Machine Reading, models and applications

Julien Perez
Machine Learning and Optimization

5th March, 2018
Developing the building blocks of AI
- Models
- Algorithms
- Tasks

Areas of expertise in ML
- Spectral / Deep learning
- Probabilistic graphical models
- Reinforcement learning
- Adversarial learning

Areas of expertise in Optimization
- Large scale combinatorial
- Large scale continuous
- Convex optimization
- Stochastic optimization
1. Machine reading
2. Dialog state tracking
3. End-to-end dialog learning
"A machine that produces machine operable representations of texts", **BUT**

- **Fixed/Predefined ontology**
- **Fixed/Predefined lexical domain**
- **Data duplication by triplification**

Question Answering (QA) on Knowledge Base

Large-scale knowledge graphs
- Properties of billions of entities
- Plus relations among them

An QA Example:

**Question:** what is Obama’s citizenship?

- Query parsing:
  
  (Obama, Citizenship, ?)

- Identify and infer over relevant subgraphs:
  
  (Obama, BornIn, Hawaii)
  (Hawaii, PartOf, USA)

- correlating semantically relevant relations:
  
  BornIn ~ Citizenship

**Answer:** USA
Towards the **Machine Comprehension** of Text

Towards the Machine Comprehension of Text: An Essay

Christopher J.C. Burges
Microsoft Research
One Microsoft Way
Redmond, WA 98052, USA

December 23, 2013
Machine Reading

Definition

“A machine comprehends a passage of text if, for any question regarding that text, it can be answered correctly by a majority of native speakers.

The machine needs to provide a string which human readers would agree both

1. Answers that question
2. Does not contain information irrelevant to that question." (Burges, 2013)

Applications

- Information extraction from collection of documents
- Social media mining
- End-to-End Dialog systems
James was always getting in trouble. His aunt Jane tried as hard as she could to keep him out of trouble, but he was sneaky and got into lots of trouble behind her back. He went to the grocery store and pulled all the pudding off the shelves and ate two jars. Then he walked to the fast food restaurant and ordered 15 bags of fries. He didn't pay, and instead headed home.

**Question:** Where did James go after he went to the grocery store?

- his deck
- his freezer
- a fast food restaurant
- his home

**Goals:**

- Language independent
- Ontology free
- Grammar agnostic
- Robust

Machine Reading
as Multi-choice question task

- **MCTest** (Richardson et al. 2013): 500 passages
  2000 questions about simple stories

- **RACE** (Lai et al. 2017): 28,000 passages
  100,000 questions from English comprehension tests

Formalization

\[
p(a | q, d) = s_\theta(a, q, d) = \sigma(V_a \cdot \beta (V_q, V_d)^T)
\]

\[
L(\theta) = \sum_{i=1}^{D} \sum_{a \in A} \max \{0, m - s_\theta(a^+, q, d) + s(a^-, q, d)\}
\]
Machine Reading
as Span selection

- **SQuAD** (Rajpurkar et al. 2016): 500 passages 100,000 questions on Wikipedia text

- **TriviaQA** (Joshi et al. 2017): 95k questions, 650k evidence documents (distant supervision)

**Formalization**

\[
p_s(w | q, d) = B_\theta(w, q, d)
\]

\[
p_e(w | q, d) = E_\theta(w, q, d)
\]

\[
L(\theta) = - \sum_{w \in T} p(w).\log p_\theta(w)
\]
Machine Reading
as Cloze style queries

Document
The BBC producer allegedly struck by Jeremy Clarkson will not press charges against the "Top Gear" host, his lawyer said Friday. Clarkson, who hosted one of the most-watched television shows in the world, was dropped by the BBC. Wednesday after an internal investigation by the British broadcaster found he had subjected producer Oisin Tymon to an unprovoked physical and verbal attack."

Question: Producer X will not press charges against Jeremy Clarkson, his lawyer says.

Answer: Oisin Tymon

Formalization
\[ p(a | q, d) = s_\theta(a, q, d) = \sigma(V_a \cdot \beta(V_q, V_d)^T) \]
\[ L(\theta) = \sum_{i=1}^{D} \sum_{a \in A} \max \{0, m - s_\theta(a^+, q, d) + s(a^-, q, d)\} \]

Machine Reading

Datasets

Before 2015:
- MCTest (Richardson et al, 2013): 2600 questions
- ProcessBank (Berant et al, 2014): 500 questions

After 2015:
- CNN/Daily Mail
- Children Book Test
- WikiReading
- LAMBADA
- SQuAD
- Who did What
- Maluuba
- NewsQA
- MS MARCO
- NAVER

