Semantic decoding in dialogue systems

Milica Gašić

Dialogue Systems Group, Cambridge University Engineering Department

January 31, 2017
In this lecture...

Dialogue acts

Semantic decoding as a classification task

Input features to semantic decoder

Semantic decoding as a sequence to sequence learning task
Architecture of a statistical dialogue system
Problem

Decoding meaning in utterances:

- *Do they serve Korean food*
- *Can you repeat that please*
- *Hi I want to find a restaurant that serves Italian food*
- *How about a restaurant that serves Lebanese food*
- *I want a different restaurant*
- *Is it near Union Square*
- *May I have the address*
- *No, I want an expensive restaurant*
Reminder: Dialogue acts

Semantic concepts:

dialogue act type - encodes the system or the user intention in a (part of) dialogue turn

semantic slots and values - further describe entities from the ontology that a dialogue turn refers to

Is there um maybe a cheap place in the centre of town please?

inform ( price = cheap, area = centre)

dialogue act type

semantics slots and values
Semantic decoding

- Do they serve Korean food
- Can you repeat that please
- Hi I want to find an Italian restaurant
- I want a different restaurant
- Is it near Union Square
- May I have the address
- No, I want an expensive restaurant
- How about a restaurant that serves Lebanese food
- confirm(food=Korean)
- repeat()
- hello(type=restaurant, food=Italian)
- reqalts()
- confirm(near=Union Square)
- request(addr)
- negate(type=restaurant, pricerange=expensive)
- reqalts(type=restaurant, food=Lebanise)
Semantic decoding

- Dialogue utterances labelled with semantic concepts
- The set of semantic concepts
Semantic decoding as a classification task

- Data: Dialogue utterances labelled with semantic concepts
- Model: Support vector machines
- Predictions: The set of semantic concepts
Semantic decoding as a classification task

Is there um maybe a cheap place in the centre of town please?

Dialogue act types:
- classes:
  - negate
  - deny
  - inform
  - select

Slot value pairs:
- food=Italian
- food=Chinese
- area=centre
- area=north
- price=cheap
Theory: support vector machines

- Support vector machine is a maximum margin classifier
- Support vectors are input data points that lie on the margin
- Input data points are mapped into a high dimensional feature space where the data is linearly separable.

\[ \mathbf{x} \rightarrow \phi(\mathbf{x}) \]

- Kernel function is the dot product of feature functions

\[ k(\mathbf{x}, \mathbf{x}) = \phi(\mathbf{x})^T \phi(\mathbf{x}) \]
The decision surface is given by

\[ f(x) = \sum_{i=1}^{n} y_i \alpha_i k(x, x_i) + \beta \]

- \( x \)  test data point
- \( x_i \)  support vectors
- \( y_i \)  labels, \( y_i \in \{1, -1\} \)
- \( \alpha_i \)  weight of the support vector in the feature space
- \( \beta \)  bias
- \( k(\cdot, \cdot) \)  kernel function
Extended to multiclass SVM using *one-versus-rest* approach

The output of an SVM is transformed into probability by fitting a sigmoid

$$p(y = 1|x) = \frac{1}{1 + \exp(a f(x) + b)}$$

and estimating $a, b$ by maximum likelihood on a validation set.
Input to semantic decoder

top ASR hypothesis Features are extracted directly from top hypothesis and the classification is performed into relevant semantic classes.
Ontology and Delexicalisation

I’m looking for an Italian restaurant.
I’m looking for an Indian restaurant.
I’m looking for a Chinese restaurant.

Delexicalise

I’m looking for an <tagged-food-value> restaurant.
I’m looking for a <tagged-food-value> restaurant.
SVMs in semantic decoding: semantic tuple classifier

Italian restaurant please

<tagged-food-value> restaurant please

Delexicalise

Count N-grams

SVM for food=<tagged-food-value>
SVM for area=<tagged-area-value>
SVM for price=<tagged-price-value>

Query SVMs

multi-class SVM for dialog act
inform() 0.9
request() 0.1
inform(food=Italian) 0.85
request() 0.15

Ontology

Produce valid dialogue acts and renormalise distributions
Input to semantic decoder

N-best list of ASR hypothesis  In real conversational systems error rate of the top hypothesis is typically 20-30%. To achieve robustness alternative hypotheses are needed.
Taking alternative ASR hypotheses into account

