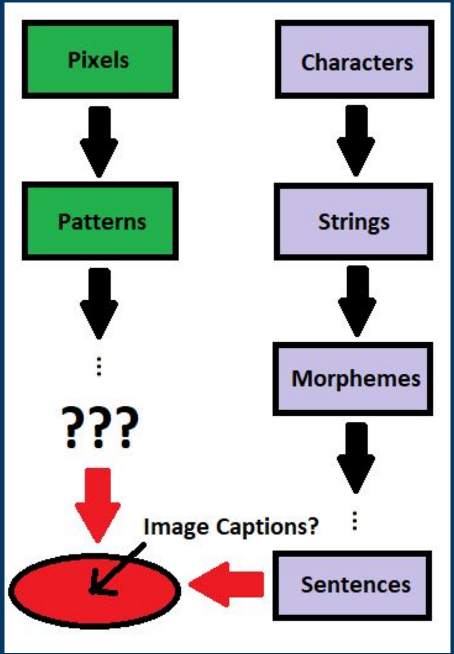
# A Computational Theory For Life-Long Learning of Semantics

Peter Sutor, Douglas Summers-Stay, Yiannis Aloimonos

Peter Sutor
University of Maryland - Dept. of CS
8/22/18
AGI 2018

# Life-Long Learning of Semantic Information

- Ongoing
  - Definite starting point.
  - Non-static dataset.
  - Online.
- Incremental
  - Learning simple concepts first, then more complicated ones.
  - New concepts or **even types of data** can appear at any moment.
  - New information absorbed dynamically.
- Semantics → Similarity between concepts
  - Interpretability?
  - Modification?
  - Semantic space of ALL things?



# Semantic Vector Learning

- Learning good vector representations for concepts.
- Embedded in a continuous, real space.
- Distance related to similarity.
- High dimensionality $\rightarrow$ more expressiveness
- Operations on vectors are meaningful.

# Hyperdimensional Binary Computing [Kanerva]

- Consider 10,000 bit long binary vectors:
  - Space contains $2^{10,000}$ unique vectors.
  - Every point has the same distribution of distances to each other point.
  - Average random Hamming Distance is 5,000/10,000 = 0.5.
  - Binary distribution, mean distance 5,000 and STD 50.
  - Resistant to random noise.
  - Compatible with probability!
- Interesting operations:
  - XOR: Is an involution ($c = a \oplus b \rightarrow a$ recoverable given b, and vice versa)
  - Permutation Π: Repeated permutation generates random distances (close to 0.5)
  - "Consensus Sum"
- Mapping with XOR and Permutation preserves distance:
  - $H(a \oplus x, a \oplus y) = | x \oplus y |$
  - $H(\Pi x, \Pi y) = | x \oplus y |$

# Representing Data Structures [Kanerva]

- **Sets:** For $\{\zeta_1, \zeta_2, \ldots, \zeta_m\} \mapsto \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_m\}$, where $\mathbf{z}_i$ are binary vectors:
  - Represent as XOR of $\mathbf{z}_i$, or $\mathbf{z} = \mathbf{z}_1 \oplus \mathbf{z}_2 \oplus \ldots \oplus \mathbf{z}_m$
  - Union and intersection computable.
- **Ordered Pairs:** $\zeta_r = (\zeta_s, \zeta_t)$ then the corresponding vectors are $r = \Pi s \oplus t$
- **Sequences:** Recursive ordered pairs!

$$\zeta_z = \zeta_{z_1}\zeta_{z_2}\ldots = (\zeta_{z_1}\ldots\zeta_{z_{m-1}}, \zeta_{z_m})$$

