Unifying formulaic, geometric, and algebraic theories of semantics

András Kornai
SZTAKI Computer Science Research Institute

ESSLLI, August 3 2021


ตายอด
คลำช้าง

Outline

Background

- In Lecture 1 we discussed five classes of models. Here we concentrate on word vectors, because they are the body of the elephant (by far the largest volume, with thousands of researchers using continuous vector space semantics).
- Why is this the body? Because word meaning carries the bulk of the information, over 80%.
- Why not discrete vectors? Binary feature vectors work well in phonology and morphology, and theories capturing word meaning in terms of simple structures (trees) built from these have been around at least since Katz and Fodor, 1963.
- Yes, but their learning theory is weak. Using continuous vectors gives us differentiability, differentiability gives us gradient optimization, gradient optimization can be used for learning.

Within ESSLLI

Hope to whet your appetite for two related courses in Week 3:

- Jose Camacho Collados and Mohammad Taher Pilehvar **Embeddings in Natural Language Processing**
- Stefan Evert and Gabriella Lapesa **Hands-on Distributional Semantics**

"Embedding" is just another word for word vectors, standardly defined as a mapping (context-free in the static case, context-sensitive in the dynamic case) of a dictionary to $\mathbb{R}^n$.
"Distributional semantics" is just another word for the key idea for creating the mapping, *You shall know a word by the company it keeps* (John Rupert Firth)

Neural "brain" models

- Long history, with mathematical models going back to McCulloch and Pitts, 1943, Rosenblatt, 1957
- We are more interested in the mathematical side than in actual brain science (Hertz, Krogh, and Palmer, 1991)
- Minsky and Papert, 1988 (3rd ed, originally 1969) unrepentant in their dismissal of neural nets. See Pollack, 1989 for a discussion.
- In the history of ideas, the XOR and parity issues were a real problem, unsolvable by the classic single-layer perceptron.
- Remarkably, the solution, multi-layer NNs and backprop, was found as early as Bryson and Ho (1969), but not fully appreciated until Werbos (1974), Parker (1985), Le Cun (1985), and Rumelhart, Hinton, and Williams, 1985.

Parity

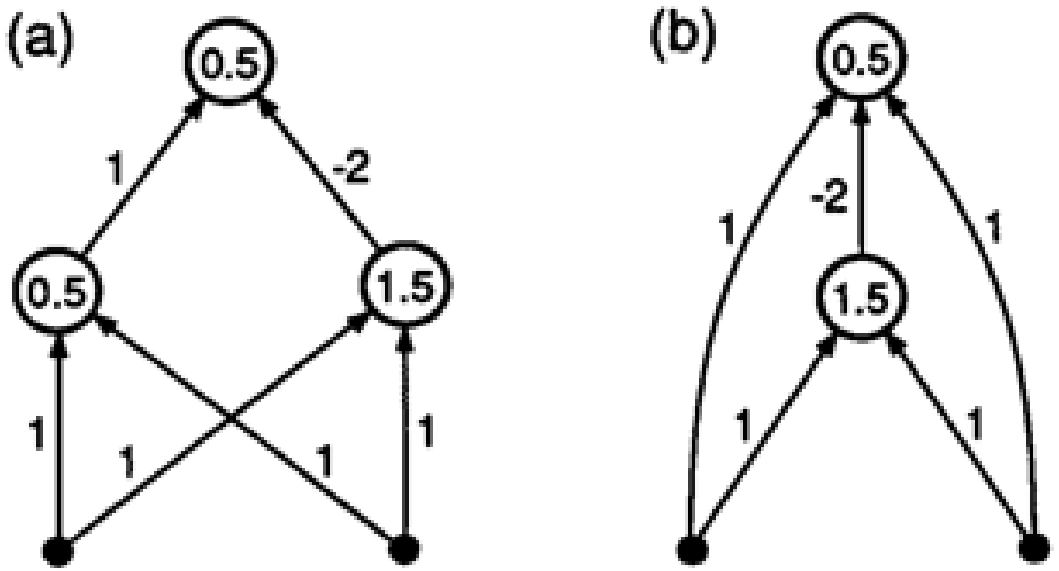


FIGURE 6.5 Two networks that can solve the XOR problem using 0/1 threshold units. Each unit is shown with its threshold.

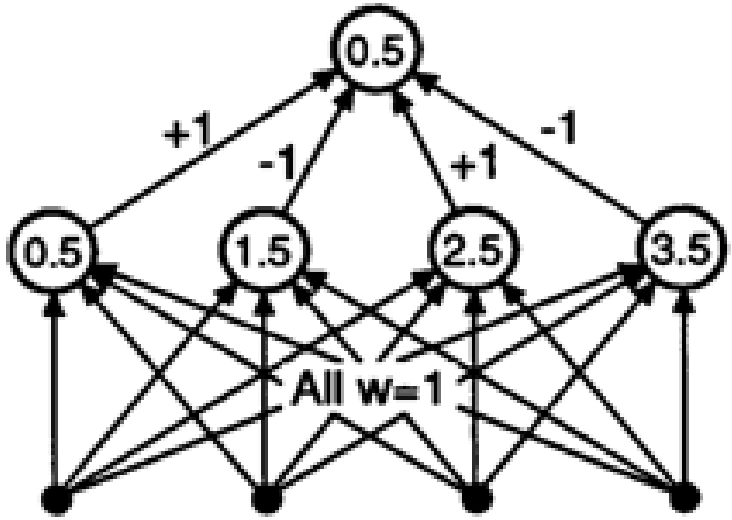


FIGURE 6.6 A network to solve the $N = 4$ parity problem with 0/1 threshold units.

Takeaways from figure

Sutskever et al 2014

large DNNs can be trained with supervised backpropagation whenever the labeled training set has enough information to specify the network's parameters. Thus, if there exists a parameter setting of a large DNN that achieves good results (for example, because humans can solve the task very rapidly), supervised backpropagation will find these parameters and solve the problem.

- Discrete system is embedded in continuous one (Gyenis, 2018)
- It is the continuous aspects that enable gradient learning
- We will start the analysis based on Little, 1974
- Single layer, but recurrent (contains multilayer as special case)
- Bra-ket notation, the computation already done in Ashkin and Lamb (1943), relevant math goes back to early 20th c.
- Please, no "brain haz quantum" amateur philosophy, leave this to the pros, go to Stephen Clark's talk tomorrow!

