

Modeling natural language emergence with integral transform theory and reinforcement learning

Bohdan B. Khomtchouk, Shyam Sudhakaran*

*Stanford University
Department of Biology
Gilbert Hall, 371 Serra Mall,
Stanford, CA 94305, USA,
Quiltomics
Palo Alto, CA 94306, USA*

Zipf’s law predicts a power-law relationship between word rank and frequency in language communication systems and has been widely reported in a variety of natural language processing applications. Information-theoretic simulations have shown that language communication systems emerge at an abrupt phase transition in the fidelity of mappings between symbols and objects. The emergence of natural language is often modeled as a function of bias between speaker and listener interests, which provides no direct way of relating information-theoretic bias to Zipfian rank. Here we bridge information-theoretic methods with natural language processing techniques by proposing a novel integral transform and kernel for mapping communicative bias functions to corresponding word frequency-rank representations at any arbitrary phase transition point. We also propose a reinforcement learning simulation to provide real-life evidence supporting the preference of modelling communicative systems as a function of rank over a function of bias.

Keywords: Natural language processing — information theory — integral transform theory — theoretical computer science — deep learning — reinforcement learning — computer vision

INTRODUCTION AND BACKGROUND

The linguist George Kingsley Zipf made the observation that the frequency of a word is proportional to the inverse of the word’s rank in a text. If the most common word occurs at frequency n , then the second most common word occurs at frequency $n/2$, the word with rank three at frequency $n/3$, etc. Generalized, Zipf’s law [1] states:

$$f \propto \frac{1}{r^\alpha} \quad (1)$$

where r is the word rank and f the frequency in the text, and α is the scaling coefficient generally found to be near 1.0 for many of the texts examined [2–5].

Ferrer i Cancho’s research group formalized the least-effort principle as it applies to Zipf’s law [2, 10–12] by employing a mutation-driven genetic algorithm. Here the listener and speaker have different and conflicting interests. The listener seeks to gain as much information as possible from a communicative exchange, and would benefit if there were no ambiguity between word-object mappings. This is the case in which the correlation between words and objects is highest; in information theory [13], this corresponds to a high mutual information, or $I(S, R)$ where S represents the symbol and R the referent or object. The speaker on the other hand looks to minimize his effort in communicating and would benefit from fewer words to choose from, assuming that the choice of words comes with an effort; in information theory, this is quantified using information entropy or $H(S)$. To this end, Ferrer i Cancho [2] introduced an energy function based on information theory that models the speaker’s and listener’s interests:

$$\Omega(\lambda) = \lambda I(S, R) - (1 - \lambda)H(S) \quad (2)$$

where λ ($0 < \lambda < 1$) controls the balance between the speaker interests, $H(S)$, and listener interests, $I(S, R)$. It is found [2, 10, 11] that natural languages emerge at the phase transition (Fig. 1) near $\lambda^* \approx 0.5$ (i.e., when listener and speaker interests are weighted about equally). For $\lambda < \lambda^*$, there is little or no communication because there are few words in the lexicon $\langle L \rangle$ (Fig. 1B) while, it is assumed, the number of objects remains constant which produces tremendous ambiguity in word-meaning mappings – one or a few words point to all the objects (i.e., low $I(S, R)$, (Fig. 1A)). For $\lambda > \lambda^*$, there is extremely efficient communication involving single word-single object mappings (i.e., high $I(S, R)$) – though this comes at a high cost for the speaker (i.e., high $H(S)$) because the lexicon abruptly rises to the number of objects.

The form of both of these phase transitions (Fig. 1) lies somewhere between a step or Heaviside function and a ramp function (Fig. 2). The unit ramp function increases gradually, one unit per unit time. The abrupt switching

