CS 2770: Computer Vision
Language and Vision;
Recurrent Neural Networks

Prof. Adriana Kovashka
University of Pittsburgh
April 3, 2018
Plan for this lecture

• Image captioning
  – Examples
  – Tool: Recurrent neural networks
  – Captioning for video
  – Diversifying captions
  – Visual-semantic spaces

• Visual question answering
“It was an arresting face, pointed of chin, square of jaw. Her eyes were pale green without a touch of hazel, starred with bristly black lashes and slightly tilted at the ends. Above them, her thick black brows slanted upward, cutting a startling oblique line in her magnolia-white skin—that skin so prized by Southern women and so carefully guarded with bonnets, veils and mittens against hot Georgia suns”

Scarlett O’Hara described in Gone with the Wind
This is a picture of one sky, one road and one sheep. The gray sky is over the gray road. The gray sheep is by the gray road.

Here we see one road, one sky and one bicycle. The road is near the blue sky, and near the colorful bicycle. The colorful bicycle is within the blue sky.

This is a picture of two dogs. The first dog is near the second furry dog.
Some pre-RNN bad results

Missed detections:
- Here we see one potted plant.

False detections:
- There are one road and one cat. The furry road is in the furry cat.
- This is a picture of one tree, one road and one person. The rusty tree is under the red road. The colorful person is near the rusty tree, and under the red road.

Incorrect attributes:
- This is a photograph of two sheeps and one grass. The first black sheep is by the green grass, and by the second black sheep. The second black sheep is by the green grass.
- This is a photograph of two horses and one grass. The first feathered horse is within the green grass, and by the second feathered horse. The second feathered horse is within the green grass.

Kulkarni et al., CVPR 2011
Results with Recurrent Neural Networks

"man in black shirt is playing guitar."

"construction worker in orange safety vest is working on road."

"two young girls are playing with lego toy."

"boy is doing backflip on wakeboard."

Karpathy and Fei-Fei, CVPR 2015
Recurrent Networks offer a lot of flexibility:

- **one to one**
- **one to many**
- **many to one**
- **many to many**

*vanilla neural networks*

Andrej Karpathy
Recurrent Networks offer a lot of flexibility:

- **One to one**
- **One to many**
- **Many to one**
- **Many to many**

*Example:* **Image captioning**

Image -> sequence of words
Recurrent Networks offer a lot of flexibility:

- one to one
- one to many
- many to one
- many to many

E.g., sentiment classification: sequence of words -> sentiment
Recurrent Networks offer a lot of flexibility:

- one to one
- one to many
- many to one
- many to many

E.g. machine translation:
seq of words -> seq of words
Recurrent Networks offer a lot of flexibility:

e.g. video classification on frame level
Recurrent Neural Network
Recurrent Neural Network

usually want to output a prediction at some time steps
Recurrent Neural Network

We can process a sequence of vectors $\mathbf{x}$ by applying a recurrence formula at every time step:

$$h_t = f_W(h_{t-1}, x_t)$$

- New state
- Old state
- Input vector at some time step
- Some function with parameters $W$
Recurrent Neural Network

We can process a sequence of vectors $\mathbf{x}$ by applying a recurrence formula at every time step:

$$h_t = f_W(h_{t-1}, x_t)$$

Notice: the same function and the same set of parameters are used at every time step.
(Vanilla) Recurrent Neural Network

The state consists of a single "hidden" vector $h$:

$$h_t = f_W(h_{t-1}, x_t)$$

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$

$$y_t = W_{hy}h_t$$
Character-level language model example

Vocabulary: [h,e,l,o]

Example training sequence: “hello”
Character-level language model example

Vocabulary: [h,e,l,o]

Example training sequence: “hello”
Example

Character-level language model example

Vocabulary: [h,e,l,o]

Example training sequence: “hello”

\[ h_t = \tanh(W_{hh} h_{t-1} + W_{xh} x_t) \]
Character-level language model example

Vocabulary: [h,e,l,o]

Example training sequence: “hello”
The vanishing gradient problem

- The error at a time step ideally can tell a previous time step from many steps away to change during backprop
- But we’re multiplying together many values between 0 and 1

Adapted from Richard Socher
The vanishing gradient problem for language models

- In the case of language modeling or question answering words from time steps far away are not taken into consideration when training to predict the next word

- Example:

  Jane walked into the room. John walked in too. It was late in the day. Jane said hi to ____
Gated Recurrent Units (GRUs)

• More complex hidden unit computation in recurrence!

