Bridging the Gap Between Computers and Semantics Through Learned Information Representations

Max Daniels
Northeastern University

SUMS: Math and Thinking Systems, March 2020

Table of Contents

Table of Contents

Introduction

Deep learning techniques allow learned representations with semantically relevant characteristics.

Representations are highly applicable in downstream tasks:

- Translation, Image editing, etc
- *Compressive Sensing*: 5-10x improved compression over sparsity methods

What are these representations, and why do they work?

Key Assumptions

- Distributional Hypothesis: Words which appear in similar *contexts* have similar *meaning*.
- Manifold Hypothesis: *Typical* data has relatively few factors of variation. In the right representation, typical data is *compressible*.

Table of Contents

Data Modeling

Given access to a dataset $X = \{ x_i \}_{i=1}^{N}$ of i.i.d. samples x_i.
Assume

$$x_i \sim p_\theta(x) \in \mathcal{P} \tag{1}$$

Where $\mathcal{P}$ contains *data generating distributions* parametrized by θ.
Goal: recover θ from $\mathcal{D}$.

Generative Modeling

1. Classical Method: *Maximum Likelihood*

$$\max_{\theta} \prod_{i=1}^{N} p_\theta(x_i) \iff \max_{\theta} \sum_{i=1}^{N} \log(p_\theta(x_i)) \tag{2}$$

2. To model p_θ using arbitrary functions (neural networks), we must normalize
3. Computing $\int p_\theta(x) \, , dx$ is intractable

How to perform generative modeling without requiring analytic integral of p_θ?

Noise Contrastive Estimation

Rather than learning data, learn to distinguish the data from noise. Generate noise $\{ y_i \}_{i=1}^{N} \sim q(y)$ and maximize

$$\mathcal{L}_N(\theta) = \frac{1}{2N} \log \left(\prod_{i=1}^{N} p_\theta(x_i) \cdot (1 - p_\theta(y_i)) \right) \tag{3}$$

$$= \sum_{i=1}^{N} \ln[p_\theta(x_i)] + \ln[1 - p_\theta(y_i)] \tag{4}$$

Noise Contrastive Estimation

Theorem ([GH12], Theorem 2: Consistency)

If conditions (a) to (c) are fulfilled then $\hat{\theta}_N$ converges in probability to θ^*, i.e. $\hat{\theta}_N \xrightarrow{P} \theta^*$.

- (a) $p_n(.)$ *is nonzero whenever* $p_d(.)$ *is nonzero*
- (b) $\sup_\theta |\mathcal{L}_N(\theta) - \mathcal{L}(\theta)| \xrightarrow{P} 0$
- (c) $\mathcal{I} = \int g(x)g(x)^T P(x) p_d(x)\, dx$ *has full rank, where*

$$P(x) = \frac{p_n(x)}{p_d(x) + p_n(x)}, \quad g(x) = \nabla_\theta \ln p_\theta(x)|_{\theta^*}$$

Word2Vec: Simplified NCE

Model words $w_i \in V$ of a training sequence $w_1, \ldots, w_N$ along with contexts $C_i = \{\ w_k\ \}_{i-c}^{i+c}$.

the quick brown fox jumps over the lazy dog

$$\mathcal{D} = \{\ (w_i, C_i)\ \} \tag{5}$$

$$p_\theta(C_i|w_i) = \prod_{w_j \in C_i} p_\theta(w_j|w_i) \tag{6}$$

- High dimensionality: $w_i = e_i \in \mathbb{R}^{|V|}$, every word gets a dimension.
- Noise density p_n gives random words.

Word2Vec: Simplified NCE

Key Simplification: p_θ learns a projection T_θ of words into a *low dimensional representation space*.

Then, $p(w_j|w_i)$ is a simple logistic regression:

$$p_\theta(w_j|w_i) = \frac{1}{1 + e^{-\langle T_\theta w_i, T_\theta w_j \rangle}} \tag{7}$$

Power of Word2Vec

Word vectors exhibit extremely useful semantic properties:

Czech + currency	Vietnam + capital	German + airlines
koruna	Hanoi	airline Lufthansa
Check crown	Ho Chi Minh City	carrier Lufthansa
Polish zolty	Viet Nam	flag carrier Lufthansa
CTK	Vietnamese	Lufthansa

Table: Vector compositionality results of [Mik+13].

Takeaways

1. New modeling paradigm: learning by comparison.
2. To avoid computing *absolute probabilities*, make a relative comparison between p_θ and p_n
3. By creative choice of modeling class $\mathcal{P}$, parameters θ are useful for downstream tasks

Table of Contents

Generative Adversarial Networks

Application of generative modeling: generate new data samples. Sampling p_θ directly is difficult, again due to intractability of $\int dp_\theta$.