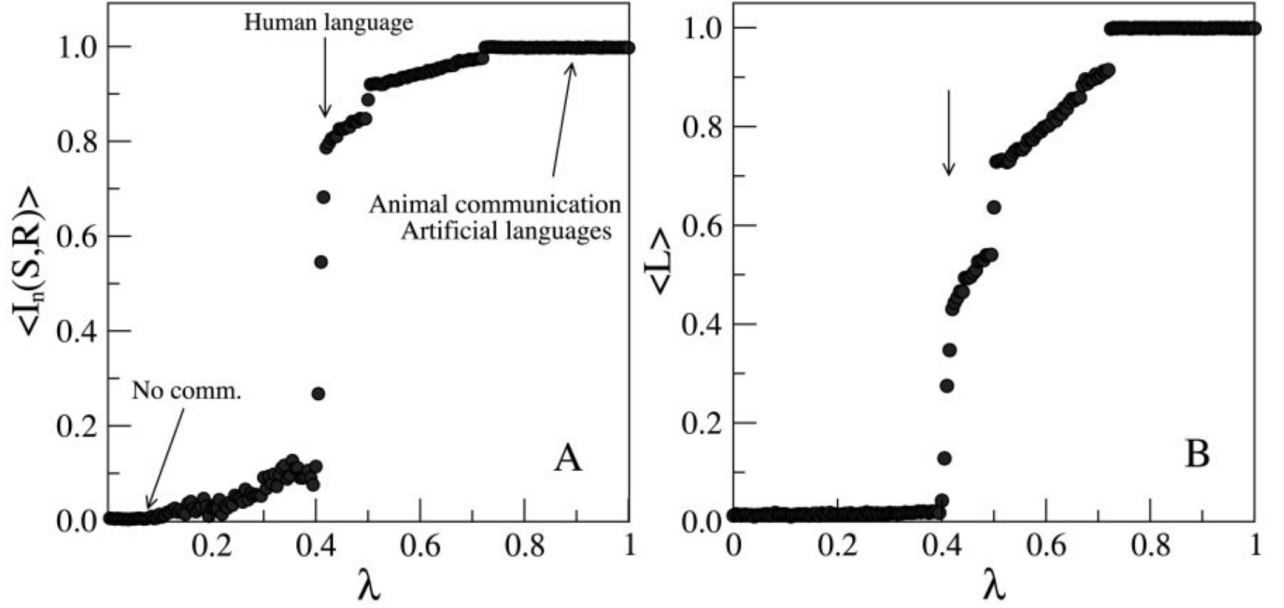


FIG. 1: Phase transition in the mutual information $\langle I_n(S,R) \rangle$ and lexical size $\langle L \rangle$ of simulated languages as a function of the proportion of effort, i.e., bias (λ), devoted to listener interests as opposed to speaker interests. Reproduced with permission from Ferrer i Cancho and Solé (2003).

between states [$x < 0$, $f(x) = 0$; $x > 0$, $f(x) = 1$] is typical of electrical circuits [14] and neural systems [15]. Indeed, prior studies performed analytical derivations of global minima from equation (9) to prove that this theoretical phase transition is well modeled by a step function [11, 16]. These studies demonstrated that the domain $\lambda < \lambda^*$ is characterized by single-signal systems (i.e., one signal refers to all objects), the domain $\lambda = \lambda^*$ is characterized by non-synonymous systems (i.e., no two signals refer to the same object, although one signal may refer to multiple objects), and the domain $\lambda > \lambda^*$ is characterized by one-to-one mappings between signals and objects.

In mathematics, a transform is a method used to convert an equation in one variable to an equation in a different variable [17]. Integrals are a common type of transform and have the generalized form:

$$T[f(x)] = F(z) = \int_a^b f(x)g(x,z) dx \quad (3)$$

where $f(x)$ is the function being transformed, T is the generalized mathematical transform, and $g(x,z)$ is the kernel of the transform. When the definite integral is evaluated, the variable x drops out of the equation and one is left with a function purely of z . For example, in a Laplace transform [14], the kernel is the negative exponential e^{-xz} , which serves as a damping function. In the special case that $f(x)$ is the unit step function (Fig. 2A), the Laplace transform simply yields $1/z$. For example, in electrical engineering, the Laplace transform is often used to map the behavior of functions in the time domain, $f(t)$, to the frequency domain, $F(z)$.

RESULTS

We propose a new integral transform called the Slavi transform, \mathcal{S} , to map communicative bias functions to corresponding word frequencies. Consider the function to transform as $N(\lambda)$: the lexical size $\langle L \rangle$ of a language (i.e., the number of words in the language that are connected and have non-zero probability) as a function of the bias, λ , imparted to the listener over the speaker (Fig. 1B). Because the lexicon size and word-meaning mappings abruptly change at the phase transition near λ^* (Fig. 1A,B), we can substitute the unit step function (Fig. 2A) for $N(\lambda)$:

$$\begin{aligned} \mathcal{S}[N(\lambda)] &= \int_0^1 N(\lambda)e^{-\lambda r} d\lambda = \int_0^x N(\lambda)e^{-\lambda r} d\lambda + \int_x^1 N(\lambda)e^{-\lambda r} d\lambda = \\ &= \int_0^x (0)e^{-\lambda r} d\lambda + \int_x^1 (1)e^{-\lambda r} d\lambda = \frac{1}{r}(e^{-xr} - e^{-r}) = N(r,x) \end{aligned} \quad (4)$$