Encoding $\longrightarrow$

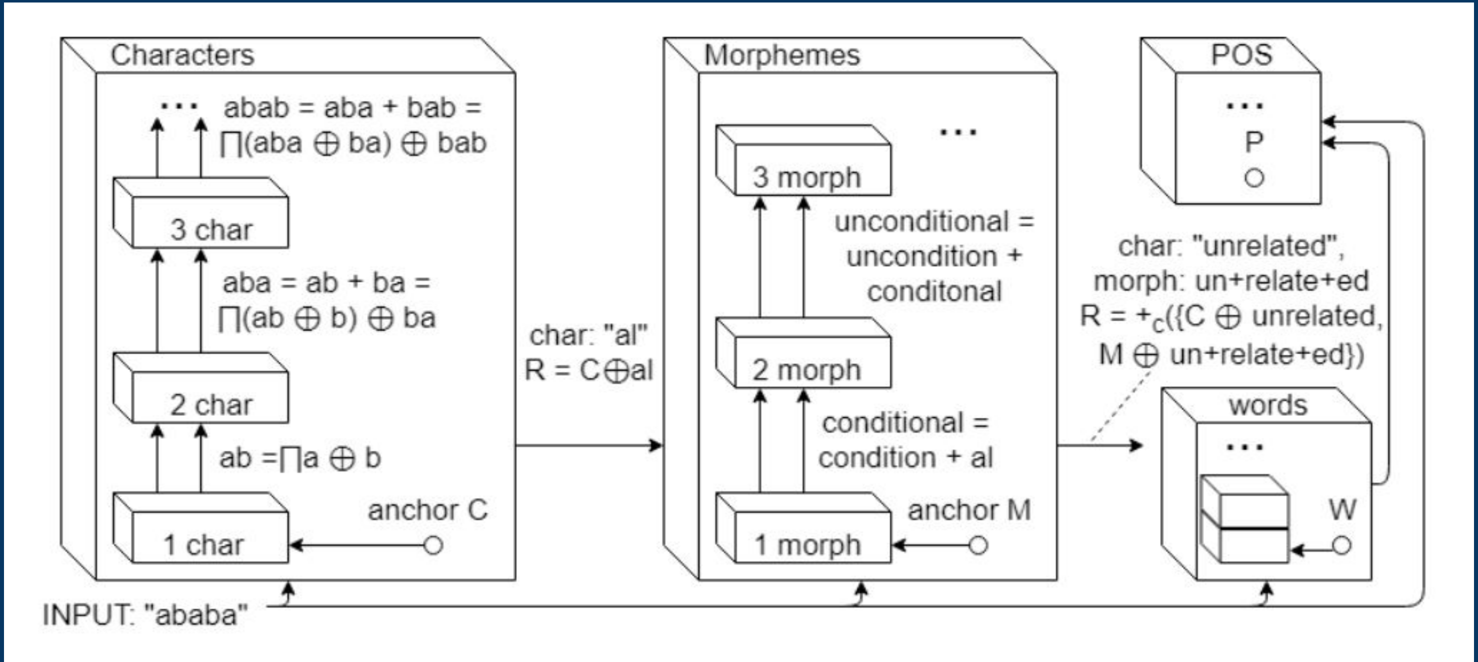
$$z = \Pi^{m-1} z_1 \oplus \Pi^{m-2} z_2 \oplus \ldots \oplus \Pi^{m-i} z_i \oplus \ldots \oplus \Pi z_{m-1} \oplus z_m$$

- **Records:** Bind XOR, sum by consensus.
  - Example: [Name, Gender, Age][Peter, Male, 26] = [Name: Peter, Gender: Male, Age: 26]
  - $$\begin{aligned} R_v &= [r_1 r_2 \ldots r_m][v_1 v_2 \ldots v_m]^T \\ &= r_1 \oplus v_1 + r_2 \oplus v_2 + \ldots + r_m \oplus v_m = +_c(\{r_i \oplus v_i\}) \end{aligned}$$

# Incrementally Learned Knowledge

- Represent all information as long binary vectors. Encode more complicated information by combining binary vectors.
  - Basic building blocks assigned random starting vectors or geometrically sensible vectors.
  - As permutation $\Pi$ and XOR $\oplus$ preserve distance, transform existing representations to new encodings with these operations.
  - A sequence of data can be encoded as shown before.
  - Final point in binary space determined by a data record of all known information.
  - Structure of this is like a “knowledge graph” of semantic relationships.
- Where do we learn the semantics?
  - Each category of data exists in its own space with an identifier (anchor).
  - Position in this space subject to change.
  - Referencing this data externally requires first encoding by anchor.
  - Can use existing models to enrich semantic information.

# Incrementally Learned Knowledge (Linguistic Example)



# Geometric Interpretation of Semantics

- "Things that co-occur should be closer to each other"
  - Envision semantics as a spring-mass system.
  - The more often data is seen, the more "mass" it has.
  - The more often a relationship occurs, the closer the related components want to be.
- ***Connective Force:***
  - The pulling force generated through relationships between two semantic points.
  - Much like springs attached to masses.
- ***Proximal Force:***
  - The pushing force resisting two semantic points from getting closer.
  - Basically, reverse gravity.
- Want to reach a low energy state across whole system.

