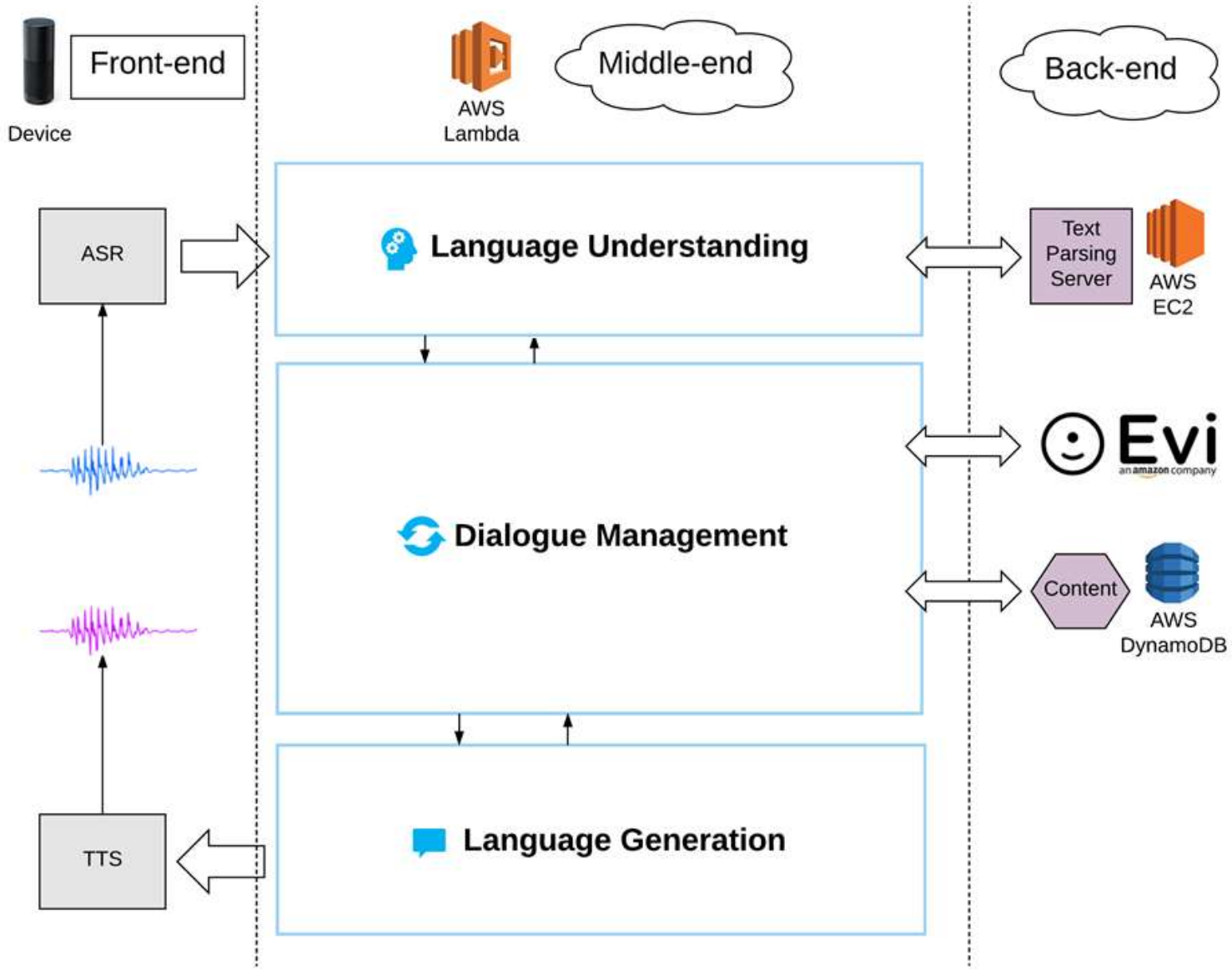
Spoken Language Understanding

EE596/LING580 -- Conversational Artificial Intelligence

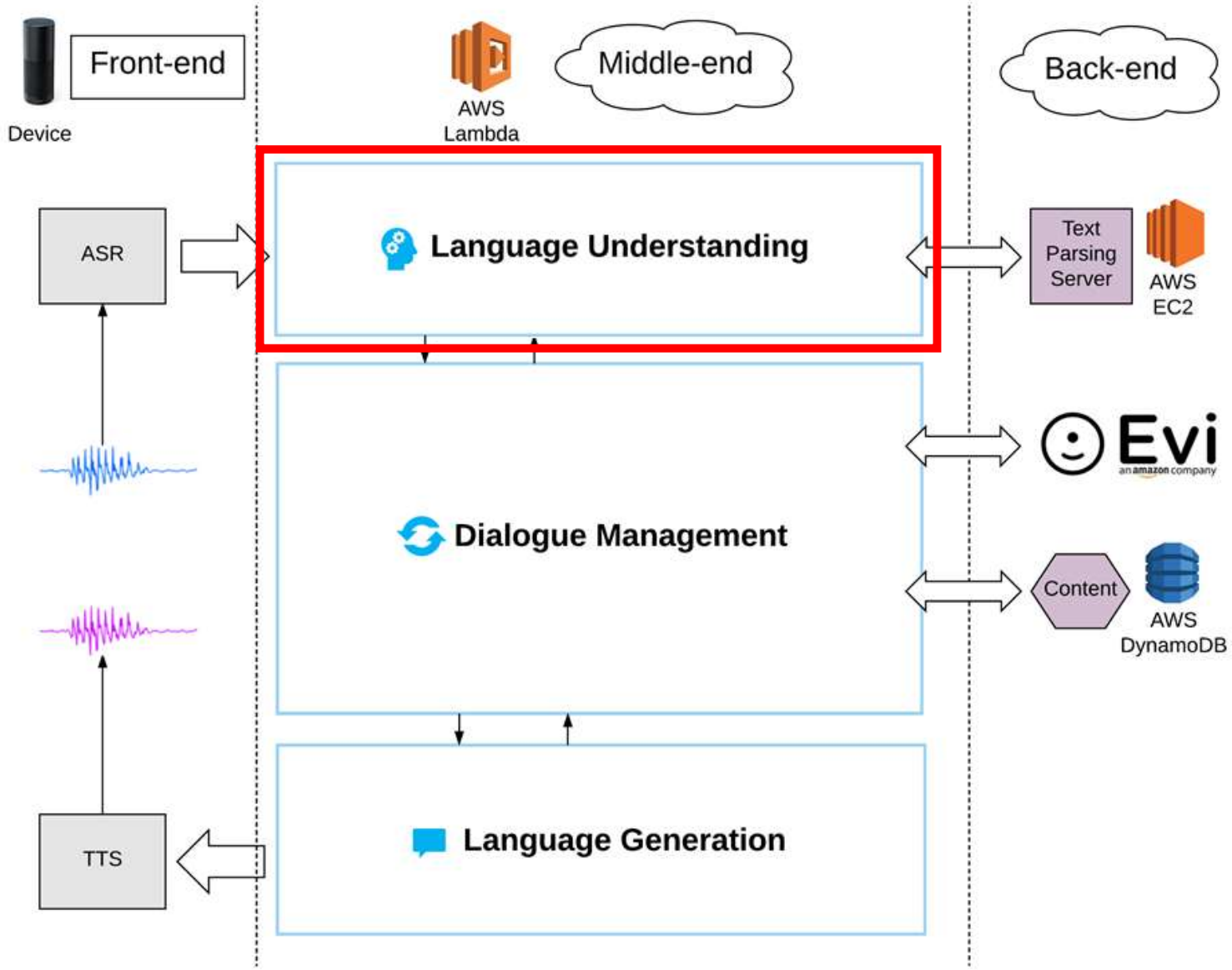
Hao Cheng

University of Washington

Conv AI System Diagram



Conv AI System Diagram



Agenda

- Spoken Language Understanding (SLU)
- Frame-based SLU
- Intent Classification
- Named Entity Recognition
- Case study: Recurrent Neural Network for SLU

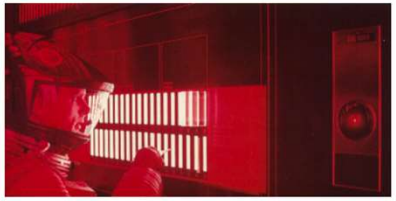
“Can machines think?”

