February 12: *Machine Learning for Materials Science*

March 12: *Mathematics of Privacy*

April 9: *Mathematics of Gravitational Waves*

May 14: *Algebraic Geometry*

June 11: *Mathematics of Transportation*

July 9: *Cryptography & Cybersecurity*

August 13: *Machine Learning in Medicine*

September 10: *Logic and Foundations*

October 8: *Mathematics of Quantum Physics*

November 12: *Quantum Encryption*

December 10: *Machine Learning for Text*

* Webinar posted

Made possible by support for BMSA from the National Science Foundation Division of Mathematical Sciences and the Department of Energy Advanced Scientific Computing Research
Machine Learning for Texts: Understanding Embeddings

Tengyu Ma, Stanford University

Assistant Professor of Computer Science and Statistics
Breakthroughs in Natural Language Processing
Machine Learning (Supervised Learning)

- Find a function $f$
  
  $\begin{array}{c}
  \text{Input} \\
  \rightarrow \\
  \text{output}
  \end{array}$

- Translation

  This is a talk about machine learning for texts.
  
  Ceci est une discussion sur l'apprentissage automatique pour les textes.

- Sentiment analysis

  This seminar series is fantastic. 👍 positive sentiment
Machine Learning (Supervised Learning)

- Find a function $f$

\[ \text{Input} \quad f \quad \rightarrow \quad \text{output} \]

How do we represent texts inputs as numerical values?
Classic “One-hot” and “Bag-of-words” Representation

- Vocabulary = \{a, aardvark, aardwolf, ..., zymurgy\} of size $N$

happy $\rightarrow$ $\nu_{\text{happy}} = \begin{bmatrix}
0 \\
0 \\
0 \\
\vdots \\
1 \\
\vdots \\
\vdots \\
0
\end{bmatrix}$ $\in \mathbb{R}^N$

\begin{align*}
a & \text{ aardvark} \\
\text{aardwolf} & \text{ happy} \\
\text{happy} & \text{ zymurgy}
\end{align*}
Vocabulary = \{a, aardvark, aardwolf, \ldots, zymurgy\} of size $N$

A happy aardwolf $\rightarrow$

\[ \begin{bmatrix} 1 \\ 0 \\ 1 \\ \vdots \\ \vdots \\ \vdots \\ 1 \\ \vdots \\ \vdots \\ 0 \end{bmatrix} \rightarrow v = \begin{bmatrix} \text{a} \\ \text{aardvark} \\ \text{aardwolf} \\ \vdots \\ \vdots \\ \vdots \\ \text{happy} \\ \vdots \\ \vdots \\ \text{zymurgy} \end{bmatrix} \in \mathbb{R}^N \]
Embeddings (in Machine Learning)

$x \in \mathcal{X}$

complicated space

$\nu_x \in \mathbb{R}^d$

Euclidean space with meaningful inner products
Goal:

- embedding captures semantics information (ideally via linear algebraic operations)
- inner products characterize similarity
  - similar words have large inner products
- differences characterize relationship
  - analogous pairs have similar differences

Picture credit: Chris Olah’s blog post
Distributional Hypothesis of Meaning ([Harris’54], [Firth’57])

Meaning of a word is determined by words it co-occurs with.

\[ \text{Def: } \Pr(x, y) \triangleq \text{prob. of co-occurrences of } x, y \text{ in a window of size 5} \]

"a window of size 5"

- Rows of co-occurrence matrix are reasonable embeddings [Lund-Burgess’96]

- [Church-Hanks’90]

\[ \nu_x = \text{row of PMI}(x, y) \triangleq \log \frac{\Pr[x, y]}{\Pr[x] \Pr[y]} \]

(PMI = point-wise mutual information)
1. Compute $\text{PMI}(x, y) = \log \frac{\Pr[x, y]}{\Pr[x] \Pr[y]}$

2. Take rank-300 SVD (best rank-300 approximation) of PMI

   $\iff$ Fit $\text{PMI}(x, y) \approx \langle v_x, v_y \rangle$ where $v_x \in \mathbb{R}^{300}$

"Linear structure" in the found $v_x$'s:

$\nu_{\text{woman}} - \nu_{\text{man}} \approx \nu_{\text{queen}} - \nu_{\text{king}} \approx \nu_{\text{uncle}} - \nu_{\text{aunt}} \approx \ldots$
Non-linear Embedding methods