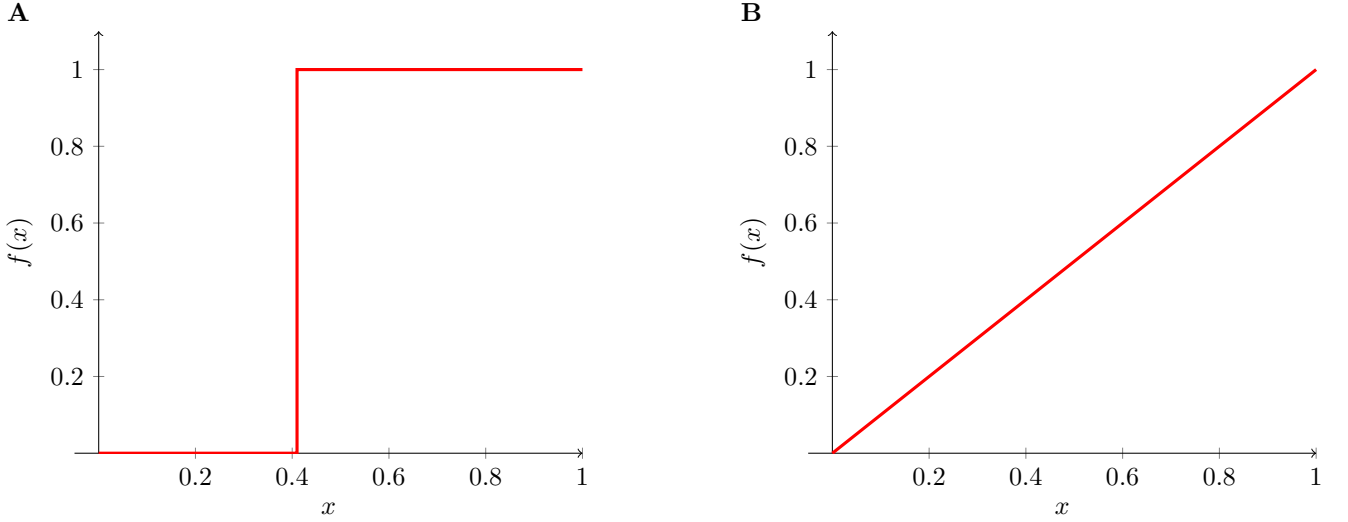


FIG. 2: (A) The unit step (Heaviside function) with phase transition at $\lambda = 0.41$. (B) The unit ramp function on domain $[0, 1]$.

where $x \in [0, 1]$ represents the phase transition near λ^* . Kernels of integral transforms of this form are called Slavi kernels, $e^{-\lambda r}$. Since $e^{-x} - e^{-r} < 1$ for all $x \in [0, 1]$, it follows that:

$$N(r, x) < \frac{1}{r} \quad (5)$$

We'll see the importance of this result in Equation 7. For now, we emphasize four key points to give some intuition behind the utility of the proposed kernel ($e^{-\lambda r}$) used during the mapping:

- Lexical size of a language has been transformed from a function of bias (λ) to a function of rank (r).
- $\lambda > \lambda^*$ (one-to-one mappings between signals and objects) corresponds to high r (rare, unique signals specific to one object). $\lambda < \lambda^*$ (single-signal systems where one signal refers to all objects) corresponds to low r (frequent, repetitive words referring to multiple objects).
- The y-axis is preserved under the transformation: it is still the number of words in the language (i.e., frequency).
- Applying dimensional analysis validates the prerequisite of dimensionless products, as the product $-\lambda r$ is dimensionless since λ is a constant in the range $[0, 1]$ (Fig. 1) and r is a rank ($r \in \mathbb{N}$) corresponding to a specific word (i.e., signal) in the lexicon.

Investigating the other boundary (Fig. 2B) by substituting the unit ramp function for $N(\lambda)$ and performing the Slavi transform yields $1/r^2$ for $r \rightarrow \infty$, a hallmark of complex languages possessing many words (where high r corresponds to rare words in the lexicon):

$$\mathcal{S}[N(\lambda)] = \int_0^1 N(\lambda) e^{-\lambda r} d\lambda = \int_0^1 (\lambda) e^{-\lambda r} d\lambda = -\frac{2}{re^r} + \frac{1}{r^2} = N(r) \quad (6)$$

Thus, depending on how abrupt the phase transition is, one should expect most words in a complex language to scale within the range:

$$\frac{1}{r^2} \leq N(r) \leq \frac{1}{r} \quad (7)$$

or, in terms of the Zipfian exponent, $1 \leq \alpha \leq 2$, which is typically found to be the case [5, 18]. Taken together, there is a connection between the rank of the r^{th} word and its frequency in the lexicon, $N(r, x)$, provided the language is organizing around a phase transition in mutual information and lexicon size.