<table>
<thead>
<tr>
<th>Question</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is there an expensive restaurant?</td>
<td>0.35</td>
</tr>
<tr>
<td>Is there an inexpensive restaurant?</td>
<td>0.30</td>
</tr>
<tr>
<td>Inexpensive restaurant?</td>
<td>0.20</td>
</tr>
<tr>
<td>In expensive restaurant?</td>
<td>0.05</td>
</tr>
</tbody>
</table>

Semantic Decoding

<table>
<thead>
<tr>
<th>Action</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>inform(type=restaurant, price=expensive)</td>
<td>0.35</td>
</tr>
<tr>
<td>inform(type=restaurant, price=inexpensive)</td>
<td>0.30</td>
</tr>
<tr>
<td>inform(type=restaurant, price=inexpensive)</td>
<td>0.20</td>
</tr>
<tr>
<td>inform(type=restaurant, price=expensive)</td>
<td>0.05</td>
</tr>
</tbody>
</table>

Combine outputs

<table>
<thead>
<tr>
<th>Action</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>inform(type=restaurant, price=inexpensive)</td>
<td>0.50</td>
</tr>
<tr>
<td>inform(type=restaurant, price=expensive)</td>
<td>0.40</td>
</tr>
</tbody>
</table>
Input to semantic decoder

Word confusion network summarises the posterior distribution of ASR better, without pruning low probability words. Each arc in the word confusion network has a posterior probability for that word. That is the sum of all paths which contain that word at around that approximate time.

Context features can be extracted from the last system action. The user response may be dependent on the system question. Caution! We still want to understand utterances where the user is not following the system.
Evaluate the quality of semantic decoder

**F-score**  
\[ F = \frac{2|C_{ref} \cap C_{hyp_0}|}{|C_{ref}| + |C_{hyp_0}|} \]

**Item level cross entropy (ICE)**  
[Thomson et al., 2008] measures the quality of the output distribution \( p_i \) for every concept

\[ ICE = \frac{1}{1 + |C_{ref}|} \sum_{c \in C} \log(p(c)p^*(c) + (1-p(c))(1-p^*(c)))) \]

where \( p(c) = \sum_i p_i(c) \), \( c \in C_{hyp_i} \) and

\[ p^*(c) = \begin{cases} 1 & c \in C_{ref} \\ 0 & \text{otherwise} \end{cases} \]
Results [Henderson et al., 2012]

- Cambridge Restaurant Information Domain
- Semantic concepts area, price-range, food-type, phone number, post-code, signature dish, address of restaurant and dialogue act types: inform, request, confirm etc
- Data collected in car WER 37.0%

<table>
<thead>
<tr>
<th>Input</th>
<th>F-Score</th>
<th>ICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top ASR hypothesis</td>
<td>0.692 ± 0.012</td>
<td>1.790 ± 0.065</td>
</tr>
<tr>
<td>N-best ASR hypotheses</td>
<td>0.708 ± 0.012</td>
<td>1.760 ± 0.074</td>
</tr>
<tr>
<td>Confusion network</td>
<td>0.730 ± 0.011</td>
<td>1.680 ± 0.063</td>
</tr>
<tr>
<td>Confusion network + context</td>
<td>0.767 ± 0.011</td>
<td>0.880 ± 0.063</td>
</tr>
</tbody>
</table>
Semantic decoding as a sequence to sequence learning task

- Reads the input word by word, or window of words
- Outputs sequence of concepts using BIO labelling (begin, inside, other)
- These are then heuristically mapped into slot-value pairs

Is there um maybe a cheap place in the centre of town serving Chinese food please?

Heuristics

price=cheap  area=centre  food=Chinese
Semantic decoding as a sequence to sequence learning task

Data
- Dialogue utterances with semantic concepts

Model
- Conditional random fields

Predictions
- The sequence of semantic concepts
Theory: Conditional random fields

\[ p(y|x) = \frac{1}{Z(x)} \exp \left( \sum_i \lambda_i f_i(x, y) \right) \]

This is a fully connected undirected graph where:

- **x**: input sequence \((x_0, \cdots, x_n)\)
- **y**: output sequence \((y_0, \cdots, y_n)\)
- **\(f_i\)**: given feature functions
- **\(\lambda_i\)**: parameters to be estimated
Theory: Linear chain conditional random field

In this case the graph is not fully connected any more, the label at time step \( t \) depends on the label in the previous time step \( t - 1 \).

\[
p(y|x) = \prod_t \frac{1}{Z(x)} \exp \left( \sum_i \lambda_i f_i(x, y_t, y_{t-1}) \right) \tag{1}
\]

\[
= \frac{1}{Z(x)} \exp(\lambda^T F(x, y)) \tag{2}
\]
Training a linear chain conditional random field

- Maximise the log probability $\log p(y|x)$ with respect to parameters $\lambda$.