Must we pay the cost of sampling an unknown density in high dimensions if the data is probably low dimensional anyways?

Generative Adversarial Networks

GAN Approach:

- Assume the data is low dimensional. Pick your favorite analytic prior distribution over the representation space.
- Construct parametrized mappings from representations to data. These are *Generators*.
- Each Generator induces its own density over the data space.
- The density estimator plays the role of a *Critic*.

Generative Adversarial Networks

With access to a density estimate for the data, we can optimize the quality of a Generator.

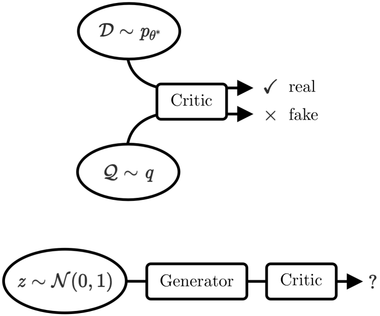

Figure: Applying contrastive density estimation to learn good generators.

GAN Objective

Setup:

1. Data $x \in \mathbb{R}^n$ with density $x \sim p_{\text{data}}(x)$.
2. Representations $z \in \mathbb{R}^m$, $n \gg m$ with density $z \sim \mathcal{N}(0,1)$.
3. Generator $G(z) \in C^1(\mathbb{R}^m, \mathbb{R}^n)$, maps $z \mapsto x$.
4. Critic $D(x) \in C^1(\mathbb{R}^n, [0,1])$, maps $x \mapsto p_D(x)$.

Optimization objective [Goo+14]:

$$\min_G \max_D V(D,G) = \underbrace{\mathbb{E}_{x \sim p_{\text{data}}}[\log D(x)]}_{\text{(I)}} + \underbrace{\mathbb{E}_{z \sim \mathcal{N}(0,1)}[\log 1 - D(G(z))]}_{\text{(II)}} \tag{8}$$

$$\min_G \max_D V(D,G) = \underbrace{\mathbb{E}_{x \sim p_{\text{data}}}[\log D(x)]}_{\text{(I)}} + \underbrace{\mathbb{E}_{z \sim \mathcal{N}(0,1)}[\log 1 - D(G(z))]}_{\text{(II)}} \tag{9}$$

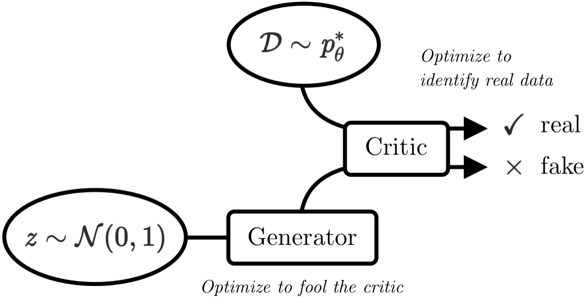

Figure: GAN optimization.

Power of GAN Representations

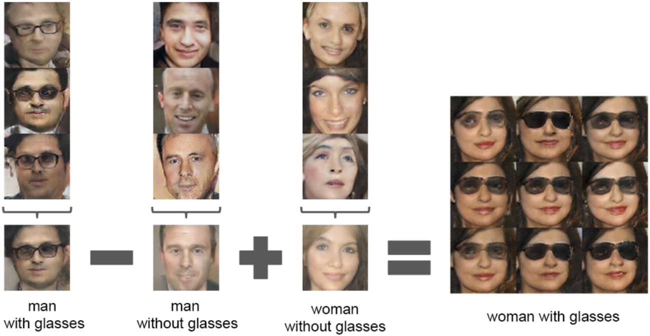

Figure: Compositionality of learned image representations [RMC15].

Takeaways

- The Critic: contrastive estimation approximates the data density function.
- The Generator: by explicitly parametrizing an approximate data manifold, we can also have tractable (approximate) sampling
- Optimizing the GAN objective trains both in parallel.

End

Thanks!

Download these slides and see associated work:

Michael U. Gutmann and Aapo Hyvärinen. "Noise-Contrastive Estimation of Unnormalized Statistical Models, with Applications to Natural Image Statistics". In: *J. Mach. Learn. Res.* 13.null (Feb. 2012), pp. 307–361. ISSN: 1532-4435.

Ian Goodfellow et al. "Generative Adversarial Nets". In: *Advances in Neural Information Processing Systems 27*. Ed. by Z. Ghahramani et al. Curran Associates, Inc., 2014, pp. 2672–2680. URL: http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf.

Tomas Mikolov et al. "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., 2013, pp. 3111–3119. URL: http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf.

Alec Radford, Luke Metz, and Soumith Chintala. "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks". In: *arXiv e-prints*, arXiv:1511.06434 (Nov. 2015), arXiv:1511.06434. arXiv: 1511.06434 [cs.LG].