More than 100k questions!
### Machine Reading

#### Datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Company</th>
<th>Query Source</th>
<th>Answer</th>
<th># Queries</th>
<th># Docs</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC Test</td>
<td>Microsoft</td>
<td>Crowdsourced</td>
<td>Multiple Choice</td>
<td>2640</td>
<td>660</td>
</tr>
<tr>
<td>WikiQA</td>
<td>Microsoft</td>
<td>User Logs</td>
<td>Sentence Selection</td>
<td>3047</td>
<td>29K sentences</td>
</tr>
<tr>
<td>CNN/DailyMail</td>
<td>DeepMind</td>
<td>Cloze</td>
<td>Fill in entity</td>
<td>1.4M</td>
<td>93K CNN, 220K DM</td>
</tr>
<tr>
<td>Children’s Book</td>
<td>Facebook</td>
<td>Cloze</td>
<td>Fill in the word</td>
<td>688K</td>
<td>688K contexts</td>
</tr>
<tr>
<td>SQuAD</td>
<td>Stanford</td>
<td>Crowdsourced</td>
<td>Span</td>
<td>100K</td>
<td>536</td>
</tr>
<tr>
<td>News QA</td>
<td>Maluuba</td>
<td>Crowdsourced</td>
<td>Span</td>
<td>120K</td>
<td>12K</td>
</tr>
<tr>
<td>MS MARCO</td>
<td>Microsoft</td>
<td>User Logs</td>
<td>Human Synthesized</td>
<td>100k</td>
<td>1M passages, 200K+ docs</td>
</tr>
</tbody>
</table>
Machine Reading Models

Before 2015:
- Lexical matching
- Logistic regression

After 2015:
- Attentive Reader
- Memory Networks
- Gated-attention Reader
- ReasoNet
- Match-LSTM
- Attention Sum Reader
- Attention-over-Attention Reader
- Iterative Attentive Reader
- Dynamic coattention networks
- Bi-directional Attention Flow Network
- Multi-Perspective Context Matching
- ...
Microsoft, Alibaba AI programs beat humans in a Stanford reading test

January 19, 2018 by Song Lee, The Mercury News

AI models beat humans at reading comprehension, but they’ve still got a ways to go

Alibaba’s AI outperforms humans in one of the toughest reading comprehension tests ever created in a remarkable world first

- AI was created by retail firm Alibaba’s Institute of Data Science and Technologies
- It took part in Stanford Question Answering Dataset reading comprehension test
- It scored 82.44 in the exact answer category beating the human score of 82.304
- The company said it’s the first time a machine has out-done a real person

Alibaba and Microsoft AI beat humans in Stanford reading test

Top-place tie by tech groups provides symbol of race between US and China
Machine Reading

Building block – Recurrent Neural Network

LSTM with a forget gate

Compact form of the equations for the forward pass of a LSTM unit with a forget gate:[1][2]

\[ f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \]
\[ i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \]
\[ o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \]
\[ c_t = f_t \odot c_{t-1} + i_t \odot \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \]
\[ h_t = o_t \odot \sigma_h(c_t) \]

where the initial values are \( c_0 = 0 \) and \( h_0 = 0 \) and the operator \( \odot \) denotes the Hadamard product (entry-wise product).

Variables

- \( x_t \in \mathbb{R}^d \): input vector to the LSTM unit
- \( f_t \in \mathbb{R}^h \): forget gate's activation vector
- \( i_t \in \mathbb{R}^h \): input gate's activation vector
- \( o_t \in \mathbb{R}^h \): output gate's activation vector
- \( h_t \in \mathbb{R}^h \): output vector of the LSTM unit
- \( c_t \in \mathbb{R}^h \): cell state vector
- \( W \in \mathbb{R}^{h \times d}, U \in \mathbb{R}^{h \times h} \) and \( b \in \mathbb{R}^h \): weight matrices and bias vector parameters

Machine Reading

Attention Sum Reader Network

\[
\begin{align*}
\text{Document} & \\
\text{Input text} & \ldots \text{Obama} \quad \text{and} \quad \text{Putin} \quad \ldots \quad \text{said} \quad \text{Obama} \quad \text{in} \quad \text{Prague} \\
\text{Embeddings} & \quad e(\text{Obama}) \quad e(\text{and}) \quad e(\text{Putin}) \quad \ldots \quad e(\text{said}) \quad e(\text{Obama}) \quad e(\text{in}) \quad e(\text{Prague}) \\
\text{Recurrent neural networks} & \quad \vdots \\
\text{Dot products} & \\
\text{Softmax } s_i \text{ over words in the document} & \\
\text{Probability of the answer} & P(\text{Obama}|q, d) = \sum_{i \in \{\text{Obama}, d\}} s_i = s_j + s_{j+5} \\
\text{Question} & XXXXX \quad \text{visited} \quad \text{Prague} \\
\end{align*}
\]

\[
\begin{align}
\begin{align*}
s_i & \propto \exp(f_i(d) \cdot g(q)) \\
P(w|q, d) & \propto \sum_{i \in I(w, d)} s_i
\end{align*}
\end{align}
\]

where \(I(w, d)\) is a set of positions where \(w\) appears in the document \(d\).

