot N^2 + \texttt{r\_index}$
24: return $\texttt{token}$

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 the 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 index for left hand $\texttt{l\_index} \in \{1, \ldots, N\}$ and right hand $\texttt{r\_index} \in \{1, \ldots, N\}$. According to line 23, the final gesture token combing information from both hands $\texttt{token} \in \{1, 2, \ldots, N^2\}$, which contains 81 distinct values when $N = 3$. The code for the encoding algorithm will be published in a public repository under the MIT license.

A.2 Hyper-parameters and training procedures

The LSTM-based encoder has an embedding size of 300 and hidden size of 200, with 2 layers; 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, the 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, 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 into 80% for training and 20% for testing. After each training epoch, evaluation is done over the test set, and the model with the lowest perplexity scores is saved as the best one.

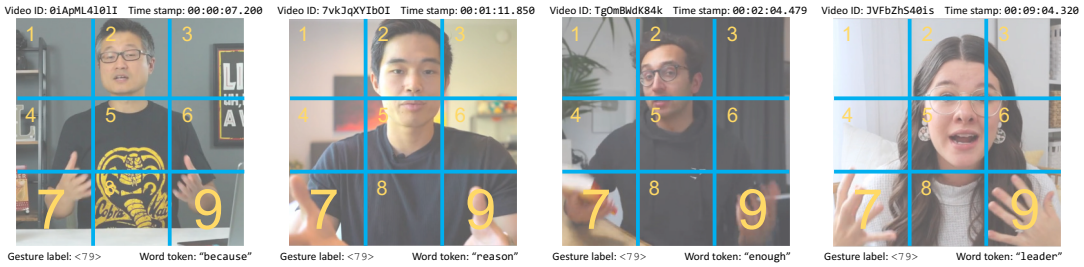
Models are implemented with PyTorch. `torch.nn.CrossEntropyLoss` module is used as the loss function. The mathematical meaning of the output from this function is the negative logarithm likelihood (*NLL* in eq. (3)), and thus we compute the exponential values of the output to get the local entropy scores. The entropy scores used in the plot and statistical analysis are obtained from both train and test sets. Models are trained on 2 Nvidia A1000 cards. The total GPU hours needed is about 2 hours.

The code for training and testing the language models will be published in a public repository under the MIT license. The binary files of the trained model will also be provided via URLs included in the repository. The intended use of the trained language models is for scientific research about

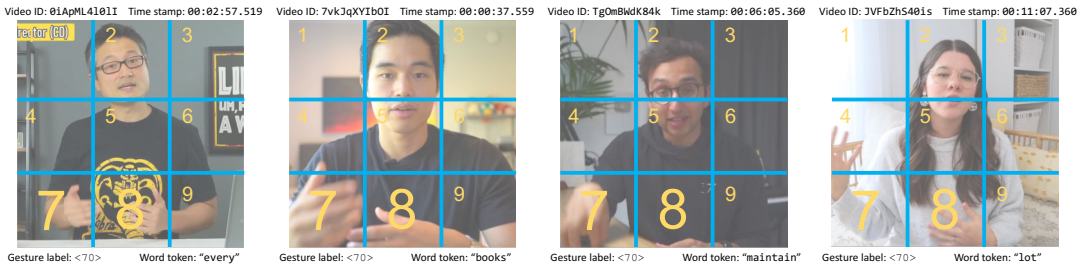
general patterns in human non-verbal communication, but not for the identification of individual speakers nor for other commercial use.

A.3 Screenshots for frequent gestures

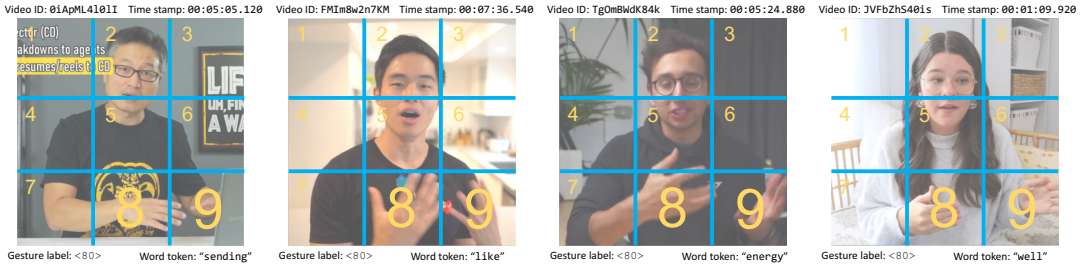
Some typical screenshots for the top 4 frequent gestures from all four speakers are shown in fig. 8. We can find similar appearances of the same gesture across different speakers.



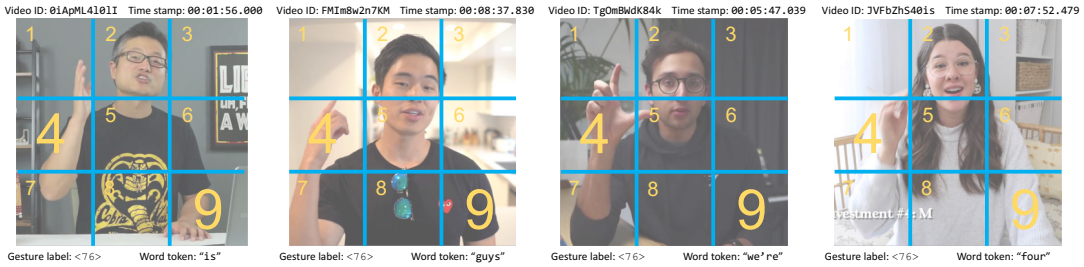
(a) Gesture label <79>



(b) Gesture label <70>



(c) Gesture label <80>



(d) Gesture label <76>

Figure 8: Typical screenshots for gesture tokens <79>, <70>, <80> and <76>.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section 5.4 (last paragraph)

☒ A2. Did you discuss any potential risks of your work?
No potential risks from this study is identified. The data and model are small scaled, and no user oriented system is developed.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
The abstract summarizes the main results and conclusions. The introduction motivates the study.

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Section 4.4, LSTM and Transformer are cited.

☑ B1. Did you cite the creators of artifacts you used?
Section 4.4, LSTM and Transformer are cited.

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
The discussion of license and terms of use for the annotation algorithm and the code for training/testing models is provided in Appendix A.1 and A.2

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
The intended use of the model created in this study is discussed in Appendex A.2 (last paragraph).

☒ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
The video data used in this study are publicly available on YouTube. Using public vidoes for scientific research conforms to the copyright policy.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
The description of data source (language, demographic groups) is provided in Section 4.1

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Statistic of data (e.g., token numbers) are reported in Section 5.1. The train/test/dev splits is described in Appendix A.2

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

C ☑ Did you run computational experiments?

Section 5.2, 5.3, and 5.4

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
The number of parameters, GPU cards, and GPU hours are provided in Appendix A.2 (second paragraph).

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
The best parameters used are provided in Appendix A.2 (first paragraph).

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Error bars (shaded areas) indicating 95% bootstrap confidence intervals are plotted.

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
PyTorch is used for implementing the models. The use of specific loss function is discussed in Appendix A.2 (third paragraph).

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.