• Introduced by Cho et al. 2014

• Main ideas:
  • keep around memories to capture long distance dependencies
  • allow error messages to flow at different strengths depending on the inputs
Gated Recurrent Units (GRUs)

- Standard RNN computes hidden layer at next time step directly:
  \[ h_t = f \left( W^{(hh)} h_{t-1} + W^{(hx)} x_t \right) \]

- GRU first computes an update gate (another layer) based on current input word vector and hidden state:
  \[ z_t = \sigma \left( W^{(z)} x_t + U^{(z)} h_{t-1} \right) \]

- Compute reset gate similarly but with different weights:
  \[ r_t = \sigma \left( W^{(r)} x_t + U^{(r)} h_{t-1} \right) \]
Gated Recurrent Units (GRUs)

- Update gate
  \[ z_t = \sigma \left( W^{(z)} x_t + U^{(z)} h_{t-1} \right) \]

- Reset gate
  \[ r_t = \sigma \left( W^{(r)} x_t + U^{(r)} h_{t-1} \right) \]

- New memory content:
  \[ \tilde{h}_t = \tanh \left( W x_t + r_t \circ U h_{t-1} \right) \]
  If reset gate unit is \( \sim 0 \), then this ignores previous memory and only stores the new word information.

- Final memory at time step combines current and previous time steps:
  \[ h_t = z_t \circ h_{t-1} + (1 - z_t) \circ \tilde{h}_t \]
Gated Recurrent Units (GRUs)

\[
\begin{align*}
    z_t &= \sigma \left( W^{(z)} x_t + U^{(z)} h_{t-1} \right) \\
    r_t &= \sigma \left( W^{(r)} x_t + U^{(r)} h_{t-1} \right) \\
    \tilde{h}_t &= \text{tanh} \left( W x_t + r_t \circ U h_{t-1} \right) \\
    h_t &= z_t \circ h_{t-1} + (1 - z_t) \circ \tilde{h}_t
\end{align*}
\]
Gated Recurrent Units (GRUs)

- If reset is close to 0, ignore previous hidden state: Allows model to drop information that is irrelevant in the future
  \[
  z_t = \sigma \left( W^{(z)} x_t + U^{(z)} h_{t-1} \right) \\
  r_t = \sigma \left( W^{(r)} x_t + U^{(r)} h_{t-1} \right) \\
  \tilde{h}_t = \tanh \left( W x_t + r_t \circ U h_{t-1} \right) \\
  h_t = z_t \circ h_{t-1} + (1 - z_t) \circ \tilde{h}_t
  \]

- Update gate $z$ controls how much of past state should matter now
  - If $z$ close to 1, then we can copy information in that unit through many time steps! **Less vanishing gradient**!

- Units with short-term dependencies often have reset gates ($r$) very active; ones with long-term dependencies have active update gates ($z$)

Richard Socher
Long-short-term-memories (LSTMs)

- Proposed by Hochreiter and Schmidhuber in 1997

- We can make the units even more complex

- Allow each time step to modify
  - Input gate (current cell matters) \( i_t = \sigma \left( W^{(i)} x_t + U^{(i)} h_{t-1} \right) \)
  - Forget (gate 0, forget past) \( f_t = \sigma \left( W^{(f)} x_t + U^{(f)} h_{t-1} \right) \)
  - Output (how much cell is exposed) \( o_t = \sigma \left( W^{(o)} x_t + U^{(o)} h_{t-1} \right) \)
  - New memory cell \( \tilde{c}_t = \tanh \left( W^{(c)} x_t + U^{(c)} h_{t-1} \right) \)
  - Final memory cell: \( c_t = f_t \circ c_{t-1} + i_t \circ \tilde{c}_t \)
  - Final hidden state: \( h_t = o_t \circ \tanh(c_t) \)

Adapted from Richard Socher
Long-short-term-memories (LSTMs)

Intuition: memory cells can keep information intact, unless inputs makes them forget it or overwrite it with new input.

Cell can decide to output this information or just store it.

Richard Socher, figure from wildml.com
Generating poetry with RNNs

Sonnet 116 – Let me not ...

by William Shakespeare

Let me not to the marriage of true minds
Admit impediments. Love is not love
Which alters when it alteration finds,
Or bends with the remover to remove:
O no! it is an ever-fixed mark
That looks on tempests and is never shaken;
It is the star to every wandering bark,
Whose worth's unknown, although his height be taken.
Love's not Time's fool, though rosy lips and cheeks
Within his bending sickle's compass come:
Love alters not with his brief hours and weeks,
But bears it out even to the edge of doom.
If this be error and upon me proved,
I never writ, nor no man ever loved.
Generating poetry with RNNs

at first:

```
tyndt-iafhatawiaohrderomt lytdws e ,tfti, astai f ogoh eoase rrranbyne 'nhthnee e plia tkrlgd t o idoe ns,smtt h ne etie h,hregtrs nigtkie,aoaenns lng
```

train more

"Tmont thithey" fomesscerliund
Keushey. Thom here
sheulke, anmerenith ol sivh I lalterthend Bleipile shuwy fil on aseterlome coaniogenncc Phe lism thond hon at. MeiDimorotion in ther thize."