\[
\begin{align*}
f_i(d) &= \overrightarrow{f_i}(d) \parallel \overleftarrow{f_i}(d), \\
g(q) &= \overrightarrow{g}(q) \parallel \overleftarrow{g}(q).
\end{align*}
\]

We employ a Deep LSTM cell with skip connections,

\[
x'(t, k) = x(t) || y'(t, k-1),
\]
\[
i(t, k) = \sigma (W_{ki} x'(t, k) + W_{hi} h(t-1, k) + W_{ci} c(t-1, k) + b_{ki}),
\]
\[
f(t, k) = \sigma (W_{kf} x(t) + W_{hf} h(t-1, k) + W_{cf} c(t-1, k) + b_{kf}),
\]
\[
c(t, k) = f(t, k) c(t-1, k) + i(t, k) \tanh (W_{kc} x'(t, k) + W_{hc} h(t-1, k) + b_{kc}),
\]
\[
o(t, k) = \sigma (W_{ko} x'(t, k) + W_{ho} h(t-1, k) + W_{co} c(t, k) + b_{ko}),
\]
\[
h(t, k) = o(t, k) \tanh (c(t, k)),
\]
\[
y'(t, k) = W_{ky} h(t, k) + b_{ky},
\]
\[
y(t) = y'(t, 1) || \ldots || y'(t, K),
\]

where \(||\) indicates vector concatenation \(h(t, k)\) is the hidden state for layer \(k\) at time \(t\), and \(i, f, o\) are the input, forget, and output gates respectively.

\[
g^{LSTM}(d, q) = y(|d| + |q|)
\]

with input \(x(t)\) the concatenation of \(d\) and \(q\) separated by the delimiter \(||||\).
Machine Reading
Deep Long Short Term Memory readers

Denote the outputs of a bidirectional LSTM as $\hat{y}(t)$ and $\hat{y}(t)$. Form two encodings, one for the query and one for each token in the document,

$$u = \hat{y}_q(|q|) \parallel \hat{y}_q(1), \quad y_d(t) = \hat{y}_d(t) \parallel \hat{y}_d(t).$$

The representation $r$ of the document $d$ is formed by a weighted sum of the token vectors. The weights are interpreted as the model’s attention,

$$m(t) = \tanh (W_{ym} y_d(t) + W_{um} u),$$
$$s(t) \propto \exp (w_{ms}^T m(t)),$$
$$r = y_d s.$$

Define the joint document and query embedding via a non-linear combination:

$$g^{AR}(d, q) = \tanh (W_{rg} r + W_{ug} u).$$

Machine Reading

R-Net

- Extractive model
- Fully differentiable
- Based on 4 stacked layers
- Language independent