SIMULATION INTRODUCTION

We want to further demonstrate the importance of the slavi transform mapping the lexical size of a language, originally a function of bias, to a function of rank by introducing a reinforcement learning simulation. Our goal for this simulation is to introduce a realistic scenario where we can show that word rank has a bigger and more clear impact than bias on communication performance. Higher word rank corresponds to more infrequent and unique words, while lower word rank corresponds to more frequent words and synonyms. To better understand the inspiration behind the simulation, consider the following scenario. You are walking with your friend in a very busy park. There are numerous objects that you and your friend see, but a brown colored dog captures your attention and you want to communicate this to your friend verbally. In this case, you are the "speaker" and your friend is the "listener". You, the speaker, generate a "label", such as "brown dog", to alert the listener's attention to the dog in question. "Brown dog", is a very specific label and therefore it makes it very easy for the listener to associate it with the specific dog the speaker intends to communicate. The phrase, "brown dog", is exclusively for a dog whose color is brown, which means that the phrase is mapped to the brown colored dog or similar objects like it. In this scenario, the speaker is exerting more effort than the listener, who can easily map "brown dog" to the brown colored dog. If the speaker instead just says "brown object", the listener will have to exert more effort to figure out that the speaker is talking about the dog. This would mean that the listener is exerting more effort than the speaker. Recall from (2) that this effort measurement is represented as bias in Ferrer i Cancho's energy function:

$$\Omega(\lambda) = \lambda I(S, R) - (1 - \lambda) H(S) \quad (8)$$

Bias, λ , controls the balance between the speaker interests, $H(S)$, and listener interests, $I(S, R)$. It can be viewed as a learning rate, which will help with the formulation of the training model later on. It is also important to note that even if the listener initially has no idea that the "brown dog" means brown colored dog, through repetition it would learn, in order to communicate effectively with the speaker. This can be viewed as natural language emergence, where labels are chosen, or "generated", and repeated until the listener and speaker reach a consensus. Our main goal is to simulate this natural language emergence and compare the effects of bias and word rank on the ability to communicate efficiently between a speaker and listener, ultimately leading to evidence that points to word rank as a better alternative to bias.

DATA

We chose to a set of 10 unique image classes, provided by the CIFAR10 dataset. The image classes are "airplane", "automobile", "bird", "cat", "deer", "dog", "frog", "horse", "ship", and "truck".

MODEL STRUCTURE

We attempt to model the scenario introduced above with a two player game between two reinforcement learning agents, a speaker and a listener, where images are the objects that are to be communicated. The speaker and listener start off as almost independent, not communicating successfully. Through a training process they will create a language, or a mapping, between objects and one hot encoded values to communicate effectively. We adapted the model introduced by Angeliki Lazaridou, Alexander Peysakhovich, Marco Baroni [19].

The game between the speaker, parameterized as θ_s , and listener, parameterized as θ_l , is as follows:

1. A sample image from each of the n unique classes from an image dataset is drawn and passed through a pretrained VGG19 network [20], the output represented by image vectors $\{i_0, \dots, i_{n-1}\}$. One of the vectors is chosen to be the "target image", represented as $i_t \in \{i_0, \dots, i_{n-1}\}$, where $t \in \{0, \dots, n-1\}$.
2. The speaker takes as input the target image i_t and generates a label from a vocabulary of size m , where $m > 1$. The label is represented as a one hot encoded vector of size m . This action is the speaker's policy, $\pi_{\theta_s}(i_t, m)$.
3. The listener takes in each image vector, $\{i_0, \dots, i_{n-1}\}$, and the action label, $\pi_{\theta_s}(i_t, m)$, generated by the speaker. It tries to guess which image the speaker saw by matching the label generated by the speaker to the correct target image i_t . This guess is the listener's policy, $\pi_{\theta_l}(\{i_0, \dots, i_{n-1}\}, \pi_{\theta_s}(i_t, m))$.
4. If the listener guesses the target correctly, or $\pi_{\theta_l}(\{i_0, \dots, i_{n-1}\}, \pi_{\theta_s}(i_t, m)) = i_t$, then both the speaker and listener receive a reward of 1. If the listener gets it wrong, the speaker and listener receive a reward of 0.

5. update parameters θ_s and θ_l .

Over time, the listener and the speaker will develop a mapping to communicate the target images.