- It can be shown that the gradient of the log probability is the difference between the feature function values and the expected feature function values:

$$\nabla_{\lambda} \mathcal{L} = F(x, y) - \sum_{y'} p(y'|x) F(y', x).$$

- Since the label at each time step only depends on the label in the previous time step, message passing can be used to find the expectation.
Input data  Word confusion networks where each bin is annotated with semantic concept

Features  For each bin in the confusion network extract N-grams of the neighbouring bins and weight them by their confidence scores.

Task is conversational understanding system with real users about movies (22 concepts)

<table>
<thead>
<tr>
<th>Input</th>
<th>F-Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top ASR hypothesis</td>
<td>0.77</td>
</tr>
<tr>
<td>Confusion network</td>
<td>0.83</td>
</tr>
</tbody>
</table>
Semantic decoding as a sequence to sequence learning task

- Data
  - Dialogue utterances with semantic concepts
- Model
  - Recurrent neural networks
- Predictions
  - The sequence of semantic concepts
Theory: Neural networks

Neural network transforms input vector $\mathbf{x}$ into an output categorical probability distribution $\mathbf{y}$:

$$
\mathbf{h}_0 = g_0(\mathbf{W}_0 \mathbf{x}^T + \mathbf{b}_0)
$$

$$
\mathbf{h}_i = g_i(\mathbf{W}_i \mathbf{h}_{i-1}^T + \mathbf{b}_i), \quad 0 < i < m
$$

$$
\mathbf{y} = \text{softmax}(\mathbf{W}_m \mathbf{h}_{m-1}^T + \mathbf{b}_m)
$$

$$
\text{softmax}(\mathbf{h})_i = \frac{\exp(h_i)}{\sum_j \exp(h_j)}
$$

where

$g_i$ (differentiable) activation functions hyperbolic tangent tanh or sigmoid $\sigma$

$\mathbf{W}_i, \mathbf{b}_i$ parameters to be estimated
Theory: Neural networks

Neural network structure

- Input x
- Weights w
- Hidden nodes h
- Output y
- Weights w
- Input x
- Output y
- Hidden nodes h
Theory: Training neural networks

Cost function is the negative log probability of true label
\[ -\sum_j y_{ij} \log y'_{ij} \]

- \( y_i \) is delta distribution (zero everywhere except for the correct category)

- \( y'_i \) is the probability distribution estimated by a Neural network

The cost function can be minimised by stochastic gradient descent.
Neural networks for semantic decoding

Example network which does not take into account context.

I’m looking for a cheap restaurant

I’m looking for a <tagged-price-value> restaurant
Recurrent neural networks are deep neural networks unrolled through time. Elman-type neural network has recurrent connections between hidden layers of the neural networks:
Recurrent neural networks – Jordan-type

Jordan-type neural network feeds the output of previous time step into the next time step:
RNNs in semantic decoding [Mesnil et al., 2015]

ATIS dataset - flight booking information.

- I want to fly from Boston to New York.

Input features 1-hot representation or context window.

<table>
<thead>
<tr>
<th>F-score</th>
<th>Elman</th>
<th>Jordan</th>
<th>CRF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-hot</td>
<td>0.932</td>
<td>0.652</td>
<td>0.67</td>
</tr>
<tr>
<td>window</td>
<td>0.950</td>
<td>0.942</td>
<td>0.929</td>
</tr>
</tbody>
</table>

F-score on entertainment dataset

<table>
<thead>
<tr>
<th>CRF</th>
<th>RNN</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.906</td>
<td>0.881</td>
</tr>
</tbody>
</table>
Long short-term memory neural networks

RNNs automatically learn context information but suffer from vanishing gradient problem. Long short-term memory neural networks are an alternative model which to some extent avoid this problem and have been successfully used in semantic decoding [Yao et al., 2014].
Summary

Input
- Input can be 1-best or N-best list from the ASR or a confusion network.
- Taking into account alternative recognition result improves robustness.

Model
- Semantic decoding can be defined as a classification task.
- In this case a collection of SVMs can be used.
- Semantic decoding can be more naturally defined as a sequence to sequence learning task.
- CRFs are one sequence-to-sequence model which require predefined context feature functions.
- RNNs automatically provide context but suffer from vanishing gradient problem.