Figure 1: R-NET structure overview.
## Performances

### SQuAD and MsMARCO

<table>
<thead>
<tr>
<th>Model Description</th>
<th>Dev Set</th>
<th>Test Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single model</td>
<td>EM / F1</td>
<td>EM / F1</td>
</tr>
<tr>
<td>LR Baseline</td>
<td>40.0 / 51.0</td>
<td>40.4 / 51.0</td>
</tr>
<tr>
<td>Dynamic Chunk Reader (Yu et al. 2016)</td>
<td>62.5 / 71.2</td>
<td>62.5 / 71.0</td>
</tr>
<tr>
<td>Attentive CNN context with LSTM (NLPR, CASIA)</td>
<td>- / -</td>
<td>63.3 / 73.5</td>
</tr>
<tr>
<td>Match-LSTM with Ans-Ptr (Wang &amp; Jiang 2016b)</td>
<td>64.1 / 73.9</td>
<td>64.7 / 73.7</td>
</tr>
<tr>
<td>Dynamic Coattention Networks (Xiong et al. 2016)</td>
<td>65.4 / 75.6</td>
<td>66.2 / 75.9</td>
</tr>
<tr>
<td>Iterative Coattention Network (Fudan University)</td>
<td>- / -</td>
<td>67.5 / 76.8</td>
</tr>
<tr>
<td>FastQA (Weissenborn et al. 2017)</td>
<td>- / -</td>
<td>68.4 / 77.1</td>
</tr>
<tr>
<td>BiDAF (Seo et al. 2016)</td>
<td>68.0 / 77.3</td>
<td>68.0 / 77.3</td>
</tr>
<tr>
<td>T-gating (Peking University)</td>
<td>- / -</td>
<td>68.1 / 77.6</td>
</tr>
<tr>
<td>RaSoR (Lee et al. 2016)</td>
<td>- / -</td>
<td>69.6 / 77.7</td>
</tr>
<tr>
<td>SEDT+BiDAF (Liu et al. 2017)</td>
<td>- / -</td>
<td>68.5 / 78.0</td>
</tr>
<tr>
<td>Multi-Perspective Matching (Wang et al. 2016)</td>
<td>- / -</td>
<td>70.4 / 78.8</td>
</tr>
<tr>
<td>FastQAExt (Weissenborn et al. 2017)</td>
<td>- / -</td>
<td>70.8 / 78.9</td>
</tr>
<tr>
<td>Mnemonic Reader (NUDT &amp; Fudan University)</td>
<td>- / -</td>
<td>69.9 / 79.2</td>
</tr>
<tr>
<td>Document Reader (Chen et al. 2017)</td>
<td>- / -</td>
<td>70.7 / 79.4</td>
</tr>
<tr>
<td>ReasoNet (Shen et al. 2016)</td>
<td>- / -</td>
<td>70.6 / 79.4</td>
</tr>
<tr>
<td>Ruminating Reader (Gong &amp; Bowman 2017)</td>
<td>- / -</td>
<td>70.6 / 79.5</td>
</tr>
<tr>
<td>jNet (Zhang et al. 2017)</td>
<td>- / -</td>
<td>70.6 / 79.8</td>
</tr>
<tr>
<td>Interactive AoA Reader (Joint Laboratory of HIT and iFLYTEK Research)</td>
<td>- / -</td>
<td>71.2 / 79.9</td>
</tr>
<tr>
<td>R-NET (Wang et al. 2017)</td>
<td>71.1 / 79.5</td>
<td>71.3 / 79.7</td>
</tr>
<tr>
<td>R-NET (March 2017)</td>
<td>72.3 / 80.6</td>
<td>72.3 / 80.7</td>
</tr>
</tbody>
</table>
Machine Reading
Competent statistical NLP

Featured Logistic Regression
• Whether \( e \) is in the passage
• Whether \( e \) is in the question
• Frequency of \( e \) in passage
• First position of \( e \) in passage
• n-gram exact match
• Syntactic dependency around \( e \)

- The required reasoning and inference level is can be limited
- There isn’t much room left for improvement
- However, the scale and ease of data production is appealing
- Not yet proven whether NNs can do more challenging RC tasks

Memory Networks

• Class of models that combine large memory with learning component that can read and write to it.

• Most ML has limited memory which is more-or-less all that’s needed for “low level” tasks e.g. object detection.

• Incorporates reasoning with attention over memory (RAM).
Beyond structure extraction

- Much of this information comes in the form of unstructured text which cannot easily be searched, mined, visualized or, ultimately, acted upon.

- Textual data can specify reasoning capabilities

- **Goal**: build machines that can "understand" textual information, *i.e.* converting it into interpretable structured knowledge to be leveraged by humans and other machines alike.

- **Reasoning capability is a frontier of current ML approaches**

End-to-End Memory Networks

Annotation scheme
• Input sentences $x_1, x_2, \ldots, x_n$
• Query $q$
• Answer $a$

Model performs by
• Generating memories from inputs
• Transforming query into suitable representation
• Process query and memories jointly using multiple hops to produce the answer
• Backpropagate through the whole procedure

Joe went to the kitchen.
Fred went to the kitchen.
Joe picked up the milk.
Joe travelled to his office.
Joe left the milk.
Joe went to the bathroom.

Where is the milk now?

Office
End-to-End Memory Network for dialog

Description

Model

\[ m_i = A\Phi(x_i) \quad u = B\Phi(q) \]
\[ c_i = C\Phi(x_i) \]
\[ p_i = \text{softmax}(u^T m_i) \]
\[ o = \sum_i p_i c_i \]
\[ u^{k+1} = o^k + u^k \]
\[ \hat{a} = \text{softmax}(u^T W' \Phi(y_1), \ldots, u^T W' \Phi(y_{|C|})) \]

