Toward
Text-to-Picture Synthesis

Andrew B. Goldberg, Jake Rosin, Xiaojin Zhu, Charles Dyer

Department of Computer Sciences
University of Wisconsin-Madison
Augmentative & Alternative Communication (AAC)
Augmentative & Alternative Communication (AAC)

* Fact: More than 2 million people in the U.S. cannot rely on natural speech alone for communication

* One solution: AAC software for pictorial communication

* Existing systems transliterate words into icons
Augmentative & Alternative Communication (AAC)

* Fact: More than 2 million people in the U.S. cannot rely on natural speech alone for communication

* One solution: AAC software for pictorial communication

* Existing systems transliterate words into icons
Augmentative & Alternative Communication (AAC)

* Fact: More than 2 million people in the U.S. cannot rely on natural speech alone for communication

* One solution: AAC software for pictorial communication

* Existing systems transliterate words into icons

Jane saw orchids and treefrogs in the rainforest
Augmentative & Alternative Communication (AAC)

• Fact: More than 2 million people in the U.S. cannot rely on natural speech alone for communication

• One solution: AAC software for pictorial communication

• Existing systems transliterate words into icons

Jane saw orchids and treefrogs in the rainforest

• Users must be trained to recognize specialized symbols
Text-to-Picture Synthesis

Goal: Convert from text to image modalities
Text-to-Picture Synthesis

Goal: Convert from text to image modalities

Diagram showing the relationship between Generality and Comprehensibility with Ideal point.
Text-to-Picture Synthesis

Goal: Convert from text to image modalities

Rebus symbols (e.g., widgit.com)
Text-to-Picture Synthesis

Goal: Convert from text to image modalities

Rebus symbols (e.g., widgit.com)

Generality

Comprehensibility

Ideal

CarSim (Johansson et al, IJCAI 05)
Text-to-Picture Synthesis

Goal: Convert from text to image modalities

Rebus symbols (e.g., widgit.com)

WordsEye (wordseye.com, Coyne & Sproat, SIGGRAPH 01)

CarSim (Johansson et al, IJCAI 05)
Text-to-Picture Synthesis

Goal: Convert from text to image modalities

Rebus symbols (e.g., widgit.com)

Our approach

Ideal

WordsEye (wordseye.com, Coyne & Sproat, SIGGRAPH 01)

CarSim (Johansson et al, IJCAI 05)

(Zhu et al, AAAI 07) (Goldberg et al, CoNLL 08)
Main TTP Components
Main TTP Components

- Keyphrase extraction
  - TextRank with picturability
  - Semantic role labeling
Main TTP Components

- Keyphrase extraction
  - TextRank with picturability
  - Semantic role labeling
- Image selection
  - Search result clustering
  - Context-sensitive re-ranking
Main TTP Components

- Keyphrase extraction
  - TextRank with picturability
  - Semantic role labeling
- Image selection
  - Search result clustering
  - Context-sensitive re-ranking
- Layout optimization
  - Structured output prediction
  - Heuristic objective minimization
Example Machine Learning Problem #1: Picture-Driven Keyphrase Extraction

* Given: English text string

  The Bayesian statistician ate a banana.

* Do: Extract a set of words to be depicted visually

  \{\text{statistician, ate, banana}\}
Example Machine Learning Problem #1: Picture-Driven Keyphrase Extraction

* Given: English text string
  
  The Bayesian statistician ate a banana.

* Do: Extract a set of words to be depicted visually
  
  \{statistician, ate, banana\}

Approach in Zhu et al, AAAI 07:

**TextRank**: Teleporting random walk (like PageRank) on a word co-occurrence graph [Mihalcea & Tarau 04]

**Picturability**: Bias teleporting to easy-to-visualize words
Annotation instructions: Imagine you're playing Pictionary...
Label $y=1$ if you can draw or find a good image of the word.
Label $y=0$ if you don't think this word has a picture.
Annotation instructions: Imagine you're playing Pictionary...
Label $y=1$ if you can draw or find a good image of the word. Label $y=0$ if you don't think this word has a picture.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>writ</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>yolks</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>zebras</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>zigzag</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Five annotators independently judged 500 words each.
Predicting Word Picturability
Predicting Word Picturability

- How can we automatically predict which words are easy to draw or visualize?
Predicting Word Picturability

* How can we automatically predict which words are easy to draw or visualize? Use the Web!
Predicting Word Picturability

• How can we automatically predict which words are easy to draw or visualize? Use the Web!