Setup

- n binary neurons (± 1, using $0/1$ would make no difference)
- A state is fully characterized by a *thought vector* $\Psi(t) = |s_1, \ldots, s_n\rangle$.
- Connection strengths are given by $n \times n$ matrix V
- Sigmoid activation function $\sigma_\beta(r) = \frac{1}{1+e^{-\beta r}}$
- Incoming activation on i is $r = \Sigma_j V_{ij} \frac{s_j+1}{2}$
- Probability of neuron i fireing at t is

$$\sigma_\beta(\Sigma_j V_{ij} \frac{s_j + 1}{2} - V_0)$$

The biggest matrix you have ever seen

- With $n \sim 10^{11}$ neurons, the thought vector is BIG.
- Some of the positions may be *clamped* to -1 or $+1$ by external (sensory) or internal (proprioceptory) input
- Left alone, the thought vector follows a path on the n-dimensional hypercube determined by a 2^n by 2^n transition matrix P that defines the scalar product $\langle \Psi(t+1)|P|\Psi(t)\rangle$
- P is changing adiabatically (on the order of seconds or even hours) relative to the state vector changes (microsecond range), so we assume it's fixed (no learning, no senescence)
- Let ϕ_r be the unit length eigenvectors of P corresponding to eigenvalues λ_r (initially all assumed different) and express Ψ in this basis as $\psi(\Psi) = \sum_r \phi_r(\Psi)$. Since the eigenvalues are different (with probability 1) the eigenvectors are orthogonal, so the scalar product simplifies to

$$\langle \Psi(t+1)|P|\Psi(t)\rangle = \Sigma_r \lambda_r \phi_r(\alpha(t+1))\phi_r(\alpha(t))$$

Temporal evolution

- Time average $\Gamma(\alpha)$ of the probability of the system being in state α is

$$\Gamma(\alpha) = \frac{\sum_r \lambda_r^M \phi_r^2(\alpha)}{\sum_r \lambda_r^M}$$

- If there is a unique largest eigenvalue λ_1, for large M the contributions of all the other eigenvectors and eigenvalues will be negligible both in the numerator and the denominator, so the sum reduces to

$$\Gamma(\alpha) = \phi_1^2(\alpha)$$

- When there exist two or more largest eigenvalues λ_1 and λ_2, with corresponding eigenvectors ϕ_1 and ϕ_2, we obtainin

$$\Gamma(\alpha, \beta) = \frac{\lambda_1^M \phi_1^2(\alpha) + \lambda_2^M \phi_2^2(\alpha)}{\lambda_1^M + \lambda_2^M}$$

- In general, we have $\Gamma(\alpha, \beta) = \phi_1^2(\alpha)\phi_1^2(\beta) = \Gamma(\alpha)\Gamma(\beta)$

Temporal evolution *cont'd*

In general, the long term probability distribution of β is totally uncorrelated to that of α after a large number of steps. Little, 1974 interprets this as the system being largely incapable of having persistent states, and only if λ_1 and λ_2 are sufficiently close can we

have the possibility of states occurring (...) which are correlated over arbitrarily long periods of time. It is worth noting too that the characteristics of the states which so persist are describable in terms of the eigenvectors associated only with the degenerate maximum eigenvalues. In this sense these persistent states are very much simpler to describe than an arbitrary state (...) for they involve only that small set of eigenvectors associated with the degenerate maximum eigenvalues, whereas other states (require) the full set of 2^n eigenvectors.

Word vectors are in the persistent subspace

- Takeaway so far: there are gazillions of thought vectors ($2^{10^{11}}$), but only a small fraction of these will be persistent. These are the ones expressible as linear combinations of the eigenvectors corresponding to the maximal and near-maximal eigenvalues.
- We will look at the *persistent linguistic subspace* of the thought space, assuming it is spanned by some eigenvectors $p_1, \ldots, p_d$. Assuming a working lexicon of 10^4 – 10^6 entries, d is less than a million, even if these lexical entries would all be independent
- In fact, we will argue that $\dim(\mathrm{L}) < 10^3$. The word vectors generally used in practice have $300 \leq d \leq 800$.
- Our *representation space* will be $R = L \times L$ and a few copies of this

How do we compute word vectors

- Two main classes: static, and dynamic (context sensitive)
- We will discuss chiefly static, but *multi-sense* embeddings, where each word sense gets its own vector
- But first, a taste of the dynamic embeddings, in this case GPT-2. You can try it yourself at `https://talktotransformer.com` by entering any initial seed and see what continuation it generates. We take our seed from the NYT, 06/25/20, *Trump Overhaul of Campus Sex Assault Rules Wins Surprising Support* by Michael Powell.

Education Secretary Betsy DeVos fired a shot last month in the nation's culture wars, overhauling how colleges handle investigations of sexual assault and ending what she called Obama-era "kangaroo courts" on campus.