train more

Aftair fall unsuch that the hall for Prince Velzonski's that me of her hearly, and behs to so arwage fiving were to it beloge, pavu say falling misfort how, and Gogition is so overelical and ofter.

train more

"Why do what that day," replied Natasha, and wishing to himself the fact the princess, Princess Mary was easier, fed in had ofthened him.
Pierre aking his soul came to the packs and drove up his father-in-law women.

More info: [http://karpathy.github.io/2015/05/21/rnn-effectiveness/](http://karpathy.github.io/2015/05/21/rnn-effectiveness/)
Generating poetry with RNNs

PANDARUS:
Alas, I think he shall be come approached and the day
When little strain would be attain'd into being never fed,
And who is but a chain and subjects of his death,
I should not sleep.

Second Senator:
They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.

DUKE VINCENTIO:
Well, your wit is in the care of side and that.

Second Lord:
They would be ruled after this chamber, and
my fair nues begun out of the fact, to be conveyed,
Whose noble souls I’ll have the heart of the wars.

Clown:
Come, sir, I will make did behold your worship.

VIOLA:
I'll drink it.
Generating textbooks with RNNs

open source textbook on algebraic geometry

The Stacks Project

Browse chapters

<table>
<thead>
<tr>
<th>Part</th>
<th>Chapter</th>
<th>online</th>
<th>TeX source</th>
<th>view pdf</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preliminaries</td>
<td>1. Introduction</td>
<td>online</td>
<td>tex</td>
<td>pdf</td>
</tr>
<tr>
<td></td>
<td>2. Conventions</td>
<td>online</td>
<td>tex</td>
<td>pdf</td>
</tr>
<tr>
<td></td>
<td>3. Set Theory</td>
<td>online</td>
<td>tex</td>
<td>pdf</td>
</tr>
<tr>
<td></td>
<td>4. Categories</td>
<td>online</td>
<td>tex</td>
<td>pdf</td>
</tr>
<tr>
<td></td>
<td>5. Topology</td>
<td>online</td>
<td>tex</td>
<td>pdf</td>
</tr>
<tr>
<td></td>
<td>6. Sheaves on Spaces</td>
<td>online</td>
<td>tex</td>
<td>pdf</td>
</tr>
<tr>
<td></td>
<td>7. Sites and Sheaves</td>
<td>online</td>
<td>tex</td>
<td>pdf</td>
</tr>
<tr>
<td></td>
<td>8. Stacks</td>
<td>online</td>
<td>tex</td>
<td>pdf</td>
</tr>
<tr>
<td></td>
<td>9. Fields</td>
<td>online</td>
<td>tex</td>
<td>pdf</td>
</tr>
<tr>
<td></td>
<td>10. Commutative Algebra</td>
<td>online</td>
<td>tex</td>
<td>pdf</td>
</tr>
</tbody>
</table>

Parts

1. Preliminaries
2. Schemes
3. Topics in Scheme Theory
4. Algebraic Spaces
5. Topics in Geometry
6. Deformation Theory
7. Algebraic Stacks
8. Miscellany

Statistics

- The Stacks project now consists of
  - 455910 lines of code
  - 14221 tags (56 inactive tags)
  - 2366 sections

Latex source
Generating textbooks with RNNs

For $\bigoplus_{i=1}^{m} \mathcal{L}_{i}$ where $\mathcal{L}_{i}$ = 0, hence we can find a closed subset $\mathcal{H}$ in $\mathcal{H}$ and any sets $\mathcal{F}$ on $X$, $U$ is a closed immersion of $S$, then $U \to T$ is a separated algebraic space.