TRAINING AND TESTING

We chose the speaker and listener to be reinforcement learning agents because the way they learn to communicate effectively is similar to how humans would learn: through repetition and based on whether the speaker and listener reached a consensus. The update rule we chose to optimize the speaker's and listener's parameters is based off the Monte Carlo Policy Gradient (REINFORCE) algorithm [22]:

Algorithm 1 Monte Carlo Policy Gradient (REINFORCE) algorithm

```

procedure REINFORCE
  initialize parameters  $\theta$  arbitrarily
  for each episode  $\{s_0, a_0, r_0, \dots, s_T, a_T, r_T\} \sim \pi_\theta$  do
    for  $t = 0$  to  $T$  do
      generate long term value  $v_t$  from function  $Q^\pi(s, a)$ 
       $\theta \leftarrow \theta + \alpha \nabla_\theta \log \pi_\theta(s_t, a_t) v_t$ 
    end for
  end for
  return  $\theta$ 
end procedure

```

We chose the long term reward v_t for our algorithm to simply be r_t , because trials are independent of each other. We also chose to modify the update rule by incorporating bias, $0 < \lambda < 1$. Returning to Ferrer i Cancho's energy function:

$$\Omega(\lambda) = \lambda I(S, R) - (1 - \lambda) H(S) \quad (9)$$

We can view bias, or λ , as "learning rate", measuring the importance of the speaker's and listener's performances, and updating the model accordingly. This lets us easily incorporate λ in our simulation. $(1 - \lambda)$ scales speaker interests, $H(S)$, and λ scales listener interests, $I(S, R)$, so we scale our normal learning rate by $(1 - \lambda)$ for the speaker's update and λ for the listener's update and formulate a modified Monte Carlo Policy Gradient (REINFORCE) algorithm:

Algorithm 2 Modified Monte Carlo Policy Gradient (REINFORCE) algorithm

```

procedure MODIFIED REINFORCE
   $s_t$  = speaker state at time step  $t$ 
   $l_t$  = listener state at time step  $t$ 
  initialize speaker parameters  $\theta_s$  and listener parameters  $\theta_l$  arbitrarily.
  for each episode  $\{s_0, l_0, \pi_{\theta_s}(s_0), \pi_{\theta_l}(l_0, \pi_{\theta_s}(s_0)), r_0, \dots, s_T, l_T, \pi_{\theta_s}(s_T), \pi_{\theta_l}(l_T, \pi_{\theta_s}(s_T)), r_T\}$  do
    for  $t = 0$  to  $T$  do
      update speaker parameters:  $\theta_s \leftarrow \theta_s + (\alpha \times (1 - \lambda)) \nabla_{\theta_s} \log \pi_{\theta_s}(s_t) r_t$ 
      update listener parameters:  $\theta_l \leftarrow \theta_l + (\alpha \times \lambda) \nabla_{\theta_l} \log \pi_{\theta_l}(l_t, \pi_{\theta_s}(s_t)) r_t$ 
    end for
  end for
  return  $\theta_s, \theta_l$ 
end procedure

```

AGENT ARCHITECTURES

Both the speaker and listener are feedforward neural networks, implemented in Keras. The neural networks' weights are initialized using Glorot Initialization [21]. The speaker passes the target image through a pretrained VGG19 to generate an image vector, and passes the output through the network with a softmax output activation to generate probabilities for each label in a vocabulary of size m . The listener passes a sample image from each class, including the target image, through a pretrained VGG19 to generate image vectors, which are passed through a shared linear layer with a relu activation function to create image embeddings. It passes the label generated by the speaker through a linear layer and computes dot products with the image embeddings. The dot products are then passed through a softmax function to generate probabilities.