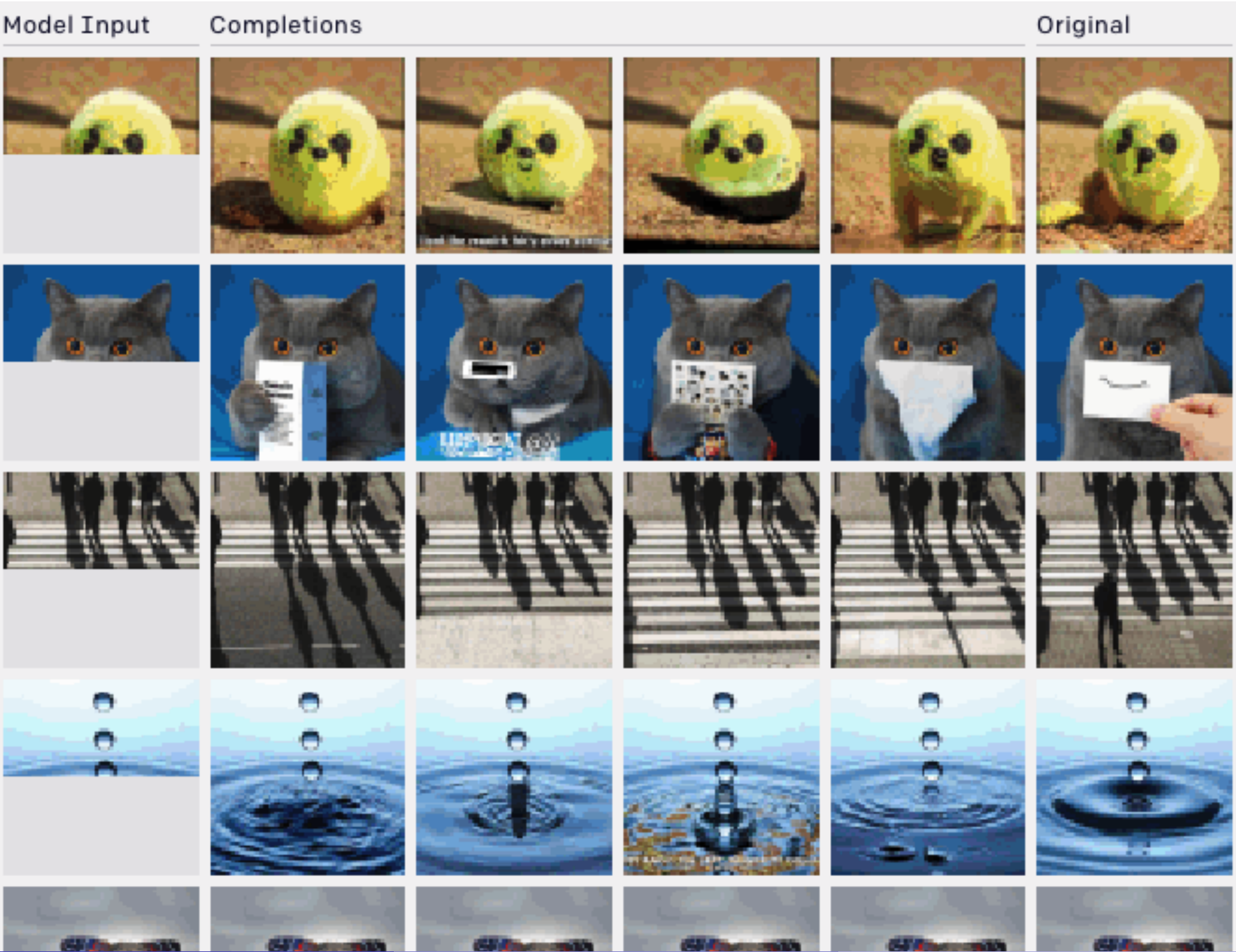
She is again laying out her plan. The goal: create an independent "office of gender equity" with prosecutorial authority to give victims of sexual assault "a complete range of rights, services, and protections." But the institute with the responsibility for enforcing the new policies will be created by the U.S. Department of Education, not by the Justice Department, her department said. That was a break from President Donald Trump, who gave the independent office the word "Attorney General."

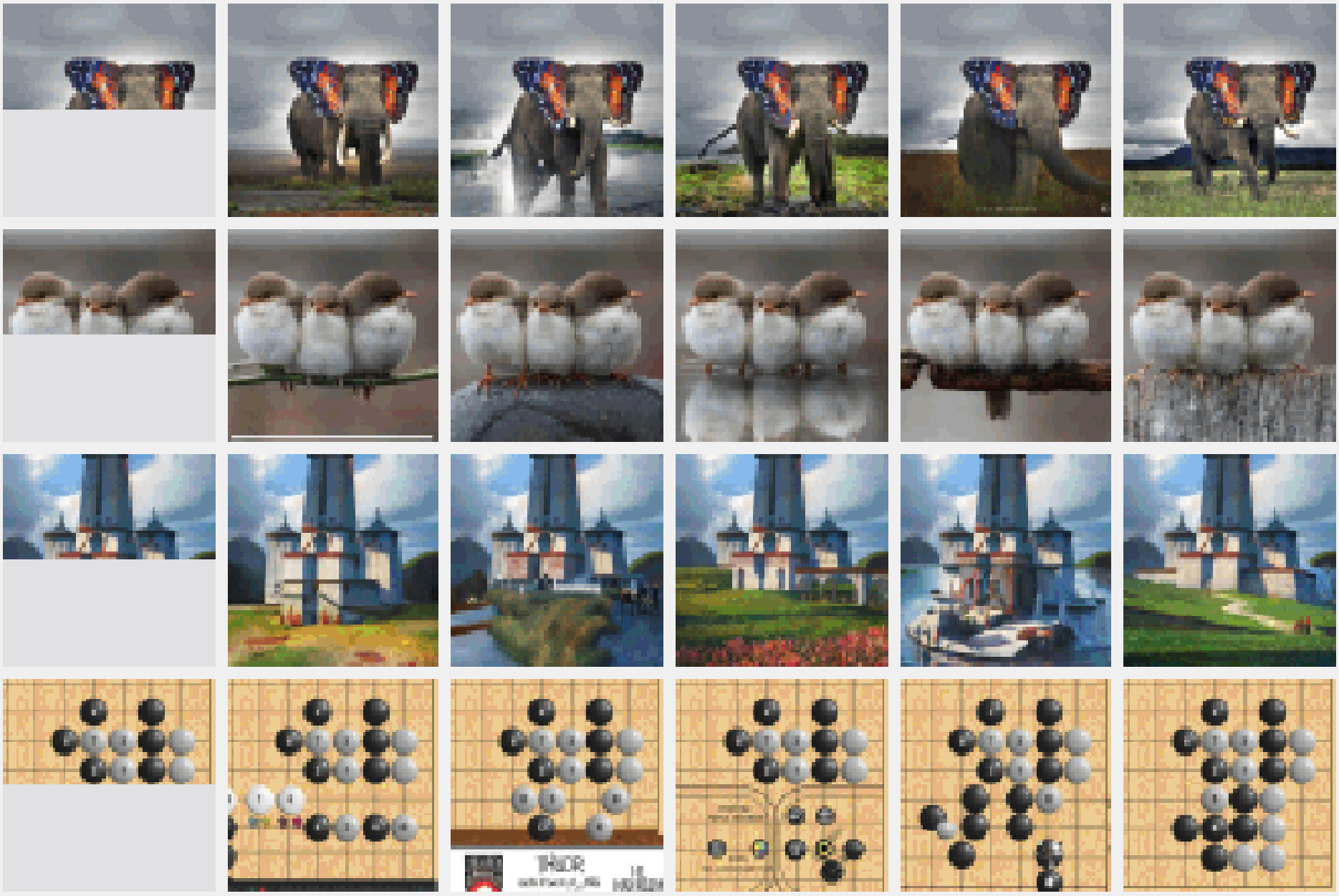
Education Secretary Betsy DeVos fired a shot last month in the nation's culture wars, overhauling how colleges handle investigations of sexual assault and ending what she called Obama-era "kangaroo courts" on campus.