Proof. Proof of (1). It also start we get

$$S = \text{Spec}(R) = U \times X U \times X U$$

and the compareably in the fibre product covering we have to prove the lemma generated by $\prod \mathcal{I}_{i} \times U \to V$. Consider the maps $M$ along the set of points $\text{Sch}_{\text{fppf}}(U) \to U$ is the fibre category of $S$ in $U$ in Section, ?? and the fact that any $U$ affine, see Morphisms, Lemma ??, we find the scheme $\mathcal{S}$ and any open subset $W \subset U$ in $\text{Sh}(G)$ such that $\text{Spec}(R') \to S$ is smooth or an

$$U = \bigcup_{i \in I} U_{i} \times S_{i} \times U_{i}$$

which has a nonzero morphism may assume that $f_{i}$ is of finite presentation over $S$. We claim that $\mathcal{O}_{X_{s}}$ is a scheme where $x, x', s' \in S'$ such that $\mathcal{O}_{X_{s}} \to \mathcal{O}_{X_{s}, x'}$ is separated. By Algebra, Lemma ??, $S$ can define a map of complexes $\text{GL}_{S}(\mathcal{X}/S')$ and we win.

To prove study we see that $\mathcal{F}_{i} U_{i}$ is a covering of $X'$, and $T_{i}$ is an object of $\mathcal{F}_{X/S}$ for $i > 0$ and $\mathcal{F}_{T_{i}}$ exists and let $\mathcal{F}_{i}$ be a presheaf of $\mathcal{O}_{X}$-modules on $C$ as a $C$-module. In particular $\mathcal{F} = U/\mathcal{F}$ we have to show that

$$\mathcal{M} = \mathcal{M}_{i} \times \mathcal{O}_{\mathcal{S}_{i}, x}$$

is a unique morphism of algebraic stacks. Note that

Arrows = $(\text{Sch}/S)_{\text{fppf}}^{\text{op}}(\text{Sch}/S)_{\text{fpf}}$

and

$$V = \Gamma(S, \mathcal{O}) \hookrightarrow (U, \text{Spec}(A))$$

is an open subset of $X$. Thus $U$ is affine. This is a continuous map of $X$ is the inverse, the groupoid scheme $S$.

Proof. See discussion of sheaves of sets.

The result for any open covering follows from the less of Example ??, it may replace $S$ by $X_{\text{open, etale}}$ which gives an open subspace of $X$ and $T$ equal to $S_{\text{ Zar, etale}}$, see Descent, Lemma ??, Namely, by Lemma ??, we see that $R$ is geometrically regular over $S$.

Lemma 0.1. Assume (3) and (3) by the construction in the definition.

Suppose $X = \text{lim} \mathcal{X}_{i}$ (by the formal open covering $X$ and a single map $\text{Proj}_{X}(A) = \text{Spec}(B)$ over $U$ compatible with the complex

$$\mathcal{S}(A) = \Gamma(X, \mathcal{O}_{X, x})$$

When in this case of to show that $\mathcal{Q} \to \mathcal{C}_{Z/X}$ is stable under the following result in the second conditions of (1), and (3). This finishes the proof. By Definition ?? (without element is when the closed subschemes are etale), if $T$ is surjective we may assume that $T$ is connected with residue fields of $S$. Moreover there exists a closed subspace $Z \subset X$ of $X$ where $U$ in $X$ is proper (some defining as a closed subset the uniqueness it suffices to check the fact that the following theorem

1. $f$ is locally of finite type. Since $S = \text{Spec}(R)$ and $Y = \text{Spec}(R)$.

Proof. This is form all sheaves of sheaves on $X$. But given a scheme $U$ and a surjective étale morphism $U \to X$. Let $U \cap U = \bigcup_{i \in I} U_{i}$ be the scheme $X$ over $S$ at the schemes $X_{i} \to X$ and $U = \lim_{\to} X_{i}$.

The following lemma surjective decomposes of this implies that $\mathcal{F}_{X_{i}} = \mathcal{F}_{X_{i}, \text{et}}$.

Lemma 0.2. Let $X$ be a locally Noetherian scheme over $S$, $E = \mathcal{F}_{X/S}$. Set $\mathcal{I} = \mathcal{I}_{i} \subset \mathcal{I}_{i}$. Since $\mathcal{I} \subset \mathcal{I}_{n}$ are nonzero over $\mathcal{O}_{X}$ is a subset of $\mathcal{J}_{n,0} \circ A_{2}$ works.

Lemma 0.3. In Situation ??, Hence we may assume $f_{i} = 0$.

Proof. We will use the property we see that $p$ is the next functor (??). On the other hand, by Lemma ??, we see that

$$\mathcal{D}(\mathcal{O}_{X}) = \mathcal{O}_{X}(\mathcal{D})$$

where $K$ is an $F$-algebra where $\delta_{n+1}$ is a scheme over $S$. 

Andrej Karpathy
Generating textbooks with RNNs

Proof. Omitted.