Optimization task

- Categorical cross-entropy
- Stochastic Gradient Descent with clipping
- Grid-searched Hyper Parameters
Gated End-to-End Memory Network

\[ m_i = A\Phi(x_i) \quad u = B\Phi(q) \]
\[ c_i = C\Phi(x_i) \]
\[ p_i = \text{softmax}(u^\top m_i) \]
\[ o = \sum_i p_ic_i \]
\[ T^k(u^k) = \sigma(W^k_Tu^k + b^k_T) \]
\[ u^{k+1} = o^k \odot T^k(u^k) + u^k \odot (1 - T^k(u^k)) \]
\[ \hat{a} = \text{softmax}(u^\top W'\Phi(y_1), \ldots, u^\top W'\Phi(y_{|C|})) \]

gated controller update

Properties
- End-to-End memory access regulation
- Close to Highway Network and Residual Network

[8] Fei Liu and Julien Perez, Gated End-to-End Memory Network, EACL 2017
End to End Memory Networks

- Addressing = **Soft attention**
- requires explicit supervision of attention during training
- Only feasible for simple tasks
Memory Module

- Attention weights / Soft address
- Memory vectors
- \( \{ \vec{m}_1, \vec{m}_2, \ldots, \vec{m}_N \} \)
- \( \{ p_1, p_2, \ldots, p_N \} \)
- Softmax
- Dot Product
- \( \sum_i p_i \vec{m}_i \)
- To controller (added to controller state)
- Addressing signal (controller state vector)
## 20 bAbi tasks: Benchmark results

<table>
<thead>
<tr>
<th>Task</th>
<th>Baseline</th>
<th>MemN2N</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LSTM</td>
<td>MemNN</td>
</tr>
<tr>
<td>1: 1 supporting fact</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>2: 2 supporting facts</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>3: 3 supporting facts</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>4: 2 argument relations</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>5: 3 argument relations</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>6: yes/no questions</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>7: counting</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>8: lists/sets</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>9: simple negation</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>10: indefinite knowledge</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>11: basic coreference</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>12: conjunction</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>13: compound coreference</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>14: time reasoning</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>15: basic deduction</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>16: basic induction</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>17: positional reasoning</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>18: size reasoning</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>19: path finding</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>20: agent’s motivation</td>
<td>0.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

### Table 1: Test error rates (%) on the 20 QA tasks for models using 1k training examples (mean test errors for 10k training examples are shown at the bottom). Key: BoW = bag-of-words representation; PE = position encoding representation; LS = linear start training; RN = random injection of time index noise; LW = RNN-style layer-wise weight tying (if not stated, adjacent weight tying is used); joint = joint training on all tasks (as opposed to per-task training).
Gated End-to-End Memory Network

further works

<table>
<thead>
<tr>
<th></th>
<th>Laptop</th>
<th>Restaurant</th>
</tr>
</thead>
<tbody>
<tr>
<td>Majority</td>
<td>53.45</td>
<td>65.00</td>
</tr>
<tr>
<td>Feature+ SVM</td>
<td><strong>72.10</strong></td>
<td><strong>80.89</strong></td>
</tr>
<tr>
<td>LSTM</td>
<td>66.45</td>
<td>74.28</td>
</tr>
<tr>
<td>TDLSTM</td>
<td>68.13</td>
<td>75.63</td>
</tr>
<tr>
<td>TDLSTM+ATT</td>
<td>66.24</td>
<td>74.31</td>
</tr>
<tr>
<td>ContextAVG</td>
<td>61.22</td>
<td>71.33</td>
</tr>
<tr>
<td>MemNet (1)</td>
<td>67.66</td>
<td>76.10</td>
</tr>
<tr>
<td>MemNet (2)</td>
<td>71.14</td>
<td>78.61</td>
</tr>
<tr>
<td>MemNet (3)</td>
<td>71.74</td>
<td>79.06</td>
</tr>
<tr>
<td>MemNet (4)</td>
<td>72.21</td>
<td>79.87</td>
</tr>
<tr>
<td>MemNet (5)</td>
<td>71.89</td>
<td>80.14</td>
</tr>
<tr>
<td>MemNet (6)</td>
<td>72.21</td>
<td>80.05</td>
</tr>
<tr>
<td>MemNet (7)</td>
<td>72.37</td>
<td>80.32</td>
</tr>
<tr>
<td>MemNet (8)</td>
<td>72.05</td>
<td>80.14</td>
</tr>
<tr>
<td>MemNet (9)</td>
<td>72.21</td>
<td><strong>80.95</strong></td>
</tr>
</tbody>
</table>

Table 2: Classification accuracy of different methods on laptop and restaurant datasets. Best scores in each group are in bold.

Open Questions
Open-Domain QA

Q: How many of Warsaw's inhabitants spoke Polish in 1933?

[Image showing a flowchart with a question (Q), a document retriever, and a document reader, leading to an answer (833,500).]