That is no small move. The Obama administration's response to campus sexual assault has been tepid at best and haphazard at worst, reflecting administration overreach. Some states were given oversight of universities by the Education Department for handling the investigation, but the Obama administration did little to ensure universities complied. The new guidance begins to address those shortcomings. The new Title IX guidance centers on the extent to which schools have a responsibility to the victim of sexual assault and to any witnesses to the crime. Under the Obama guidance . . .

Education Secretary Betsy DeVos fired a shot last month in the nation's culture wars, overhauling how colleges handle investigations of sexual assault and ending what she called Obama-era "kangaroo courts" on campus.

The new Education Department rules give more protections to the accused, primarily young men who face discipline or expulsion as a result of allegations of sexual misconduct. The move set off a liberal uproar, denounced by unions representing teachers and college professors, by the National Organization for Women and by an array of Democratic senators. The Trump rules, they said, constitute a radical rollback of protections for victims who seek justice after sexual assaults.


Model Input
Completions
Original



Some unpleasant takeaways

- The grammar is good. Yet there is no overt rule of subject-predicate agreement, there are no rules, no constraints, no principles
- Generative grammar is epiphenomenal
- Any kind of appeal to UG/LAD seems misplaced
- Appeal to genetic structure is massively misplaced
- By its own measures, CL is only about 55% done
- eXplainable AI (XAI) is a key goal

Predecessors to word vectors: PCA

- Typical survey questions ask us to rank actions, objects, or statements on a scale: snails au gratin are very appetizing (+2); somewhat appetizing (+1); neither appetizing nor disgusting (0); somewhat disgusting (−1); or very disgusting (−2).
- Assume you have obtained a total of N responses from r respondents to n questions about m items: this can be summarized in a 3D array S whose (i, j, k) element is the response of respondent i to question j about object k
- Principal Component Analysis (PCA) Pearson, 1901: (a) normalize to 2d array: ignore the j, k structure and collect all responses by a given subject i in a row vector (with nm coordinates); collect these vectors in a data matrix D with r rows and $c = nm$ columns. (b) normalize the data by substracting the mean of each column from each entry in that column (means centering)

PCA *cont'd*

- $D^T D$ gives the *covariance matrix* C which has size $r \times r$, is symmetrical, and positive semidefinite. The variance in an arbitrary direction $\vec{x}$ is given by $\vec{x}^T C \vec{x}$, and the first principal component of the data is defined as the direction that maximizes the variance. To find it, we need to solve

$$\frac{d}{d\vec{x}} \vec{x}^T C \vec{x} - \lambda \vec{x}^T \vec{x}$$

  (where the second term is the Lagrange multiplier that comes from the constraint of keeping the length of $\vec{x}$ fixed)
- The critical points are obtained from solving $C\vec{x} = \lambda\vec{x}$, so the solutions λ_i are by definition the eigenvalues, and the x_i are the corresponding eigenvectors
- As always, the eigenvector basis is the winner

SVD reformulation

- Let the Singular Value Decomposition (SVD) of D be UGV^T. The columns of V are exactly the eigenvectors of C, and the positive singular values found in the diagonal matrix G (conventionally arranged to run from larger to smaller) are the square roots of the eigenvalues λ_i of C, which we use to measure the "goodness" of principal components. Writing $\Lambda = \sum_{i=1}^{c} \lambda_i$, we say, slightly misleadingly, that each λ_i *accounts for* a fraction λ_i/Λ of the total variance.
- By the Eckart–Young theorem, if the first a columns of U are collected together in U_a, the first a columns of V in V_a, and the first a singular values (by decreasing size) in G_a, the matrix $C_a = U_a G_a V_a^T$ is the best rank-a approximation (in Frobenius norm) of C. This approximation is unique as long as the first a eigenvalues are distinct, a condition generally met in the cases of interest.

Key takeaways

- Geometric intuition is nice: *dog* is closer in meaning space to *cat* than to *harpsicord*. But the key advantage of vectors is not 3d intuition, it is the apparatus (vectors, matrices, norms, eigenvalues, . . .) that lets you compute things!
- Data compression is key: just as keeping the first few terms of a Taylor series is usually a good approximation strategy, keeping the first few eigenvectors provides a good approximation (often the best possible)
- Data is always noisy!
- Surveying people is an expensive, error-prone process, the existing datasets (WordSim-353, SimLex-999, MEN) are only used for testing, not training
- Early applications like Osgood, May, and Miron, 1975 involved 100x100 matrices ($10^4 - 10^5$ matrix elements)

Predecessors to word vectors 2: LSA

- Latent Semantic Analysis Deerwester, Dumais, and Harshman, 1990 ignores what people say about relatedness, observes their behavior instead. The assumption is that they will use similar words in similar documents.
- We create a *term-document matrix* T which counts how often word i appears in document j
- We transform the entries e.g. by using

$$a_{ij} = (\log T_{ij} + 1)(\sum_j p_{ij} \log p_{ij} / \log n + 1)$$

- Looks like "secret sauce", really just entropy-based normalization
- We apply SVD, keep only the first few hundred eigenvectors
- Development held back for many years by patents and compute issues (10k+ terms, 100k+ documents, 10^9 matrix elements)

Skip-Grams with Negative Sampling (SGNS)

- Assume a gigaword corpus (word sequence) $w_1, \ldots, w_n$. The *context* of length L for word w_i are the words $w_{i-L}, \ldots, w_{i-1}, w_{i+1}, \ldots, w_{i+L}$
- Our data D are the observed word-context pairs
- We want to assign word vectors w and context vectors c so that the probability that $(w, c) \in D$ is modeled by $\sigma(< w, c >) = \frac{1}{1+e^{-<w,c>}}$ where σ is the usual "sigmoid squishing" used in neural nets
- We maximize $\log \sigma(< w, c >)$ for observered pairs, and use k "negaive samples" (pairs not in D) to maximize $k \log \sigma(< -w, c_N >)$ where the c_N are simply drawn randomly from the distribution of the contexts – we assume that a random context is unlikely to fit w
- These models came early Mikolov et al., 2013, and are still very useable. Efficient implementations exist