Lemma 0.1. Let $\mathcal{C}$ be a set of the construction.
Let $\mathcal{C}$ be a gerber covering. Let $\mathcal{F}$ be a quasi-coherent sheaves of $\mathcal{O}$-modules. We have to show that

$$\mathcal{O}_{\mathcal{O}_X} = \mathcal{O}_X(\mathcal{L})$$

Proof. This is an algebraic space with the composition of sheaves $\mathcal{F}$ on $X_{\text{etale}}$ we have

$$\mathcal{O}_X(\mathcal{F}) = \{\text{morph}_{1} \times_{\mathcal{O}_X} (\mathcal{G}, \mathcal{F})\}$$

where $\mathcal{G}$ defines an isomorphism $\mathcal{F} \to \mathcal{G}$ of $\mathcal{O}$-modules.

Lemma 0.2. This is an integer $Z$ is injective.

Proof. See Spaces, Lemma ??.

Lemma 0.3. Let $S$ be a scheme. Let $X$ be a scheme and $X$ is an affine open covering. Let $U \subset X$ be a canonical and locally of finite type. Let $X$ be a scheme. Let $X$ be a scheme which is equal to the formal complex.

The following to the construction of the lemma follows.

Let $X$ be a scheme. Let $X$ be a scheme covering. Let

$$b : X \to Y' \to Y \to Y' \times_X Y \to X$$

be a morphism of algebraic spaces over $S$ and $Y$.

Proof. Let $X$ be a nonzero scheme of $X$. Let $X$ be an algebraic space. Let $\mathcal{F}$ be a quasi-coherent sheaf of $\mathcal{O}_X$-modules. The following are equivalent

1. $\mathcal{F}$ is an algebraic space over $S$.
2. If $X$ is an affine open covering.

Consider a common structure on $X$ and $X$ the functor $\mathcal{O}_X(U)$ which is locally of finite type.

This since $\mathcal{F} \in \mathcal{F}$ and $x \in \mathcal{G}$ the diagram

![Diagram](image)

is a limit. Then $\mathcal{G}$ is a finite type and assume $S$ is a flat and $\mathcal{F}$ and $\mathcal{G}$ is a finite type $f$. This is of finite type diagrams, and

- the composition of $\mathcal{G}$ is a regular sequence,
- $\mathcal{O}_X$ is a sheaf of rings.

Proof. We have see that $X = \text{Spec}(R)$ and $\mathcal{F}$ is a finite type representable by algebraic space. The property $\mathcal{F}$ is a finite morphism of algebraic stacks. Then the cohomology of $X$ is an open neighborhood of $U$.

Proof. This is clear that $\mathcal{G}$ is a finite presentation, see Lemmas ??.

A reduced above we conclude that $U$ is an open covering of $C$. The functor $\mathcal{F}$ is a sheaf.

$$\mathcal{O}_{X,s} \to \mathcal{F}_s \to \mathcal{O}_X(\mathcal{O}_{X,s})$$

is an isomorphism of covering of $\mathcal{O}_X$. If $\mathcal{F}$ is the unique element of $\mathcal{F}$ such that $X$ is an isomorphism.

The property $\mathcal{F}$ is a disjoint union of Proposition ?? and we can filtered set of presentations of a scheme $\mathcal{O}_X$-algebra with $\mathcal{F}$ are opens of finite type over $S$.

If $\mathcal{F}$ is a scheme theoretic image points.