Figure 1: An overview of our question answering system DrQA.

Open Questions
Multi document reasoning

- Most Reading Comprehension methods limit themselves to queries which can be answered using a single sentence, paragraph, or document.

- Enabling models to combine disjoint pieces of textual evidence would extend the scope of machine comprehension.

- Text understanding across multiple documents and to investigate the limits of existing methods.

Open Questions
Adversarial Examples

- Add a sentence or word string specifically designed to distract the model
- Drops accuracy of state-of-the-art models from 81 to 46
- Current issue of deep models, already observed on image tasks

1. Machine reading
2. Dialog state tracking
3. End-to-end dialog learning
Dialog systems design

Modularity is the current solution
• Divide and Conquer approach
• Annotation processes are required
• Hand-crafted models, case-by-case adaptation

End-to-End opportunities
• Leveraging raw dialogs
• Can be (automatically) enriched with meta-data
• Seamless integration of back-end access

Promising research context
• (Deep) imitation learning
• Non-Markovian control tasks
Components of a Dialog System

User

Words

My iphone can't access the internet
inform(device=?),
inform(symptom=no-internet-557)

NLU

Dialog Acts

What kind of iphone do you have?
request(device)

NLG

Dialog Manager

Knowledge
Dialog State tracking

Examples

<table>
<thead>
<tr>
<th>Utterance</th>
<th>Food</th>
</tr>
</thead>
<tbody>
<tr>
<td>S Hello, How may I help you?</td>
<td></td>
</tr>
<tr>
<td>U I need a <strong>Persian</strong> restaurant in the south part of town.</td>
<td>0.2 Persian</td>
</tr>
<tr>
<td>S What kind of food would you like?</td>
<td></td>
</tr>
<tr>
<td>U <strong>Persian</strong>.</td>
<td>0.8 Persian</td>
</tr>
<tr>
<td>S I'm sorry but there is no restaurant serving persian food.</td>
<td></td>
</tr>
<tr>
<td>U How about <strong>Portuguese</strong> food?</td>
<td>0.4 Persian</td>
</tr>
<tr>
<td>S Are you looking for Portuguese food?</td>
<td>0.6 Portuguese</td>
</tr>
<tr>
<td>U Yes.</td>
<td>0.1 Persian</td>
</tr>
<tr>
<td>S <strong>Nandos</strong> is a nice place in the south of town serving tasty Portuguese food.</td>
<td>0.9 Portuguese</td>
</tr>
</tbody>
</table>

Informable slots in DSTC3 (Tourist Information Domain)

<table>
<thead>
<tr>
<th>Slot</th>
<th>User may give as a constraint?</th>
</tr>
</thead>
<tbody>
<tr>
<td>area</td>
<td>Yes, 15 possible values</td>
</tr>
<tr>
<td>children allowed</td>
<td>Yes, 2 possible values</td>
</tr>
<tr>
<td>food</td>
<td>Yes, 28 possible values</td>
</tr>
<tr>
<td>has internet</td>
<td>Yes, 2 possible values</td>
</tr>
<tr>
<td>has tv</td>
<td>Yes, 2 possible values</td>
</tr>
<tr>
<td>name</td>
<td>Yes, 163 possible values</td>
</tr>
<tr>
<td>near</td>
<td>Yes, 52 possible values</td>
</tr>
<tr>
<td>pricerange</td>
<td>Yes, 4 possible values</td>
</tr>
<tr>
<td>type</td>
<td>Yes, 3 possible values (restaurant, pub, coffee-shop)</td>
</tr>
<tr>
<td>addr</td>
<td>No</td>
</tr>
<tr>
<td>phone</td>
<td>No</td>
</tr>
<tr>
<td>postcode</td>
<td>No</td>
</tr>
<tr>
<td>price</td>
<td>No</td>
</tr>
</tbody>
</table>

Informable slots in DSTC2 (Restaurant Information Domain)

<table>
<thead>
<tr>
<th>Slot</th>
<th>User may give as a constraint?</th>
</tr>
</thead>
<tbody>
<tr>
<td>area</td>
<td>Yes, 5 possible values</td>
</tr>
<tr>
<td>food</td>
<td>Yes, 91 possible values</td>
</tr>
<tr>
<td>name</td>
<td>Yes, 113 possible values</td>
</tr>
<tr>
<td>pricerange</td>
<td>Yes, 3 possible values</td>
</tr>
<tr>
<td>addr</td>
<td>No</td>
</tr>
<tr>
<td>phone</td>
<td>No</td>
</tr>
<tr>
<td>postcode</td>
<td>No</td>
</tr>
<tr>
<td>signature</td>
<td>No</td>
</tr>
</tbody>
</table>
Dialogue State Tracking
State of the art

Generative
- \{Factorial\} HMM
- Particle Filter

Discriminative
- Rule-based
- CRF/Max Entropy
- Deep Neural Network

Figure 1: The Neural Network structure for computing $E(s, v) \in \mathbb{R}$ for each possible value $v$ in the set $S_t, s$. The vector $f$ is a concatenation of all the input nodes.