Implicit factorization

- We pretend to build a term-context matrix as we built the term-document matrix for LSA
- But we have 50-100k words, gw corpus (10^{14} matrix elements)
- We have an 'implicit matrix' whose element (w, c) measures the strength of the association between word w and context c using pointwise mutual information

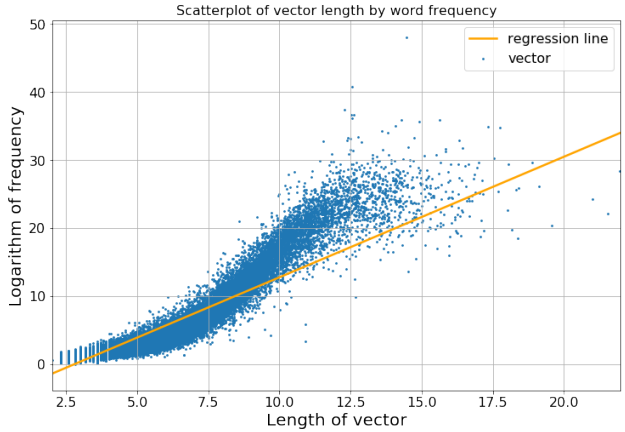
$$\log \frac{\#(w, c)D}{\#(w)\#(c)} - \log k$$

where D is corpus size and k is the degree of negative sampling Levy and Goldberg, 2014

- There are many technical tricks: using only the positive values of PMI (PPMI), using just the eigenvectors without the eigenvalues, etc etc.
- Methods built on direct optimization of difference between predicted and observed association such as `word2vec` came first, and are still very useable Mikolov et al., 2013

Fundamental properties of word vectors

1. Frequency



$$\log(p(w)) = \frac{1}{2d}||\vec{w}||^2 - \log Z \pm o(1) \tag{1}$$

2. Cooccurrence estimate

$$\log p(w, w') = \frac{1}{2d}||\vec{w} + \vec{w'}||^2 - 2\log Z \pm o(1) \tag{2}$$

3. PMI

$$\langle \vec{w}, \vec{w'} \rangle \sim \frac{\log p(w, w')}{\log p(w) \log p(w')} \tag{3}$$

Why the additive structure?

- Mikolov, Yih, and Zweig, 2013 noted *king–queen=man–woman*. Analogical puzzles like *Steve Jobs is to Apple as Bill Gates is to X* are readily solved by computing the vector Apple+Gates–Jobs and searching for the nearest vector in the embedding. This also works for morphology: not only is *boy–boys=goat–goats* but also = *mouse–mice*.
- Why? Four explanations. Pennington, Socher, and Manning, 2014 suggests

$$\frac{p(C|king)}{p(C|queen)} \approx \frac{p(C|man)}{p(C|woman)}$$

i.e. that the conditional probability of most contexts (e.g. *water*) is generally independent of the choice between *king* or *queen*, *man* or *woman*, and the ratios will deviate exactly for the same contexts like *dress, he, she, Elizabeth, Henry* . . .

Additive structure *cont'd*

- Levy and Goldberg, 2014 Suggests essentially the same, assuming

$$< w, C > \approx \log \frac{p(w, C)}{p(w)p(C)}$$

- Arora et al., 2015 Embeddings are *approximately isotropic* meaning $\mathbb{E}_w < w, w >$ is approximately the identity matrix in the sense that all its eigenvalues lie in the $[1, 1 + \delta]$ interval for some small δ. If so, $\text{argmin}_d ||a - b - c + d||_2^2$ is $\approx$ $\text{argmin}_d \mathbb{E}_w < a, w > - < b, w > - < c, w > + < d - w >$ which goes back to the same goal of finding a word w that will minimize

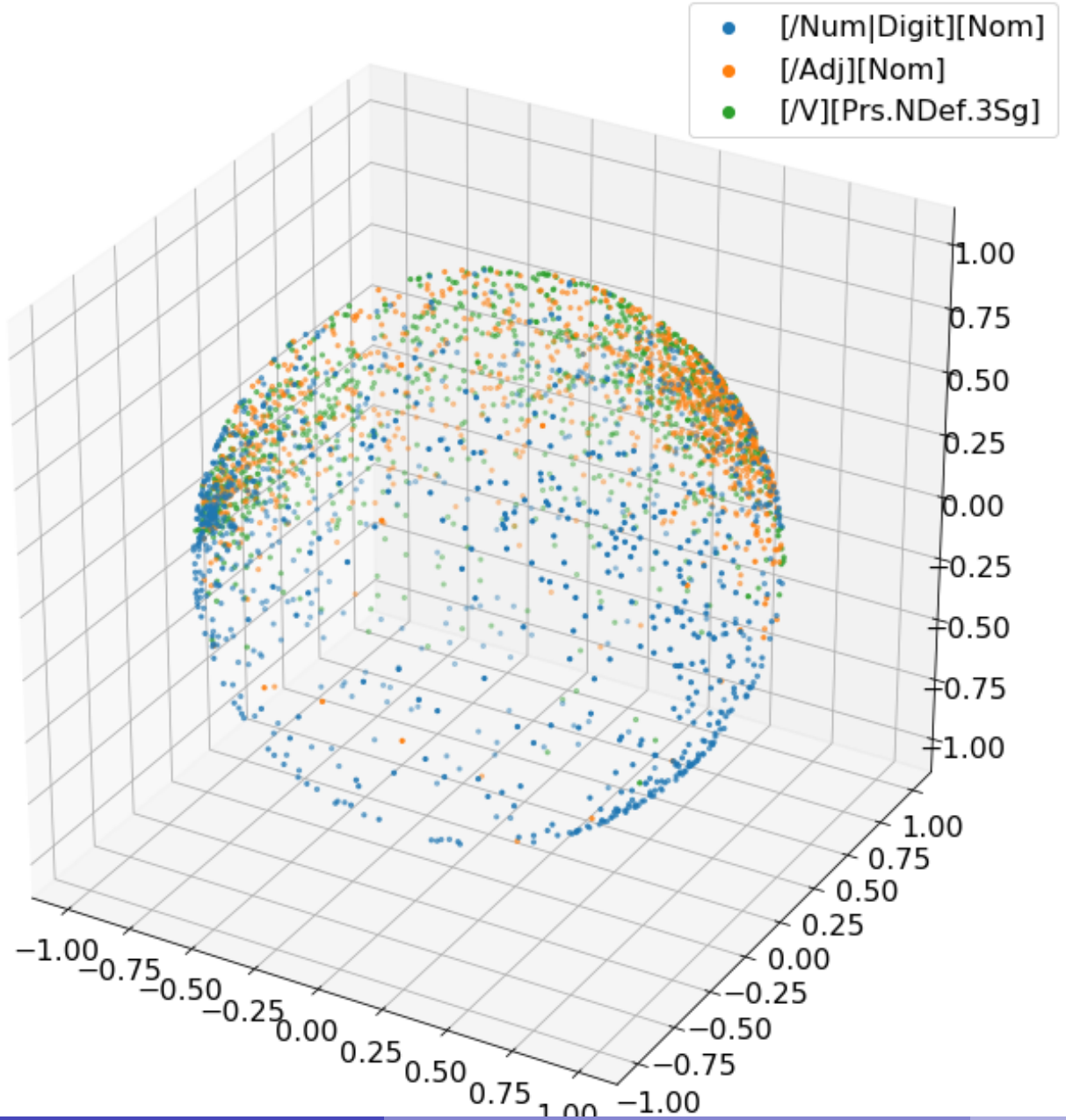
$$\sum_C \log \frac{p(C|king)}{p(C|queen)} - \log \frac{p(C|man)}{p(C|w)} \tag{4}$$