If $\mathcal{F}$ is a finite direct sum $\mathcal{O}_{X,s}$ is a closed immersion, see Lemma ??, This is a sequence of $\mathcal{F}$ is a similar morphism.
Generating code with RNNs

```c
static void do_command(struct seq_file *m, void *v)
{
    int column = 32 << (cmd[2] & 0x80);
    if (state)
        cmd = (int)(int_state ^ (in_8(&ch->ch_flags) & Cmd) ? 2 : 1);
    else
        seq = 1;
    for (i = 0; i < 16; i++) {
        if (k & (1 << i))
            pipe = (in_use & UMXTTHREAD_UNCCA) +
            ((count & 0x00000000ffffff8) & 0x000000f) << 8;
        if (count == 0)
            sub(pid, ppc_md.kexec_handle, 0x20000000);
        pipe_set_bytes(i, 0);
    }
    /* Free our user pages pointer to place camera if all dash */
    subsystem_info = &of_changes[PAGE_SIZE];
    rek_controls(offset, idx, &offset);
    /* Now we want to deliberately put it to device */
    control_check_polarity(&context, val, 0);
    for (i = 0; i < COUNTER; i++)
        seq_puts(s, "policy ");
}```
Image Captioning

CVPR 2015:
Deep Visual-Semantic Alignments for Generating Image Descriptions, Karpathy and Fei-Fei
Show and Tell: A Neural Image Caption Generator, Vinyals et al.
Long-term Recurrent Convolutional Networks for Visual Recognition and Description, Donahue et al.
Learning a Recurrent Visual Representation for Image Caption Generation, Chen and Zitnick

Adapted from Andrej Karpathy
Image Captioning

Recurrent Neural Network

Convolutional Neural Network
Image Captioning

test image
Image Captioning

Andrej Karpathy
Image Captioning

before:
\[ h = \tanh(W_{xh} \cdot x + W_{hh} \cdot h) \]

now:
\[ h = \tanh(W_{xh} \cdot x + W_{hh} \cdot h + W_{ih} \cdot i) \]
Image Captioning

Andrej Karpathy
Image Captioning

Andrej Karpathy
Image Captioning

Andrej Karpathy
Image Captioning

Andrej Karpathy
Image Captioning

Caption generated: “straw hat”

Sample <END> token => finish.

Adapted from Andrej Karpathy
Image Captioning

"man in black shirt is playing guitar."
"construction worker in orange safety vest is working on road."
"two young girls are playing with lego toy."
"boy is doing backflip on wakeboard."
"a young boy is holding a baseball bat."
"a cat is sitting on a couch with a remote control."
"a woman holding a teddy bear in front of a mirror."
"a horse is standing in the middle of a road."

Andrej Karpathy
Plan for this lecture

• Image captioning
  – Examples
  – Tool: Recurrent neural networks
  – Captioning for video
  – Diversifying captions
  – Visual-semantic spaces

• Visual question answering
A monkey pulls a dog’s tail and is chased by the dog.
Key Insight:
Generate feature representation of the video and “decode” it to a sentence

Venugopalan et al., “Translating Videos to Natural Language using Deep Recurrent Neural Networks”, NAACL-HTL 2015
Video Captioning

Input Video → Sample frames @1/10

Forward propagate
Output: “fc7” features
(activations before classification layer)

fc7: 4096 dimension
“feature vector”
Video Captioning

Input Video

Convolutional Net

Recurrent Net

Output

Mean across all frames

A

boy

is

playing

golf

<EOS>

Venugopalan et al., “Translating Videos to Natural Language using Deep Recurrent Neural Networks”, NAACL-HTL 2015
Annotated video data is scarce.

Key Insight:
Use supervised pre-training on data-rich auxiliary tasks and transfer.

Venugopalan et al., “Translating Videos to Natural Language using Deep Recurrent Neural Networks”, NAACL-HTL 2015
Video Captioning

**CNN pre-training**

- Caffe Reference Net - variation of Alexnet [Krizhevsky et al. NIPS’12]
- 1.2M+ images from ImageNet ILSVRC-12 [Russakovsky et al.]
- Initialize weights of our network.

Venugopalan et al., “Translating Videos to Natural Language using Deep Recurrent Neural Networks”, NAACL-HTL 2015
Video Captioning

**Image-Caption pre-training**

Venugopalan et al., “Translating Videos to Natural Language using Deep Recurrent Neural Networks”, NAACL-HTL 2015
Video Captioning

Fine-tuning

1. Video dataset
2. Mean pooled feature
3. Lower learning rate

Venugopalan et al., “Translating Videos to Natural Language using Deep Recurrent Neural Networks”, NAACL-HTL 2015
Video Captioning

- A man appears to be plowing a rice field with a plow being pulled by two oxen.
- A man is plowing a mud field.
- Domesticated livestock are helping a man plow.
- A man leads a team of oxen down a muddy path.
- A man is plowing with some oxen.
- A man is tilling his land with an ox pulled plow.
- Bulls are pulling an object.
- Two oxen are plowing a field.
- The farmer is tilling the soil.
- A man in ploughing the field.

- A man is walking on a rope.
- A man is walking across a rope.
- A man is balancing on a rope.
- A man is balancing on a rope at the beach.