• Logistic regression model based on Web statistics:
  • Features: log-ratios of various search result counts
  • For fast prediction, used single feature chosen by CV:
    \[ x = \log(\text{Google image hits} / \text{Google page hits}) \]
  • Final model: \( \Pr(y = 1|x) = \frac{1}{1 + \exp(-2.78x - 15.4)} \)
Predicting Word Picturability

* How can we automatically predict which words are easy to draw or visualize? **Use the Web!**

The Bayesian statistician ate a banana.

Bayesian  17K image hits, 10.4M page hits :  \( \Pr(y = 1|x) = 0.09 \)

banana  356K image hits, 49.4M page hits :  \( \Pr(y = 1|x) = 0.84 \)

* Final model:  \( \Pr(y = 1|x) = \frac{1}{1 + \exp(-2.78x - 15.4)} \)
Example Machine Learning Problem #2: Semantically Enhanced Layout

- Given: Set of images representing keywords
- Do: Arrange images to help elicit desired interpretation
Example Machine Learning Problem #2: Semantically Enhanced Layout

- Given: Set of images representing keywords
- Do: Arrange images to help elicit desired interpretation

Approach in **Goldberg et al., CoNLL 08**: 

**ABC Template**: Three “semantic” boxes and action arrow

A: “who” (~subject)  
B: “did what / how / when” (~verb, adv)  
C: “to what” (~object)

**Structured output prediction**:  
Fill template by tagging words in input sequence.
Collecting ABC Pictures

- Used Web-based tool to create over 500 ABC pictures

- Great crowdsourcing / human computing potential

Fleiss’ $\kappa = 0.71$ for 48 layouts by 3 people
Layout Prediction using CRFs

* Given: Text sequence $\mathbf{x}$ (e.g., words, chunks)
  Features: semantic role labels, POS, WordNet supersenses, ...

* Do: Predict layout-position sequence $\mathbf{y}$, $y_t \in \{A, B, C, O\}$

<table>
<thead>
<tr>
<th>The girl</th>
<th>ARG0, DT, NN, n.person</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>rides the bus</td>
<td>Verb, ARG1, VBZ, DT, NN, v.transport, n.vehicle</td>
<td>B</td>
</tr>
<tr>
<td>to</td>
<td>TO</td>
<td>O</td>
</tr>
<tr>
<td>school</td>
<td>ARGM-LOC, NN, n.building</td>
<td>C</td>
</tr>
<tr>
<td>in the morning</td>
<td>ARGM-TMP, IN, DT, NN, n.time</td>
<td>B</td>
</tr>
</tbody>
</table>
Layout Prediction using CRFs

* Given: Text sequence $\mathbf{x}$ (e.g., words, chunks)
  Features: semantic role labels, POS, WordNet supersenses, ...

* Do: Predict layout-position sequence $\mathbf{y}$, $y_t \in \{A, B, C, O\}$

<table>
<thead>
<tr>
<th>The girl</th>
<th>ARG0, DT, NN, n.person</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>rides the bus</td>
<td>Verb, ARG1, VBZ, DT, NN, v.transport, n.vehicle</td>
<td>B</td>
</tr>
<tr>
<td>to</td>
<td></td>
<td></td>
</tr>
<tr>
<td>school</td>
<td></td>
<td></td>
</tr>
<tr>
<td>in the morning</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

A: The girl
B: rides the bus to school in the morning
C: in the morning

$A$, $B$, $C$
Layout Prediction using CRFs

* Given:  Text sequence $\mathbf{x}$ (e.g., words, chunks)
  Features: semantic role labels, POS, WordNet supersenses, ...

* Do: Predict layout-position sequence $\mathbf{y}$, $y_t \in \{A, B, C, O\}$

The girl  ARG0, DT, NN, n.person  A
rides the bus  Verb, ARG1, VBZ, DT, NN, v.transport, n.vehicle  B
to
school
in the morning

Conditional Random Field (CRF)

$$Pr(y|x) \propto \exp \left( \sum_{t=1}^{\left|\mathbf{x}\right|} \sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, x, t) \right)$$

Selected model order and feature functions via CV on 500+ training examples
The Future

- **Text extraction:**
  - Picture-driven keyphrase extraction

- **Image selection:**
  - Prototypical image selection
  - Context-based image search
  - Image sense disambiguation

- **Layout prediction:**
  - Higher-order, template-free layout prediction
  - Visual semantic role labeling with verb cartoons
Thank you

and

NSF IIS-0711887
Wisconsin Alumni Research Foundation
Yahoo! Key Technical Challenges Grant

Any questions?

Andrew B. Goldberg
goldberg@cs.wisc.edu