# Binary Vector Analogue to Geometric Semantics

- Given a knowledge graph K of m vertexes, we want to minimize for X, where X is m by n: $$\arg\min_{X}(T(X^{(k)} + X))$$
- Function T is the ***total tension*** for K in a given vector state of each vertex:

$$T(A) = \sum_{i=1}^{m}\sum_{j=1}^{n}\max(F_{conn}(A,i,j) + F_{prox}(A,i,j), 0)$$

$$F_{prox}(A,i,j) = \sum_{k=1,k\neq i}^{m} \frac{M_i M_k}{H(A_i, A_k)^2} C_{prox}(A_{ij}, A_{kj})$$

Proximal Force

$$F_{conn}(A,i,j) = \sum_{k=1,k\neq i}^{m} M_i W_{ik} C_{conn}(A_{ij}, A_{kj})$$

Connective Force

# Binary Vector Analogue (Continued)

- The C functions for Connective and Proximal Force determine the "direction" of the force on a bit.

$$C_{prox}(a,b) = \begin{Bmatrix} 1, \text{ if } a = b \\ -1, \text{ if } a \neq b \end{Bmatrix} \quad C_{conn}(a,b) = \begin{Bmatrix} -1, \text{ if } a = b \\ 1, \text{ if } a \neq b \end{Bmatrix}$$

- By substitution, the total force experienced by a bit of a particular vector:

$$\begin{aligned} F &= \sum_k M_i W_{ik} C_{conn}(A_{ij}, A_{kj}) + \sum_k \frac{M_i M_k}{H_N(A_i, A_k)^2} C_{prox}(A_{ij}, A_{kj}) \\ &= \sum_{k=1, k\neq i}^{m} M_i C_{conn}(A_{ij}, A_{kj}) \left[ W_{ik} - \frac{M_k}{H(A_i, A_k)^2} \right] \end{aligned}$$

# Minimization Across Binary Vectors

- Overall formulation for total tension:

$$T(A) = \sum_{i=1}^{m}\sum_{j=1}^{n} \max\left(\sum_k M_i C_{conn}(A_{ij}, A_{kj})\left[W_{ik} - \frac{M_k}{H(A_i, A_k)^2}\right], 0\right)$$

- Fast initial minimization of new vectors:
    - Can compute pseudo-gradients on each bit for each vector by seeing how much energy changes by flipping it.
    - Many Body Problem for Hamming Distance (consider only connected components)
    - Change as many bits of the most high energy vector that satisfy a dynamically decaying threshold before computing the effect on total tension.
    - Very similar process to simulated annealing.

# Example Total Energy Minimization

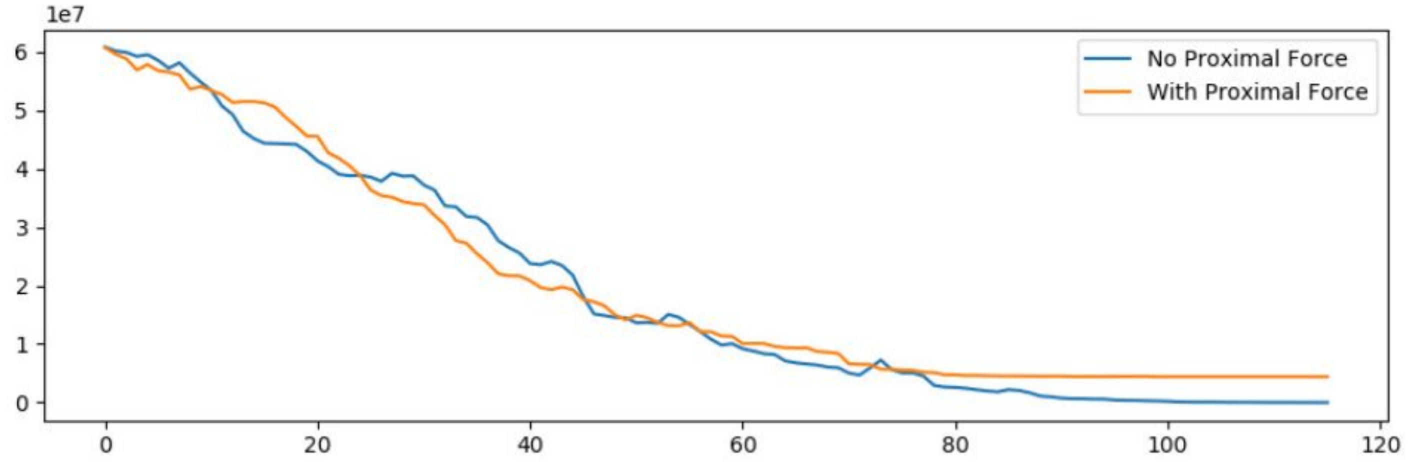
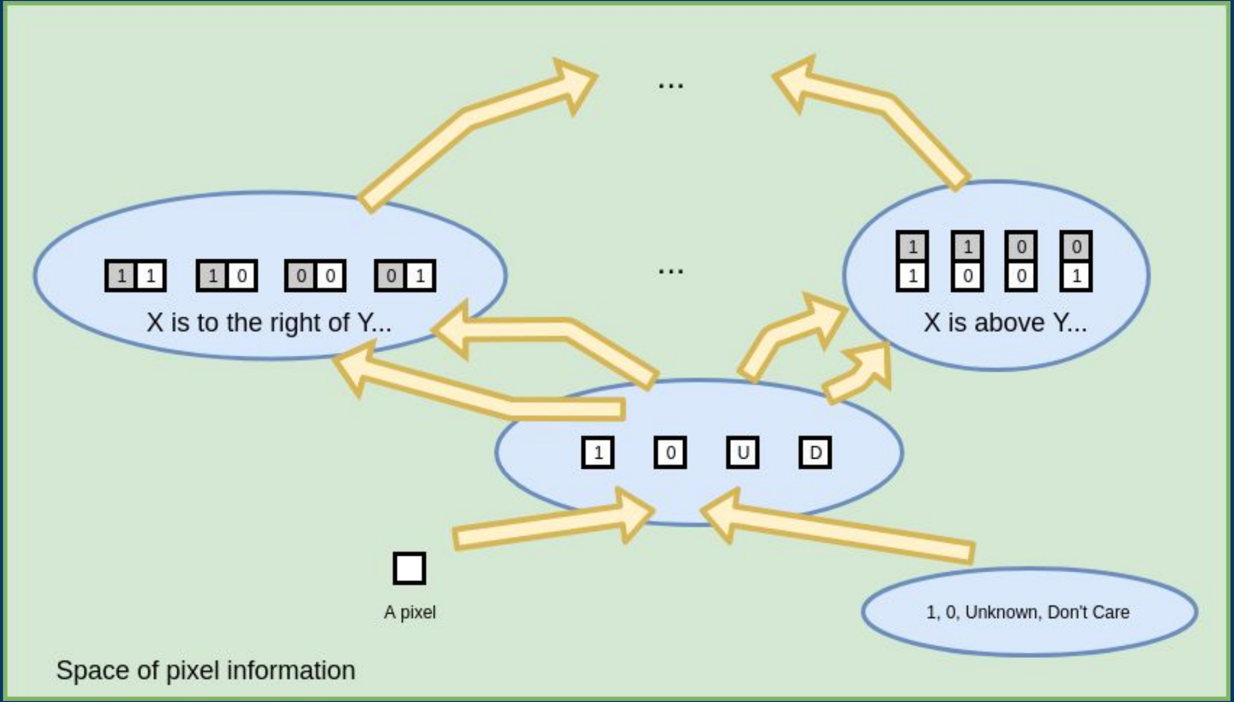