A. M. Turing (1950) – Computing Machinery and Intelligence

“Nevertheless I believe that at the end of the century the use of words and general educated opinion will have altered so much that one will be able to speak of machines thinking without expecting to be contradicted.”

Sci-fi vs. Reality

HAL



David Bowman: Open the pod bay doors, HAL.

HAL: I'm sorry, Dave, I'm afraid I can't do that.

David: What are you talking about, HAL?

HAL: I know that you and Frank were planning to disconnect me, and I'm afraid that's something I cannot allow to happen.

Siri (2011)

Colbert: … I don't want to search for anything! I want to write the show!
Siri: Searching the Web for "search for anything. I want to write the shuffle."
Colbert: … For the love of God, the cameras are on, give me something?
Siri: What kind of place are you looking for? Camera stores or churches?

example from Andrew McCallum

Language Understanding

- Goal: extract **meaning** from natural language
- Ray Jackendoff (2002) – "Foundations of Language"
  - *"meaning" is the "holy grail" for linguistics and philosophy*
- **Spoken** Language Understanding (SLU)
  - self-corrections
  - hesitations
  - repetitions
  - other irregular phenomena

Terminology: NLU, NLP, ASR, TTS

- **N**atural **L**anguage **P**rocessing
- **N**atural **L**anguage **U**nderstanding
- **A**utomatic **S**peech **R**ecognition
- **T**ext-**T**o-**S**peech

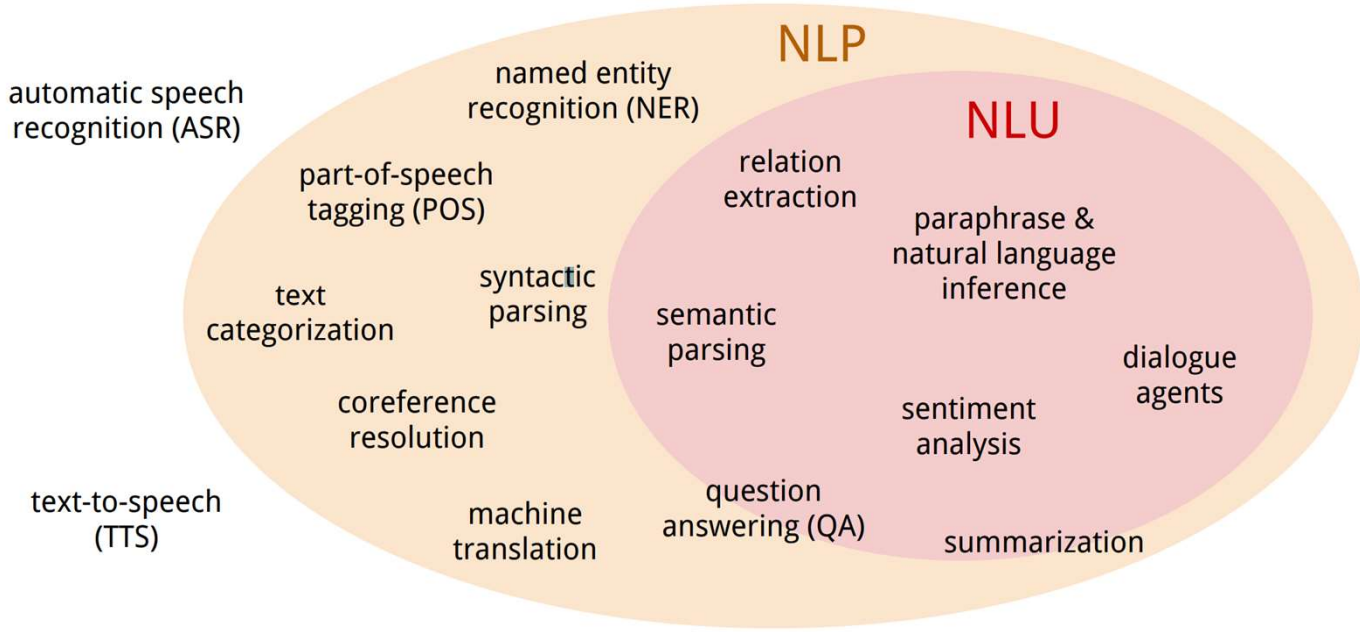


Figure from: Bill MacCarteny – "Understanding Natural Language Understanding" (July 16, 2014)