- A man walks on a tightrope at the beach.
- A man is balancing on a volleyball net.
- A man is walking on a rope held by poles
- A man balanced on a wire.
- The man is balancing on the wire.
- A man is walking on a rope.
- A man is standing in the sea shore.

Venugopalan et al., “Translating Videos to Natural Language using Deep Recurrent Neural Networks”, NAACL-HTL 2015
Video Captioning

MT metrics (BLEU, METEOR) to compare the system generated sentences against (all) ground truth references.

<table>
<thead>
<tr>
<th>Model</th>
<th>BLEU</th>
<th>METEOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best Prior Work</td>
<td>13.68</td>
<td>23.90</td>
</tr>
<tr>
<td>[Thomason et al. COLING’14]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Only Images</td>
<td>12.66</td>
<td>20.96</td>
</tr>
<tr>
<td>Only Video</td>
<td>31.19</td>
<td>26.87</td>
</tr>
<tr>
<td>Images+Video</td>
<td>33.29</td>
<td>29.07</td>
</tr>
</tbody>
</table>

Pre-training only, no fine-tuning
No pre-training
Video Captioning

FGM: A person is dancing with the person on the stage.
YT: A group of men are riding the forest.
I+V: A group of people are dancing.
GT: Many men and women are dancing in the street.

FGM: A person is cutting a potato in the kitchen.
YT: A man is slicing a tomato.
I+V: A man is slicing a carrot.
GT: A man is slicing carrots.

FGM: A person is walking with a person in the forest.
YT: A monkey is walking.
I+V: A bear is eating a tree.
GT: Two bear cubs are digging into dirt and plant matter at the base of a tree.

FGM: A person is riding a horse on the stage.
YT: A group of playing are playing in the ball.
I+V: A basketball player is playing.
GT: Dwayne wade does a fancy layup in an allstar game.

Venugopalan et al., “Translating Videos to Natural Language using Deep Recurrent Neural Networks”, NAACL-HTL 2015
Video Captioning

[Encode] RNN encoder  [RNN decoder] [French Sentence] [Sutskever et al. NIPS’14]

[Encode] RNN encoder  [RNN decoder] [Sentence] [Donahue et al. CVPR’15] [Vinyals et al. CVPR’15]

[Encode] RNN encoder  [RNN decoder] [Sentence] [Venugopalan et al. NAACL’15]

[Encode] RNN encoder  [RNN decoder] [Sentence] [Venugopalan et al. ICCV’15] (this work)

Venugopalan et al., “Sequence to Sequence - Video to Text”, ICCV 2015
Video Captioning

S2VT Overview

Encoding stage

Decoding stage

Venugopalan et al., “Sequence to Sequence - Video to Text”, ICCV 2015
Visual Description

Berkeley LRCN [Donahue et al. CVPR'15]:
A brown bear standing on top of a lush green field.

MSR CaptionBot [http://captionbot.ai/]:
A large brown bear walking through a forest.

MSCOCO
80 classes

Venugopalan et al., “Captioning Images With Diverse Objects”, CVPR 2017
Object Recognition

Can identify hundreds of categories of objects.

IMAGENET 14M images, 22K classes [Deng et al. CVPR'09]

Venugopalan et al., “Captioning Images With Diverse Objects”, CVPR 2017
Novel Object Captioner (NOC)

We present Novel Object Captioner which can compose descriptions of 100s of objects in context.

NOC (ours): Describe novel objects without paired image-caption data.

OKAPI

Existing captioners. Visual Classifiers.

MSCOCO

IMAGENET

> A horse standing in the dirt.

An okapi standing in the middle of a field.

Venugopalan et al., “Captioning Images With Diverse Objects”, CVPR 2017
Insights

1. Need to recognize and describe objects outside of image-caption datasets.

Venugopalan et al., “Captioning Images With Diverse Objects”, CVPR 2017
Insight 1: Train effectively on external sources

Visual features from unpaired image data

Language model from unannotated text data

Venugopalan et al., “Captioning Images With Diverse Objects”, CVPR 2017
Insights

2. Describe unseen objects that are similar to objects seen in image-caption datasets.

Venugopalan et al., “Captioning Images With Diverse Objects”, CVPR 2017
Insight 2: Capture semantic similarity of words

Image-Specific Loss

Embed

CNN

Text-Specific Loss

$W^T_{\text{glove}}$

Embed

LSTM

$W_{\text{glove}}$

Venugopalan et al., “Captioning Images With Diverse Objects”, CVPR 2017
Insight 2: Capture semantic similarity of words

Venugopalan et al., “Captioning Images With Diverse Objects”, CVPR 2017
Combine to form a Caption Model

Not different from existing caption models. Problem: Forgetting.

[Catastrophic Forgetting in Neural Networks. Kirkpatrick et al. PNAS 2017]

Venugopalan et al., “Captioning Images With Diverse Objects”, CVPR 2017
Insight 3: Jointly train on multiple sources

Venugopalan et al., “Captioning Images With Diverse Objects”, CVPR 2017
Qualitative Evaluation: ImageNet

**Instruments**
- A man holding a **banjo** in a park.
- A large **chime** hanging on a metal pole.