**Fig. 1.** Example minimization per random row of a randomly connected 50 node graph's binary vectors via the greedy method. Without proximal force it reaches 0.

# The Life-Long Learning Process

- Either structural composition of knowledge, or enriched by semantics.
  - We can use the geometric interpretation and minimization on statistical observations.
  - Alternatively, can be directed through interaction with its environment.
  - Self testing? Focus testing?
  - Exploration?
    - Use probability to measure likelihood of two unrelated vectors being near each other.
    - Can ask people why this might be? Update knowledge to reflect that.
- Binary vectors as features:
  - Can use whole space.
  - Can use a particular semantic subspace.
  - Self Organizing Maps to learn topology of a space of vectors? [Kohonen]
  - Highest level of abstraction can be a sequence of an entire lifetime of observations.

# Encoding Images




- Values of pixels:
  - May be definite, may be unknown, maybe we don't care.
  - One = 1 - Zero
  - Uknown 1 - Don't care, but orthogonal to one and zero.
- Encoding Location:
  - Right = 1 - Left
  - Up = 1 - Down, but both orthogonal to right and left.
  - Every pixel can be encoded to know what is around them in the entire pattern.
  - First permute repeatedly by one axis, then the other.
  - Can embed patterns into regular shapes.

# Future Work

- Implementing a general framework for the life long learning system.
- Perfecting efficient and powerful structural representations of the knowledge graph for pixel and character based data through empirical testing.
- Integrating multiple data representations into a single system and studying the effect of these on performance.
    - Particularly, does including classical learned models improve the results?
- Self-guided learning apart from supervision:
    - Mini-tests to focus on improving inadequacies.
    - Employ probability to hypothesize new, unsupervised relationships.

# Thank you for your time! Any questions?

MOST PERTINENT REFERENCES:


1. Sutor P., Summers-Stay D., Aloimonos Y. (2018) A Computational Theory for Life-Long Learning of Semantics. In: Iklé M., Franz A., Rzepka R., Goertzel B. (eds) Artificial General Intelligence. AGI 2018. Lecture Notes in Computer Science, vol 10999. Springer, Cham
2. Kanerva, P. (2009). Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors. Cognitive Computation, 1(2), 139-159.
3. Kohonen, T. The self-organizing map. Proceedings of the IEEE 78(9), 1464–1480 (Sep 1990). https://doi.org/10.1109/5.58325