Early SLU systems

- Historically, early SLU systems used **text-based NLU**.
- S control: ASR generates a sequence of word hypotheses.
  - Knowledge Source (KS): acoustic, lexical, language knowledge
- NLU control: text-based NLU
  - KS: syntactic and semantic

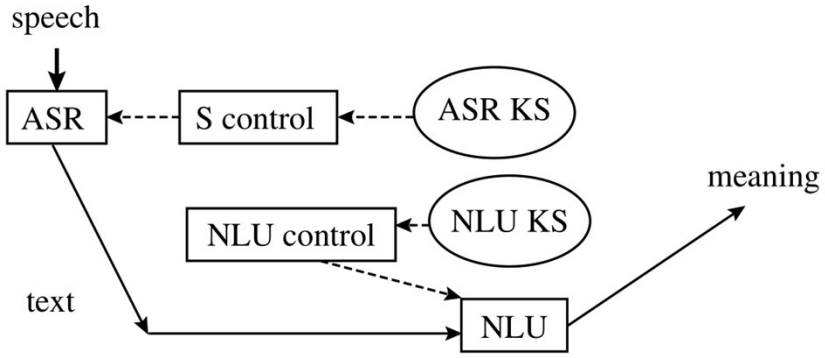


Figure 2.1 Scheme of early SLU system architectures

Figure from: Gokhan Tur and Renato De Mori (2011) – "Spoken Language Understanding".

Meaning Representation Language (MRL)

- Programming Languages
  - syntax: legal programming statements
  - semantics: operations a machine performs when a syntactically correct statement is executed
- An MRL also has its own *syntax* and *semantics*
- Coherent with a *semantic theory*
- Crafted based on the desired capability of each application
- Two widely accepted MRL framework (Who did what to Whom)
  - FrameNet: https://framenet.icsi.berkeley.edu/fndrupal/
  - PropBank: https://propbank.github.io/

Frame-based SLU

Frame-based SLU

- The structure of the semantic space can be represented by a set of ***semantic frames***.
- Each frame contains several typed components called ***slots***.
- Goal: choose correct semantic frame for an utterance and fill the slots based on the utterance.

```
<frame name="ShowFlight" type="Void">
    <slot name="topic" type="Topic">
    <slot name="flight" type="Flight">
</frame>
<frame name="GroundTrans" type="Void">
    <slot name="city" type="City">
    <slot name="type" type="TransType">
</frame>
<frame name="Flight" type="Flight">
    <slot name="DCity" type="City">
    <slot name="ACity" type="City">
    <slot name="DDate" type="Date">
</frame>
```

Table from: Gokhan Tur and Renato De Mori (2011) – "Spoken Language Understanding".

Frame-based SLU: Example

- Show me flights from Seattle to Boston on Christmas Eve.

```
<ShowFlight>
    <topic type="Freeform">FLIGHT</topic>
    <flight frame="Flight" type="Flight">
        <DCity type="City">SEA</DCity>
        <ACity type="City">BOS</ACity>
        <DDate Type="Date">12/24</DDate>
    </flight>
</ShowFlight>
```