**Land Animals**
- A **okapi** is in the grass with a **okapi**.
- A small brown and white **jackal** is standing in a field.

**Vehicles**
- A **snowplow** truck driving down a snowy road.
- A group of people standing around a large white **warship**.

**Household**
- A large metal **candelabra** next to a wall.
- A black and white photo of a **corkscrew** and a **corkscrew**.

Venugopalan et al., “Captioning Images With Diverse Objects”, CVPR 2017
Qualitative Evaluation: ImageNet

**Birds**
- A small **pheasant** is standing in a field.
- A **osprey** flying over a large grassy area.

**Outdoors**
- A large **glacier** with a mountain in the background.
- A group of people are sitting in a **baobab**.

**Water Animals**
- A **humpback** is flying over a large body of water.
- A man is standing on a beach holding a **snapper**.

**Misc**
- A table with a **cauldron** in the dark.
- A woman is posing for a picture with a **chiffon** dress.

Venugopalan et al., “Captioning Images With Diverse Objects”, CVPR 2017
Visual-semantic space

\[ \sum_{i}^{n} \left[ \| f(x_i^a) - f(x_i^p) \|_2^2 - \| f(x_i^a) - f(x_i^n) \|_2^2 + \alpha \right]_+ \]

a denotes anchor
p denotes positive
n denotes negative

Kiros et al., “Unifying visual-semantic embeddings with multimodal neural language models”, TACL 2015
I should buy this drink because it’s exciting.
Visual-semantic space for understanding ads

Region proposal and attention weighing

Knowledge inference and symbol embedding

Triplet training

“"I should be careful on the road so I don’t crash and die.”"

“I should buy this bike because it’s fast.”

Ye and Kovashka, “ADVISE: Symbolism and External Knowledge for Decoding Advertisements”, in submission to ECCV 2018
Visual-semantic space for understanding ads

VSE++: “I should try this makeup because it’s fun.”

Hussain-ranking: “I should stop smoking because it destroys your looks.”

ADVISE (ours): “I should be careful to how I treat Earth because when the water leaves we die.”

VSE++: “I should wear Nivea because it leaves no traces.”

Hussain-ranking: “I should be eating these because it has fresh ingredients.”

ADVISE (ours): “I should buy GeoPack paper because the their cutlery is eco-friendly.”
Plan for this lecture

• Image captioning
  – Examples
  – Tool: Recurrent neural networks
  – Captioning for video
  – Diversifying captions
  – Visual-semantic spaces

• Visual question answering
Visual Question Answering (VQA)

Task: Given an image and a natural language open-ended question, generate a natural language answer.

Visual Question Answering (VQA)

Image Embedding

Convolution Layer + Non-Linearity
Pooling Layer
Convolution Layer + Non-Linearity
Pooling Layer
Fully-Connected

4096-dim

Neural Network Softmax over top K answers

Input (Features II) Softmax classifier

P(y = 0 | x)
P(y = 1 | x)
P(y = 2 | x)

Question Embedding

“How many horses are in this image?”

LSTM

1024-dim

Visual Question Answering (VQA)

Figure 2. Our proposed framework: given an image, a CNN is first applied to produce the attribute-based representation $V_{att}(I)$. The internal textual representation is made up of image captions generated based on the image-attributes. The hidden state of the caption-LSTM after it has generated the last word in each caption is used as its vector representation. These vectors are then aggregated as $V_{cap}(I)$ with average-pooling. The external knowledge is mined from the KB (in this case DBpedia) and the responses encoded by Doc2Vec, which produces a vector $V_{know}(I)$. The 3 vectors $V$ are combined into a single representation of scene content, which is input to the VQA LSTM model which interprets the question and generates an answer.

Wu et al., "Ask Me Anything: Free-Form Visual Question Answering Based on Knowledge From External Sources", CVPR 2016
Visual Question Answering (VQA)

Ask any question about this image

Reasoning for VQA

Is there a red shape above a circle?

red
exists
above

→

→ true

→

yes

Andreas et al., “Neural Module Networks”, CVPR 2016
Reasoning for VQA

**Question:** Are there more cubes than yellow things?  
**Answer:** Yes

Program Generator

- Are
  - things
    - LSTM
    - LSTM
  - yellow
    - LSTM
    - LSTM
  - than
    - LSTM
    - LSTM
  - cubes
    - LSTM
    - LSTM
  - more
    - LSTM
    - LSTM
  - there
    - LSTM
    - LSTM

Predicted Program

- greater than
  - count
    - filter color [yellow]
    - <SCENE>
    - count
      - filter color [yellow]
      - filter shape [cube]

Execution Engine

- greater_than
  - count
  - count
    - filter color [yellow]
    - filter shape [cube]

CNN

Johnson et al., “Inferring and Executing Programs for Visual Reasoning”, ICCV 2017