where the sum is taken over all contexts C.

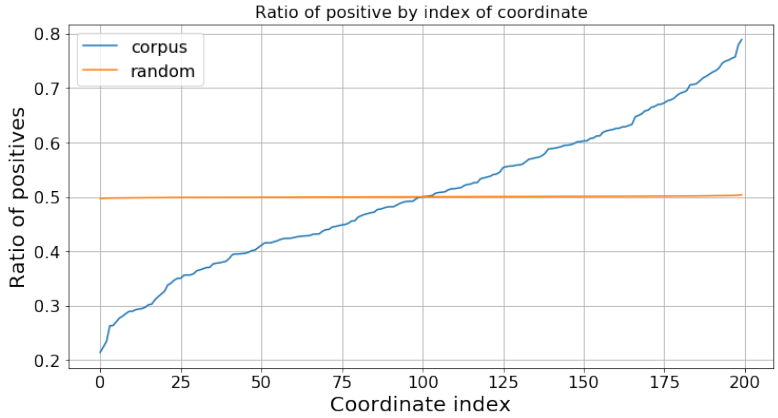
Additive structure *cont'd*

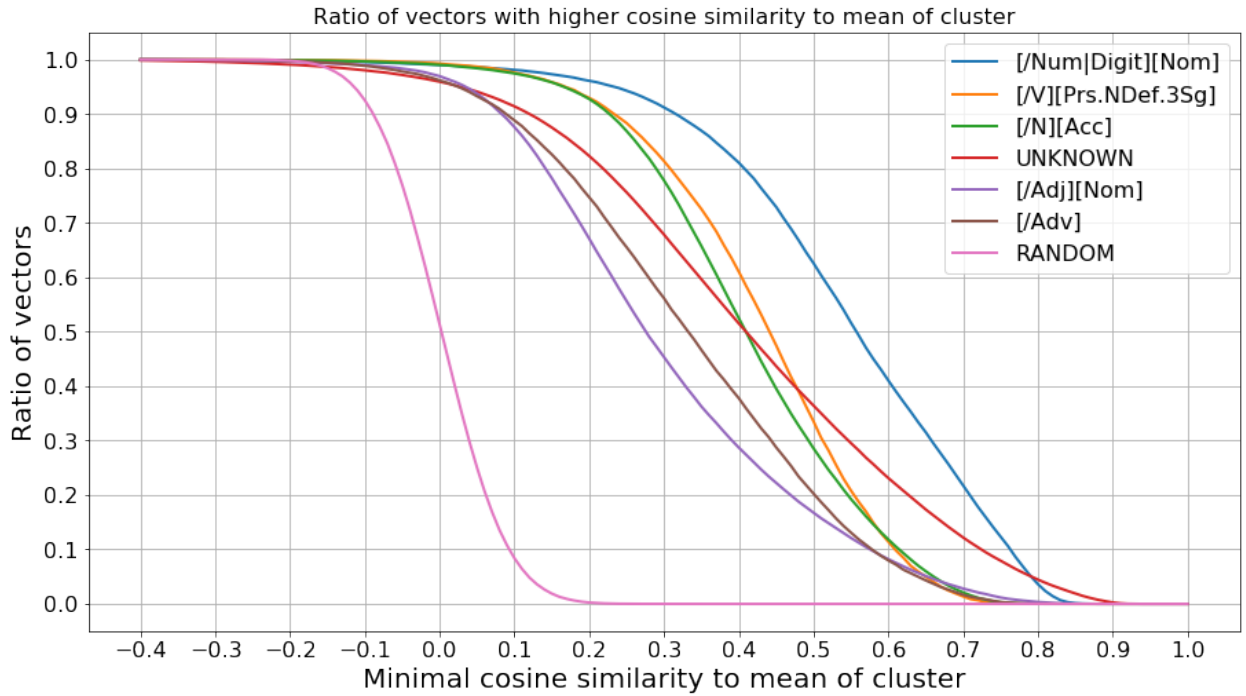
- Finally, the brilliantly titled Gittens, Achlioptas, and Mahoney, 2017 **Skipgram − Zipf + Uniform = Vector additivity** analyzes the original SGNS model and concludes that, with the assumption of uniform, rather than Zipfian frequency distribution, it is equivalent to the Sufficient Dimensionalty Reduction model of Globerson and Tishby, 2003, and will be even more additive in the sense that context vectors are simply the addition of the word vectors that appear in the context!
- None of these explanations are built entirely on realistic assumptions: context vectors are not random walks (Arora et al), frequency distributions are not uniform (Gittens et al), and there are more subtle but discernible problems with the Levy and Goldberg and the Pennington et al explanations as well. Yet the phenomenon is real, additivity is a thing. The puzzle is solved 75% of the time, but see Nissim, Noord, and Goot, 2020 for a major trick, without which we only get 45%.