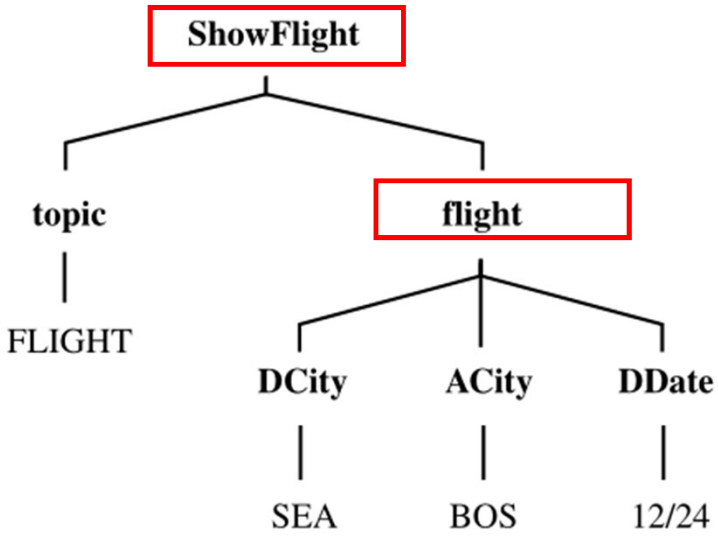


Table from: Gokhan Tur and Renato De Mori (2011) – "Spoken Language Understanding".

Simpler Frame-based SLU

- Some SLU systems do not allow any sub-structures in a frame.
- *attribute-value pairs / keyword-pairs / flat concept*

[**topic**: FLIGHT] [**DCity**: SEA] [**ACity**: BOS][**DDate**: 12/24]

Figure 3.4 The attribute-value representation is a special case of the frame representation where no embedded structure is allowed. Here is an attribute-value representation for "Show me the flights from Seattle to Boston on Christmas Eve" (Wang *et al.*, © 2005 IEEE)

Figure from: Gokhan Tur and Renato De Mori (2011) – "Spoken Language Understanding".

Technical Challenges

- Extra-grammaticality
  - not as well-formed as written language
  - people are in general less careful with speech than with writing
  - no rigid syntactic constraints
- Disfluencies
  - false starts, repairs, hesitations are pervasive
- Speech recognition errors
  - ASR is imperfect (4 miles, for miles, form isles, for my isles)
- Out-of-domain utterances

Knowledge-based Approaches

- Many advocates of the knowledge-based approach believe that general linguistic knowledge is helpful in modeling domain-specific language.
- How to inject the domain specific semantic constraints into a domain-independent grammar?