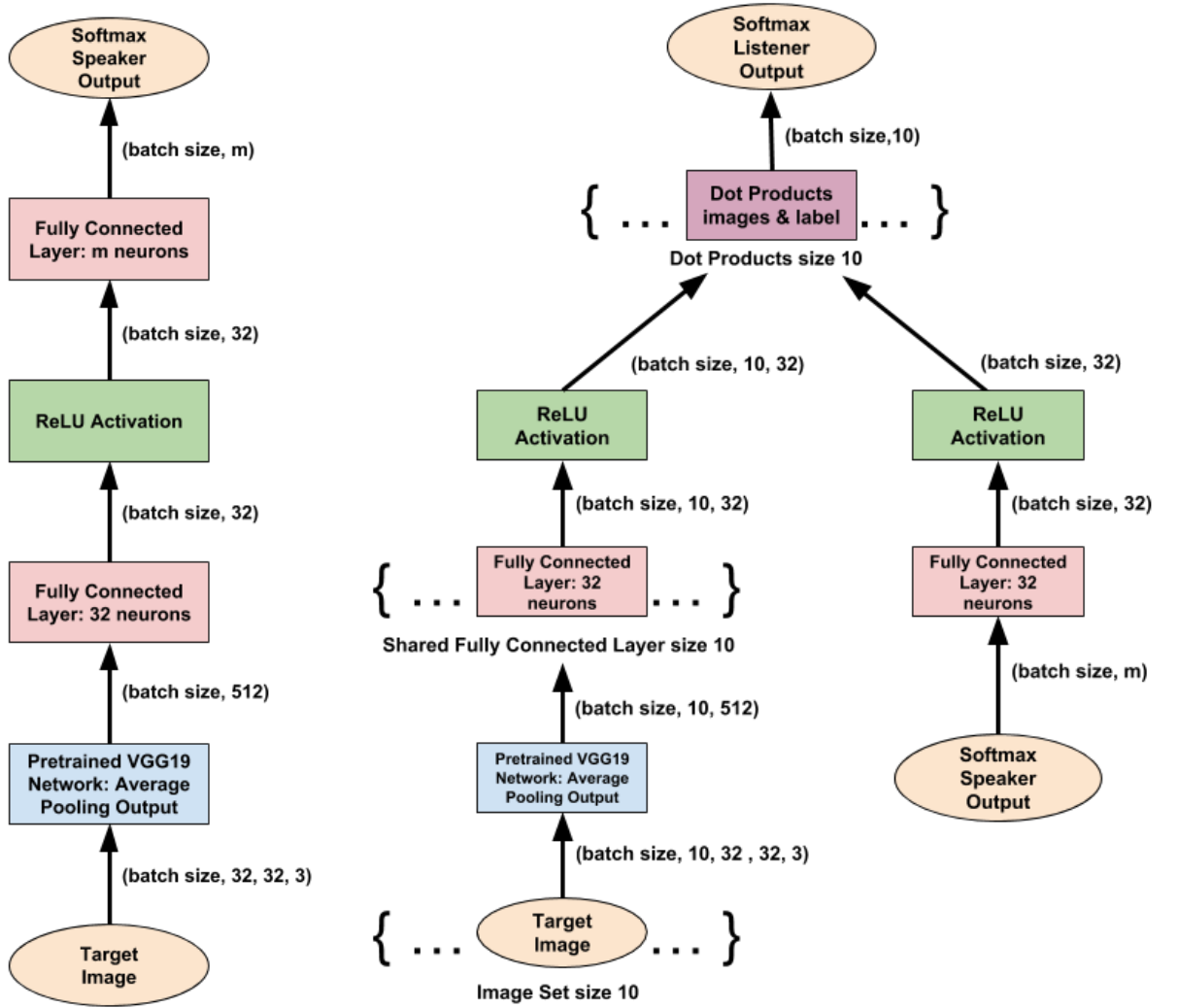


FIG. 3: (Left) Speaker model architecture: The speaker takes in a random batch of target images. After the images are passed through a pretrained VGG19 network, they are passed through a 32 neuron fully connected layer with ReLU activation to generate a target image embedding. The embedding is passed through a fully connected softmax output layer with the number of neurons equal to the vocabulary size m . (Right) Listener model architecture: The listener takes in two inputs: A batch of samples of each image class, including the target image, and the softmax speaker output of size m . The images are passed through a pretrained VGG19 network and then one shared 32 neuron fully connected layer to create image embeddings. The softmax speaker output is passed through a separate 32 neuron fully connected layer with ReLU activation to create a label embedding. The dot products between the label embedding and the image embeddings are computed and passed through a softmax activation.

SIMULATION RESULTS

To simulate word rank, we simply modify the vocabulary size m : smaller vocabulary corresponds to more frequent labels for multiple objects, resulting in lower word rank, while larger vocabulary corresponds to more one to one mappings between labels and objects, resulting in higher word rank. This is very intuitive to how humans interact, if there are less words in a speaker's vocabulary than objects, the speaker will be forced to use the same word more frequently to mean different objects, making it harder to communicate. We trained for 1000 episodes, with 100 samples from each image class. We used a learning rate α of 0.001.