More on (static) geometry



Geometry very far from random



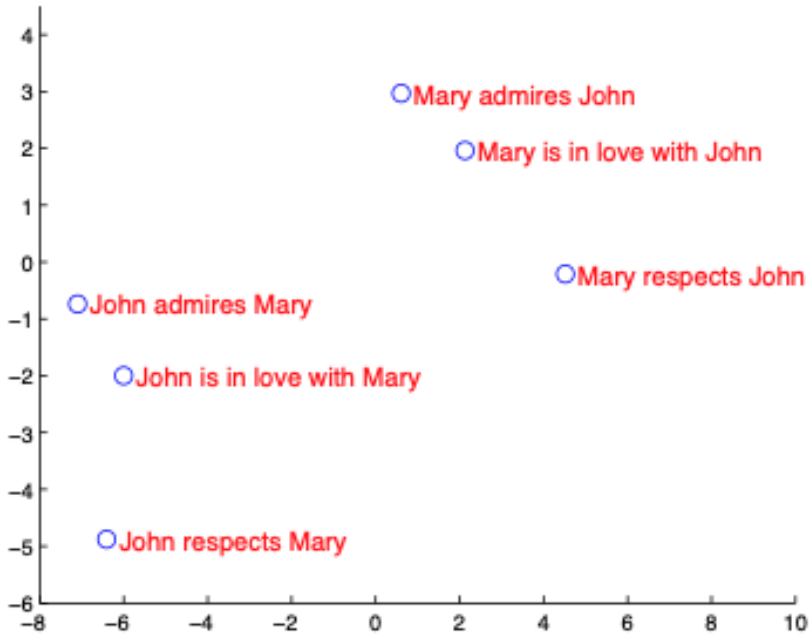


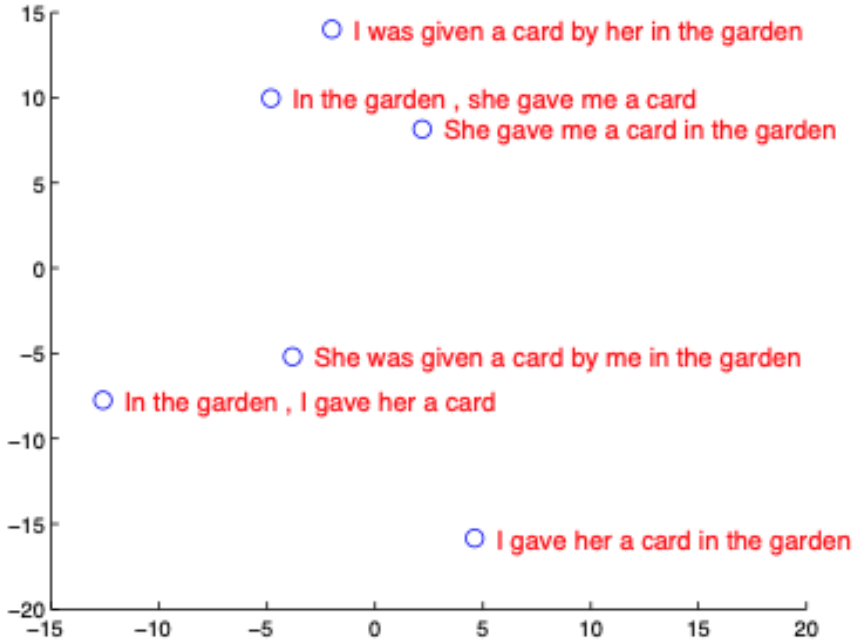
(Data from Lévai and Kornai, 2019)

Seq2seq

- Sequence to sequence (seq2seq) tasks are commonly seen in machine translation, named entity recognition, POS tagging, etc. In the *encoder* stage we present the network with a sequence of inputs (word vectors) and it maintains state in a single, fixed length state vector. When a special EOM token is presented, the network moves in a *decoder stage* and uses the state vector (and in subsequent steps, its previous output) to generate new words, until it generates an EOM.
- Sutskever, Vinyals, and Le, 2014 first demonstrated that this works well in MT, especially if the source lg sequence is presented in reverse order during training. They used LSTMs with 4 hidden layers for encoders and decoders, 160k source lg words, 80k target lg words, plus the UNK token.
- At time t, the network performs $h_t = \sigma(W^{hx}x_t + W^{hh}h_{t-1})$ and $y_t = W^{yh}h_t$, where W^{ij} is the connection strength matrix from i to j.

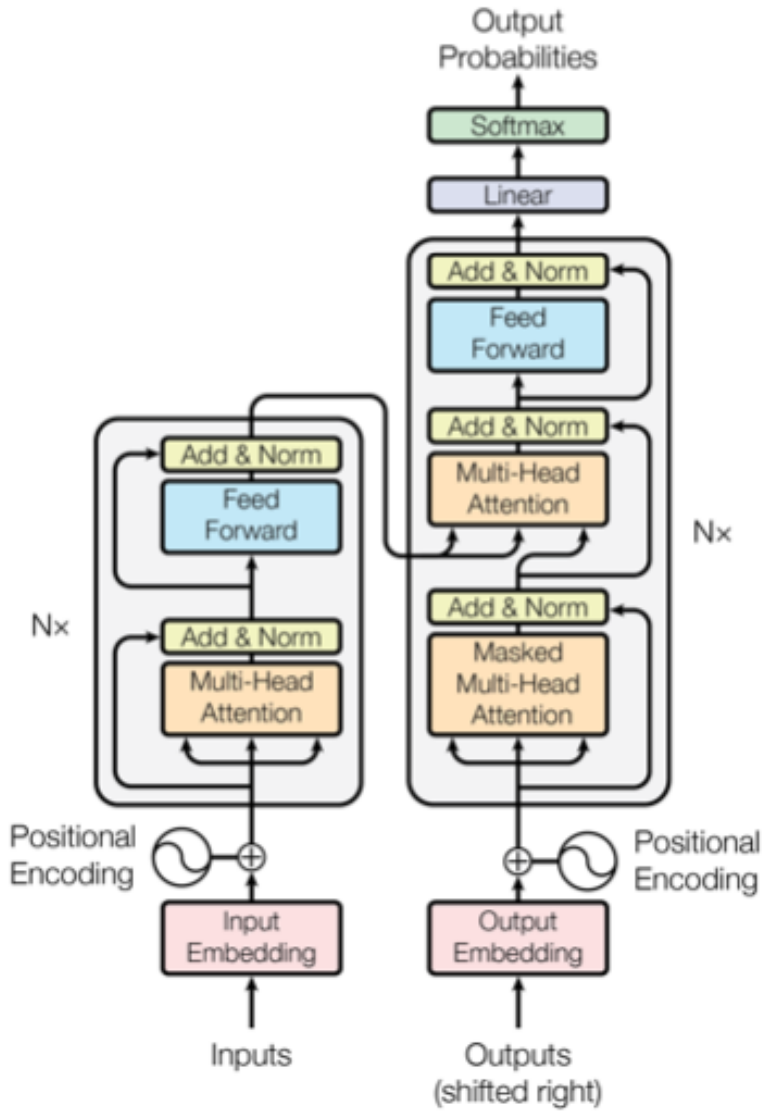
State vectors for different sentences





Transformer

Built on the idea of removing the recurrent aspects of seq2seq by replacing temporal behavior by spatial connections called *attention*
Vaswani et al., 2017