Dialog State Tracking
Open Challenges

1. Longer context
2. Looser supervision schema
3. Reasoning capability
4. Minimize intermediary reps
   - Fixed Ontology
   - Fixed KB

Good Morning, how can I help you
I need a car for March 10th to go to Paris
Ok, I’m checking this
and find me a cheap hotel for the day after
(-_-) “
## Dialog State Tracking

Machine reading approach

<table>
<thead>
<tr>
<th>Index</th>
<th>Actor</th>
<th>Utterance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Cust</td>
<td>I'm looking for a cheap restaurant in the west or east part of town.</td>
</tr>
<tr>
<td>2</td>
<td>Agent</td>
<td>Thanh Binh is a nice restaurant in the west of town in the cheap price range.</td>
</tr>
<tr>
<td>3</td>
<td>Cust</td>
<td>What is the address and post code.</td>
</tr>
<tr>
<td>4</td>
<td>Agent</td>
<td>Thanh Binh is on magdalene street city centre.</td>
</tr>
<tr>
<td>5</td>
<td>Cust</td>
<td>Thank you goodbye.</td>
</tr>
<tr>
<td>6</td>
<td>Factoid Question</td>
<td>What is the pricerange? Answer: {Cheap}</td>
</tr>
<tr>
<td>7</td>
<td>Yes/No Question</td>
<td>Is the Pricerange Expensive? Answer: {No}</td>
</tr>
<tr>
<td>8</td>
<td>Indefinite Knowledge</td>
<td>Is the FoodType chinese? Answer: {Maybe}</td>
</tr>
<tr>
<td>8</td>
<td>Listing task</td>
<td>What are the areas? Answer: {West, East}</td>
</tr>
</tbody>
</table>

Table 1. State tracking as machine reading task

What is the Pricerange?

Input story

1: Hi, how can I help you?
2: I'm looking for a cheap restaurant in the north of town
3: Do you have a preference for the type?

End-to-End Memory Network
Dialog State tracking

Memory Module

\[ 0.1 \vec{m}_1 + 0.7 \vec{m}_2 + 0.2 \vec{m}_3 \]

\{ \vec{m}_1, \vec{m}_2, \vec{m}_3 \}

Dot product + softmax

Weighted Sum

\{0.1, 0.7, 0.2\}

Controller

\[ \vec{u}_1 \rightarrow \vec{u}_2 \]

Answer

Cheap

What is the Pricerange?

End - to - End Memory Network
End-to-End Memory Network
Results on DSTC-2 – Goal Tracking and Reasoning


<table>
<thead>
<tr>
<th>Variable</th>
<th>d</th>
<th>Yes-No</th>
<th>I.K.</th>
<th>Count.</th>
<th>List.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Food</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td>0.85</td>
<td>0.79</td>
<td>0.89</td>
<td>0.41</td>
</tr>
<tr>
<td>40</td>
<td></td>
<td>0.83</td>
<td>0.84</td>
<td>0.88</td>
<td>0.42</td>
</tr>
<tr>
<td>60</td>
<td></td>
<td>0.82</td>
<td>0.82</td>
<td>0.90</td>
<td>0.39</td>
</tr>
<tr>
<td>Area</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td>0.86</td>
<td>0.83</td>
<td>0.94</td>
<td>0.79</td>
</tr>
<tr>
<td>40</td>
<td></td>
<td>0.90</td>
<td>0.89</td>
<td>0.96</td>
<td>0.75</td>
</tr>
<tr>
<td>60</td>
<td></td>
<td>0.88</td>
<td>0.90</td>
<td>0.95</td>
<td>0.78</td>
</tr>
<tr>
<td>PriceRange</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td>0.93</td>
<td>0.86</td>
<td>0.93</td>
<td>0.83</td>
</tr>
<tr>
<td>40</td>
<td></td>
<td>0.92</td>
<td>0.85</td>
<td>0.90</td>
<td>0.80</td>
</tr>
<tr>
<td>60</td>
<td></td>
<td>0.91</td>
<td>0.85</td>
<td>0.91</td>
<td>0.81</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model</th>
<th>Area</th>
<th>Food</th>
<th>Price</th>
<th>Joint</th>
</tr>
</thead>
<tbody>
<tr>
<td>RNN - no dict.</td>
<td>0.92</td>
<td>0.86</td>
<td>0.86</td>
<td>0.69</td>
</tr>
<tr>
<td>RNN + sem. dict.</td>
<td>0.91</td>
<td>0.86</td>
<td>0.93</td>
<td>0.73</td>
</tr>
<tr>
<td>NBT-DNN</td>
<td>0.90</td>
<td>0.84</td>
<td>0.94</td>
<td>0.72</td>
</tr>
<tr>
<td>NBT-CNN</td>
<td>0.90</td>
<td>0.83</td>
<td>0.93</td>
<td>0.72</td>
</tr>
<tr>
<td>MemN2N(d = 40)</td>
<td>0.89</td>
<td>0.88</td>
<td>0.95</td>
<td>0.74</td>
</tr>
</tbody>
</table>
End-to-End Memory Network
Dialog State tracking