Reiterating from the introduction, our main goal of the simulation is to provide evidence that word rank is a better and more intuitive alternative to bias when it comes to effective communication. To do this, we observed the effect on the performance of our model while changing vocabulary size m and keeping bias λ constant compared to the performance while changing bias λ and keeping vocabulary size m constant. Performance is measured as accuracy, or total reward / total number of trials, which means the proportion of trials where the receiver picked the right object. We then determine whether there is a stronger trend between accuracy and rank compared to accuracy

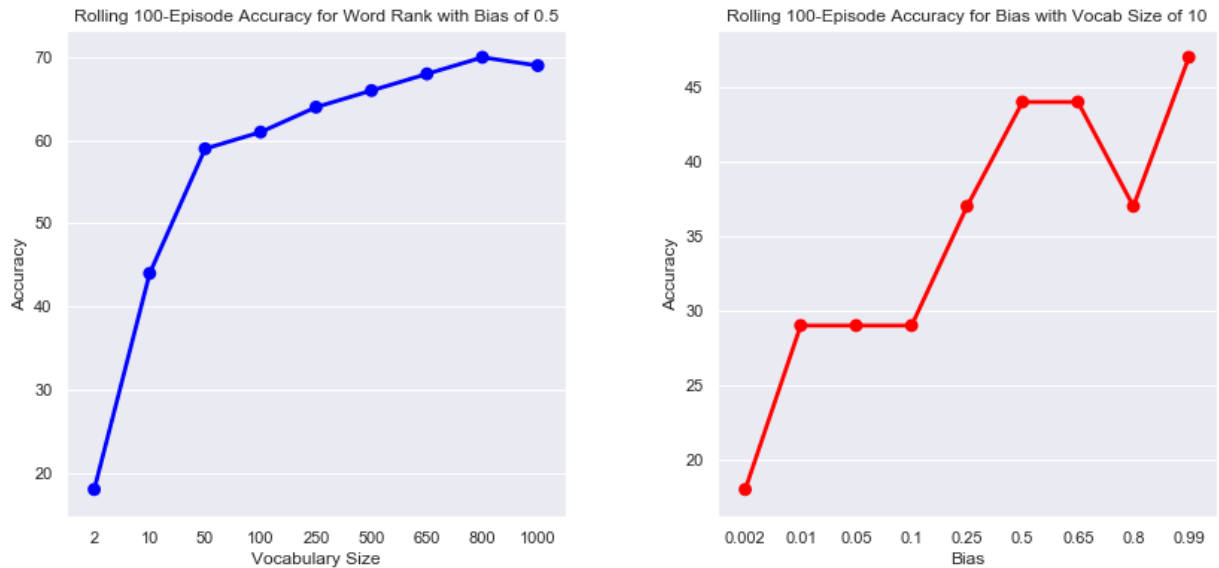


FIG. 4: (Left) Rolling average accuracy of 100 most recent episodes, over a training period of 1000, of models with different vocabulary sizes and a constant bias (λ) of 0.5. (Right) Rolling average accuracy of 100 most recent episodes, over a training period of 1000, of models with different bias (λ) values and a constant vocabulary size of 10.

and bias. To test word rank, we ran the simulation with vocabulary sizes m of 2,10,50,100,250,500,650,800,1000, with a constant λ of 0.5 (equal effort between speaker and listener). When testing for bias, we used λ values of 0.002,0.01,0.05,0.1,0.25,0.5,0.65,0.8,0.99, and a constant vocabulary size m of 10. The rolling average of the 100 most recent episodes for both the simulations are shown in fig 4. The accuracy values for rank and bias are shown in Table 1 and Table 2, respectively.

TABLE I: Rolling Accuracy 100 episodes Word Rank

Vocabulary Size	Accuracy
2	18%
10	44%
50	59%
100	61%
250	64%
500	66%
650	68%
800	70%
1000	69%

TABLE II: Rolling Accuracy 100 episodes Bias

Bias (λ)	Accuracy
0.002	18%
0.01	29%
0.05	29%
0.1	37%
0.25	44%
0.5	44%
0.65	37%
0.8	37%
0.99	47%

From the accuracy measurements, we can see that both bias (λ) and word rank (vocabulary size m) have a positive relationship with accuracy. However, it is clear that bias has a weaker relationship than word rank. For vocabulary sizes $m \leq 800$, there seemed to be a consistent increase in model performance, reaching a peak of 70%, and from $m > 800$ the accuracy seemed to level off at around 69%. Bias seemed to have a weaker relationship: at $0.001 \leq \lambda \leq 0.05$ and $0.25 \leq \lambda \leq 0.5$ the performance of the model didn't improve. Furthermore, at $0.65 \leq \lambda \leq 0.8$, the performance actually dipped from 44% to 37%, but eventually rose to 47%. Word rank is not only more intuitive to understand (vocabulary size is much easier to understand than bias), but also has a stronger positive relationship with accuracy, implying that it has a stronger effect, compared to bias, on the communicative performance between a speaker and listener.