- **word2vec** [Mikolov et al’13]:
  \[
  \Pr[x_{i+6} \mid x_{i+1}, \ldots, x_{i+5}] \propto \exp(v_{x_{i+6}}, \frac{1}{5}(v_{x_{i+1}} + \ldots + v_{x_{i+5}}))
  \]

- **GloVe** [Pennington et al’14]:
  \[
  \log \Pr[x, y] \approx \langle v_x, v_y \rangle + s_x + s_y + C
  \]

- **[Levy-Goldberg’14] (Previous slide)**
  \[
  \text{PMI}(x, y) = \log \frac{\Pr[x, y]}{\Pr[x] \Pr[y]} \approx \langle v_x, v_y \rangle + C
  \]

Logarithm (or exponential) seems to exclude linear algebra!
Where does the log come from?

[Arora et al.’16, c.f. Levy-Goldberg’14, Pennington et al’14]

- For most of the words $\chi$:

\[
\frac{\Pr[\chi | \text{king}]}{\Pr[\chi | \text{queen}]} \approx \frac{\Pr[\chi | \text{man}]}{\Pr[\chi | \text{woman}]}
\]

- For $\chi$ unrelated to gender: LHS, RHS $\approx 1$

- for $\chi = \text{dress}$, LHS, RHS $\ll 1$; for $\chi = \text{John}$, LHS, RHS $\gg 1$

\[
\Rightarrow \sum_{\chi} \left( \log \frac{\Pr[\chi | \text{king}]}{\Pr[\chi | \text{queen}]} - \log \frac{\Pr[\chi | \text{man}]}{\Pr[\chi | \text{woman}]} \right)^2
\]

\[
|| \text{PMI}(\text{king,·}) - \text{PMI}(\text{queen,·}) - \text{PMI}(\text{man,·}) + \text{PMI}(\text{woman,·}) ||^2_2 \approx 0
\]

- Rows of PMI matrix has “linear structure”
Empirically can find vectors $v_x$’s such that

$$\text{PMI}(x, y) \approx \langle v_x, v_y \rangle$$

1. PMI is not necessarily PSD

2. Relative approximation error is high (17%); Low-dimensional $v_x$’s have better linear structure than rows of PMI
RAND-WALK: A Generative Model for Language [Arora et al’16]

Hidden Markov Model:
- discourse vector $c_t \in \mathbb{R}^d$ governs the discourse/theme/context of time $t$
- words $w_t$ (observable); embedding $v_{w_t} \in \mathbb{R}^d$ (parameters to learn)
- log-linear observation model

$$\Pr[w_t \mid c_t] \propto \exp\langle v_{w_t}, c_t \rangle$$

Closely related to [Mnih-Hinton’07]
Why Does Dimension Reduction Help?

Empirically can find vectors $\nu_x$'s such that

$$\text{PMI}(x, y) \approx \langle \nu_x, \nu_y \rangle$$

1. PMI is not necessarily PSD and low-rank
   - Under rand-walk model, PMI is approximately PSD and low-rank
2. Relative approximation error is high (17%); Low-dimensional $\nu_x$'s have **better** linear structure than rows of PMI
   - Dimension-reduction reduces the noises
Summary and Looking Ahead

- Theoretical explanations of embeddings methods
  - Popular embeddings methods, such as PMI+SVD, word2vec, Glove can be viewed as algorithms for learning a generative model of language

- Follow-up works: embeddings for sentences, polysemous words, rare words [Arora et al.’17,18a&b ...]

- Open directions:
  - Understanding the state-of-the-art contextualized embeddings (Elmo, Bert, etc.)
  - Optimizations of the embeddings
  - Understanding other algorithms for other tasks in NLP (machine translation, etc.)
  - A theory of representation learning
Main References


- Linear Algebraic Structure of Word Senses, with Applications to Polysemy. Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. TACL, 2018

- A La Carte Embedding: Cheap but Effective Induction of Semantic Feature Vectors. Mikhail Khodak, Nikunj Saunshi, Yingyu Liang, Tengyu Ma, Brandon Stewart, Sanjeev Arora. ACL, 2018


Assistant Professor in Computer Science
marine@cs.umd.edu

Toward Breaking Language Barriers with Neural Machine Translation

Marine Carpuat, University of Maryland
6,800 living languages
600 with written tradition
The Chinese capital, with its surprisingly high-speed Internet, sophisticated technology such as face-recognition software, has invested heavily in artificial intelligence and has unrivaled international energy, and is one of the most exciting cities for exploration-minded foreigners.