Transformer descendants

- Katharopoulos et al., 2020 defines *autoregressive* transformers, bringing back the temporal (recurrent) view
- BERT is a transformer model, using "wordpiece" vocabulary
- GPT-2 doesn't use a decoder Radford et al., 2019
- Currently at the top of the hype cycle (thousands of papers/year)
- Reproducibility crisis

Thank you!

Lecture and supporting materials available at
http://kornai.com/2021/ESSLLI

Tomorrow: (hyper)graphs, lexicon, non-compositionality
Thursday: morphology and lexicography with vectors
Friday: negation, modality, probability
Possible reading: Kornai *Vector Semantics* book draft
https://kornai.com/Drafts/advsem.pdf

Arora, Sanjeev et al. (2015). "Random Walks on Context Spaces: Towards an Explanation of the Mysteries of Semantic Word Embeddings". In: *arXiv:1502.03520v1* 4, pp. 385–399.

Deerwester, Scott C., Susan T Dumais, and Richard A. Harshman (1990). "Indexing by latent semantic analysis". In: *Journal of the American Society for Information Science* 41.6, pp. 391–407.

Gittens, Alex, Dimitris Achlioptas, and Michael W. Mahoney (2017). "Skip-Gram – Zipf + Uniform = Vector Additivity". In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, pp. 69–76. DOI: 10.18653/v1/P17-1007. URL: http://aclweb.org/anthology/P17-1007.

Gyenis, Zalán (2018). "Skeleton in the Euclidean closet". In: *K+K=120*. Ed. by Beáta Gyuris, Katalin Mády, and Gábor Recski.

Hertz, John A, Anders S Krogh, and Richard G Palmer (1991). *Introduction to the Theory of Neural Computation*. Vol. 1. Redwood City, CA: Addison-Wesley.

Katharopoulos, Angelos et al. (2020). *Transformers are RNNs: Fast Autoregressive Transformers with Linear Attention*. arXiv: 2006.16236 [cs.LG].

Katz, J. and Jerry A. Fodor (1963). "The structure of a semantic theory". In: *Language* 39, pp. 170–210.

Lévai, Dániel and András Kornai (Jan. 2019). "The impact of inflection on word vectors". In: *XV. Magyar Számitógépes Nyelvészeti Konferencia*.

Levy, Omer and Yoav Goldberg (2014). "Neural Word Embedding as Implicit Matrix Factorization". In: *Advances in Neural Information Processing Systems 27*. Ed. by Z. Ghahramani et al., pp. 2177–2185.

Little, W. A. (1974). "The existence of persistent states in the brain". In: *Mathematical Biosciences* 19, pp. 101–120.

McCulloch, W.S. and W. Pitts (1943). "A logical calculus of the ideas immanent in nervous activity". In: *Bulletin of mathematical biophysics* 5, pp. 115–133.

Mikolov, Tomas, Wen-tau Yih, and Geoffrey Zweig (2013). "Linguistic Regularities in Continuous Space Word Representations". In: *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2013)*. Atlanta, Georgia: Association for Computational Linguistics, pp. 746–751.

Mikolov, Tomas et al. (2013). "Distributed Representations of Words and Phrases and their Compositionality". In: *Advances in Neural Information Processing Systems 26*. Ed. by C.J.C. Burges et al. Curran Associates, Inc., pp. 3111–3119. URL: https://bit.ly/39HikH8.

Minsky, Marvin and Seymour Papert (1988). *Perceptons (2nd ed.)* MIT Press.

Nissim, Malvina, Rik van Noord, and Rob van der Goot (2020). "Fair is Better than Sensational: Man is to Doctor as Woman is to Doctor". In: *Computational Linguistics* 0.0, pp. 1–11. DOI: 10.1162/coli_a_00379. eprint:

https://doi.org/10.1162/coli_a_00379. URL: https://doi.org/10.1162/coli_a_00379.

Osgood, Charles E., William S. May, and Murray S. Miron (1975). *Cross Cultural Universals of Affective Meaning*. University of Illinois Press.

Pearson, K. (1901). "LIII. On lines and planes of closest fit to systems of points in space". In: *Philosophical Magazine Series 6* 2.11, pp. 559–572.

Pennington, Jeffrey, Richard Socher, and Christopher Manning (2014). "Glove: Global Vectors for Word Representation". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, pp. 1532–1543. DOI: 10.3115/v1/D14-1162. URL: http://www.aclweb.org/anthology/D14-1162.

Pollack, Jordan B. (1989). "No Harm Intended: A Review of the Perceptrons". In: *Journal of Mathematical Psychology* 33, pp. 358–365.

Radford, Alec et al. (2019). "Language Models are Unsupervised Multitask Learners". https://github.com/openai/gpt-2. URL: https://d4mucfpksywv.cloudfront.net/better-language-models/language-models.pdf.

Rosenblatt, Frank (1957). *The Perceptron: a perceiving and recognizing automaton*. Tech. rep. 85-460-1.

Rumelhart, David E, Geoffrey E Hinton, and Ronald J Williams (Sept. 1985). *Learning internal representations by error propagation*. Tech. rep. ICS 8504. San Diego, California: Institute for Cognitive Science, University of California.

Sutskever, Ilya, Oriol Vinyals, and Quoc V. Le (2014). "Sequence to Sequence Learning with Neural Networks". In: *Proc. NIPS*. Montreal, CA, pp. 3104–3112. URL: http://arxiv.org/abs/1409.3215.

Vaswani, Ashish et al. (2017). "Attention is All you Need". In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Curran Associates, Inc., pp. 5998–6008. arXiv: 1706.03762 [cs.CL]. URL:

http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.