SHORTCOMINGS OF THE SIMULATION

The peak accuracy we reached was 70%, so there definitely is room for improvement. The model architecture can be expanded upon to improve the performance, one idea could be to include 1d convolutional layers, which may improve accuracy. In addition to model improvements, with more computation power, more image samples and training episodes can be played. It would also be interesting if CIFAR10 could be substituted with the CIFAR100 dataset, with more objects to communicate the problem would be more complex.

CONCLUSIONS

We show that the Slavi transform maps communicative functions of speaker-listener bias directly to ranks. Specifically, we demonstrate that the lexical size of a language can be mapped from a function of bias, $N(\lambda)$, to a function of rank at any arbitrary phase transition point, $N(r, x)$. We also provide evidence supporting the choice of ranks over bias in the form of a realistic scenario, modelled with a reinforcement learning game between a speaker and a listener, where ranks had a much stronger effect than bias on the communicative performance. This study highlights the importance of integral transform theory to understanding information-theoretic models of communicative systems in the context of Zipfian ranks.

Code accompanying this paper can be found here: <https://github.com/Quiltomics/NLERL>

* Electronic address: bohdan@stanford.edu, shyamsnair@protonmail.com

- [1] Zipf G (1949), *Human Behaviour and the Principle of Least Effort: An Introduction to Human Ecology* (Addison-Wesley, Cambridge, MA).
- [2] Ferrer i Cancho R (2005), Zipf's law from a communicative phase transition. *European Physical Journal B* 47: 449–457.
- [3] Baixeries J, Elvevåg B, Ferrer i Cancho R (2013), The evolution of the exponent of Zipf's law in language ontogeny. *PLoS One* 8(3): e53227.
- [4] Alday PM (2016), Towards A Rigorous Motivation For Zipf's Law. In S.G. Roberts, C. Cuskley, L. McCrohon, L. Barceló-Coblijn, O. Fehér & T. Verhoef (eds.) *The Evolution of Language: Proceedings of the 11th International Conference (EVLANG11)*.
- [5] Moreno-Sánchez I, Font-Clos F, Corral Á (2016), Large-Scale Analysis of Zipf's Law in English Texts. *PLoS One* 11(1): e0147073.
- [6] Mandelbrot B (1966), Information theory and psycholinguistics: a theory of word frequencies. In Lazafeld P, Henry N, eds.: *Readings in Mathematical Social Science* (MIT Press, Cambridge, USA).
- [7] Mandelbrot B (1983), *The Fractal Structure of Nature* (Freeman, New York).
- [8] Montemurro MA (2001), Beyond the Zipf-Mandelbrot law in quantitative linguistics. *Physica A* 300: 567–578.
- [9] Manning C, Schütze H (1999), *Foundations of Statistical Natural Language Processing* (MIT Press, Cambridge, MA).
- [10] Ferrer i Cancho R, Solé RV (2003), Least effort and the origins of scaling in human language. *Proc Natl Acad Sci USA* 100: 788–791.
- [11] Ferrer i Cancho R, Díaz-Guilera A (2007), The global minima of the communicative energy of natural communication systems. *Journal of Statistical Mechanics*: P06009.
- [12] Ferrer i Cancho R, Elvevåg B (2010), Random texts do not exhibit the real Zipf's law-like rank distribution. *PLoS One* 5(3): e9411.
- [13] Shannon C (1948), A Mathematical Theory of Communication. *Bell System Tech J* 27: 379–423 & 623–656.
- [14] Spiegel MR (1965), *Laplace Transforms* (McGraw-Hill Book Company, New York).
- [15] McCulloch W, Pitts W (1943), A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics* 7: 115–133.
- [16] Prokopenko M, Ay N, Obst O, Polani D (2010), Phase transitions in least-effort communications. *Journal of Statistical Mechanics: Theory and Experiment*. 11: P11025.
- [17] Körner TW (2008), *Transforms*. In Gowers, T., ed.: *The Princeton Companion to Mathematics* (Princeton University Press, Princeton): 303–307.
- [18] Ferrer i Cancho R (2006), When language breaks into pieces: A conflict between communication through isolated signals and language. *BioSystems* 84(3): 242–253.
- [19] Angeliki Lazaridou, Alexander Peysakhovich, Marco Baroni (2016), Multi-Agent Cooperation and the Emergence of (Natural) Language. *CoRR*: abs/1612.07182
- [20] K. Simonyan, A. Zisserman (2014), Very Deep Convolutional Networks for Large-Scale Image Recognition. *ArXiv e-prints*: 1409.1556
- [21] Xavier Glorot, Yoshua Bengio (2010) Understanding the difficulty of training deep feedforward neural networks. *Proceedings of Machine Learning Research*, 9:249–256
- [22] Ronald J Williams (1992) Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229-256