On “one supporting fact” task (DSTC-2 dataset): 83% acc vs 79% for the sota.

Table 11: Attention shifting example for the PriceRange slot from DSTC2 dataset

<table>
<thead>
<tr>
<th>Actor</th>
<th>Utterance</th>
<th>Hop 1</th>
<th>Hop 2</th>
<th>Hop 3</th>
<th>Hop 4</th>
<th>Hop 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cust</td>
<td>I'm looking for a cheap restaurant that serves chinese food</td>
<td>0.00</td>
<td>0.14</td>
<td>0.01</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Agent</td>
<td>What part of town do you have in mind</td>
<td>0.02</td>
<td>0.17</td>
<td>0.05</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Cust</td>
<td>I dont care</td>
<td>0.00</td>
<td>0.00</td>
<td>0.14</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Agent</td>
<td>Rice house serves chinese food in the cheap price range</td>
<td>0.00</td>
<td>0.02</td>
<td>0.03</td>
<td>0.98</td>
<td>1.00</td>
</tr>
<tr>
<td>Cust</td>
<td>What is the address and telephone number</td>
<td>0.57</td>
<td>0.07</td>
<td>0.15</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Agent</td>
<td>Sure rice house is on mill road city centre</td>
<td>0.03</td>
<td>0.01</td>
<td>0.13</td>
<td>0.02</td>
<td>0.00</td>
</tr>
<tr>
<td>Cust</td>
<td>Phone number</td>
<td>0.00</td>
<td>0.01</td>
<td>0.03</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Agent</td>
<td>The phone number of rice house is 765-239-09</td>
<td>0.37</td>
<td>0.58</td>
<td>0.45</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Cust</td>
<td>Thank you good bye</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

What is the pricerange? Answer: cheap

[9] Julien Perez and Fei Liu, Dialog State Tracking, a machine reading approach, EACL 2017
1. Machine reading
2. Dialog state tracking
3. End-to-end dialog learning
Learning dialog from dialogs
- Simulated dialogs
- Emphasise each step of transaction
- Include some common learning challenges

Goal oriented dialog
- Backed with a Knowledge Base
- KB interactions are included in the decision set

A testbed for deep learning
- End-to-End learnable and flexible
- Attention with Non-parametric memory
- KB-fact and utterance support of the decision
- Dialog learning as Machine Reading

End-to-End Dialog learning
Dialog System and Technology Challenge 6th - Task 1

Organization
• Task 1: Issuing API calls.
• Task 2: Updating API calls.
• Task 3: Displaying options.
• Task 4: Providing extra information.
• Task 5: Conducting full dialogs.

Corpora
• 2 corpus with/without OOV
• 2 corpus with a new slot
• 2 Knowledge Bases

Objectives
• Emphasise challenges of real world transactional dialog
• Compare the models and learning algorithms

Systems and results

Decision models
• Hybrid Code Networks [1]
• Recurrent Entity Networks [2]
• (Dynamic) Memory Networks [3,4]
• LSTMs [5]
• Quantitized Language Model

Losses
• Categorical Cross-Entropy
• Ranking loss over similarity measure

Entity/Slot resolution strategies
• Dictionary and Heuristics
• Dedicated models (CRF, LSTMs)
• Delexicalization

Optimizers
• Adam SGD
• Gradient clipping
• Early stopping strategy

Conclusion

Toward automation of repetitive cognitive tasks

Lack of theoretical analysis
- Optimization algorithm convergence
- Nature of the loss surface w.r.t parameters
- Learnability//Safety

Limitation of learning procedures
- Active//Interactive Learning
- Curriculum learning
- Regularization strategies
Thank you

europe.naverlabs.com