[Cam & Carpuat WMT 2018]
An English sentence $e$ is translated into the French sentence

$$f^* = \arg\max_f p(f | e; \theta)$$

$$\theta^* = \arg\max_\theta \sum_i \log p(f_i | e_i; \theta)$$
Translation as Deep Learning

\[ f = (\text{La, croissance, économique, s'est, ralentie, ces, dernières, années, .}) \]

\[ e = (\text{Economic, growth, has, slowed, down, in, recent, years, .}) \]

\[ p(f \mid e; \theta) = \prod_{t=1}^{|f|} p(f_t \mid f_{<t}, e; \theta) \]
Translation as Deep Learning: Challenges

- Requires millions of translation examples not available for many languages!
- Raises fundamental machine learning challenges
  - Intractably large output space, infinitely many correct outputs...
- Makes errors that have real world impact
  - Yet models are opaque, and developed independently from use cases
Some approaches:

Learn from related languages

Learn from monolingual text

Improve the training objective
Training Problem: Exposure Bias, a Gap Between Training and Inference

Maximum Likelihood Training

Reference

Inference

Model Translation

We will make dinner.

\[
P(f|e) = \prod_{t=1}^{T} p(f_t|f_{<t}, e)
\]

Loss = \[
\sum_{t=1}^{T} \log p(f_t|f_{<t}, e)
\]
How to Address Exposure Bias?

Expose models to their own predictions during training
But how to compute the loss when the partial translation diverges from the reference?

Our method:

1. **Generate translation prefixes** via differentiable sampling
2. Learn to **align** the reference words with sampled prefixes
Our Solution: Align Reference with Partial Translations

Reference: `<s> We made dinner </s>`

**Soft Alignment**

\[ a_i \propto \exp(\text{Embed}_{\text{dinner}} \cdot h_i) \]

\[
\begin{align*}
    a_1 \log p(``dinner`` | ` `<s>` `, source) + & a_2 \log p(``dinner`` | ` `<s> We`, source) + \\
    a_3 \log p(``dinner`` | ` `<s> We will`, source) + & a_4 \log p(``dinner`` | ` `<s> We will make`, source)
\end{align*}
\]

[Xu & Carpuat NAACL 2019]
Some approaches:

- Learn from related languages
- Learn from monolingual text
- Improve the training objective
Can machine translation help human translators and interpreters be more productive?

What errors matter most for different use cases?

Can we tailor machine translation output to different audiences?

Toward more user-centered machine translation
Controlling MT Complexity for Different Audiences

The Mauritshuis museum is staging an exhibition focused solely on 17th century self-portraits.

El museo Mauritshuis abre una exposición dedicada a los autorretratos del siglo XVII.

Audience: fluent English speaker
Controlling MT Complexity for Different Audiences

El museo Mauritshuis abre una exposición dedicada a los autorretratos del siglo XVII.

Audience: 2nd language learner

Complexity Controlled MT

The Mauritshuis museum is going to show self-portraits.
Adapting translation output to different audiences via multi-task learning

Multi-task loss = \[ \sum_{(s_i, g_e, e_o)} \log P(e_o | s_i, g_e; \theta) + \sum_{(e_i, g_e, e_o)} \log P(e_o | e_i, g_e; \theta) + \sum_{(s_i, e_o)} \log P(e_o | s_i; \theta) \]

- **L_{CMT}**: Spanish sentences translated into simpler English
- **L_{Simplify}**: Complex English sentences paired with simpler English
- **L_{MT}**: Spanish-English translation examples
Deep neural networks provide a powerful framework to model translation.

How can we improve quality in low-resource settings?

How can we make machine translation more user-centric when it works well?


February 12: Machine Learning for Materials Science*

March 12: Mathematics of Privacy*

April 9: Mathematics of Gravitational Waves*

May 14: Algebraic Geometry*

June 11: Mathematics of Transportation*

July 9: Cryptography & Cybersecurity*

August 13: Machine Learning in Medicine*

September 10: Logic and Foundations*

October 8: Mathematics of Quantum Physics*

November 12: Quantum Encryption*

December 10: Machine Learning for Text

Made possible by support for BMSA from the National Science Foundation Division of Mathematical Sciences and the Department of Energy Advanced Scientific Computing Research

* Webinar posted