Semantically Enhanced Syntactic Grammars

- low-level **syntactic non-terminals** -> **semantic non-terminals**

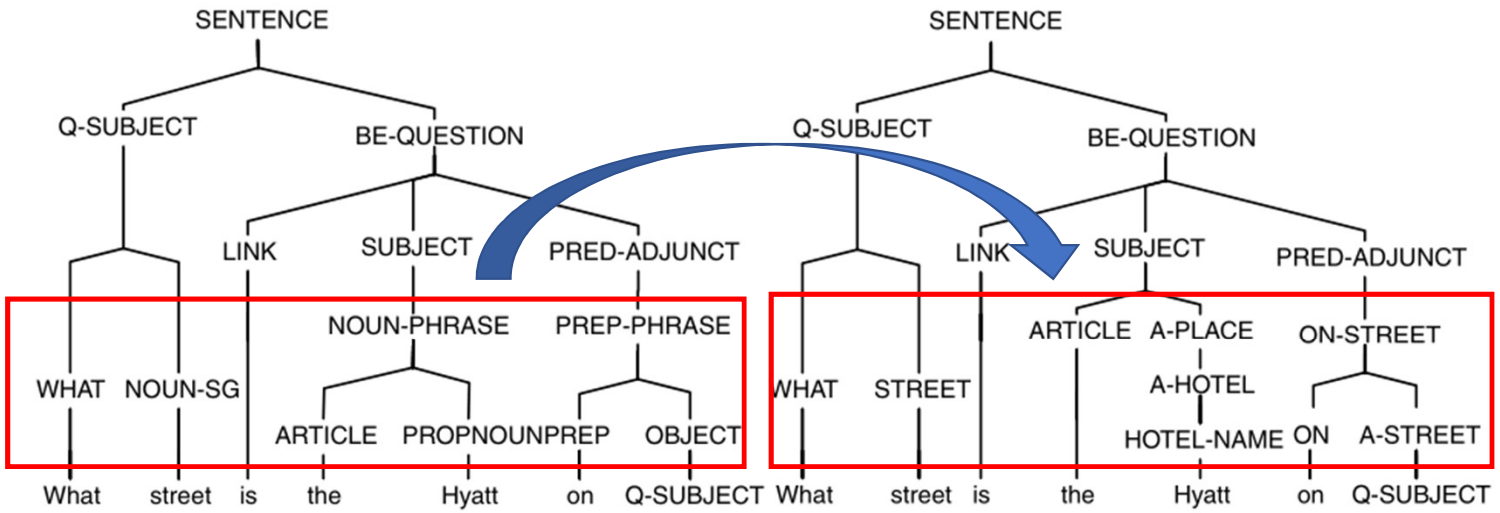


Figure 3.7 TINA parse tree with syntactic rules only (left) and with lower-level syntactic rules replaced by domain-dependent semantic rules (right) (The second tree is reproduced from Seneff (1992) (© 1992 Seneff))

Figure from: Gokhan Tur and Renato De Mori (2011) – "Spoken Language Understanding".

Semantic Grammars

- Directly models the domain-dependent semantics
- Phoenix (Ward, 1991) for ATIS
  - 3.2K non-terminals
  - 13K grammar rules

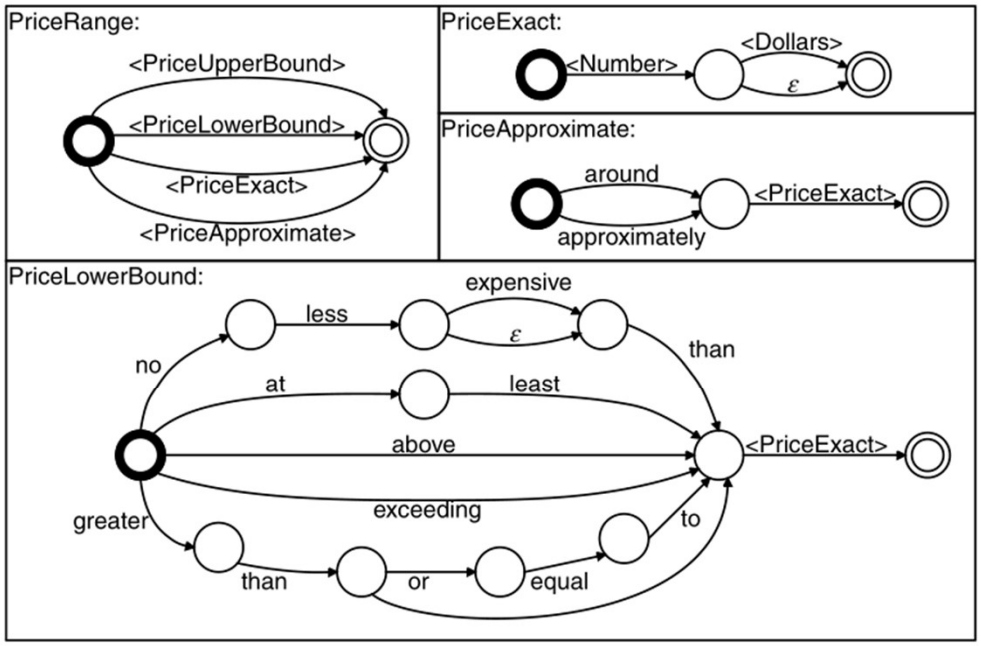


Figure 3.8 Recursive transition network for "PriceRange," together with three sub-nets called by it: "PriceExact", "PriceApproximate" and "PriceLowerBound." The arc labels in angular brackets indicate calls to sub-networks

Figure from: Gokhan Tur and Renato De Mori (2011) – "Spoken Language Understanding".

Knowledge-based Approach

- Advantage:
  - no or less dependent on labeled data
  - almost everyone can start writing a SLU grammar with some basic training
- Disadvantage
  - grammar development is an error-prone process (simplicity vs. coverage)
  - it takes multiple rounds to fine tune a grammar
  - scalability

Data-driven Approaches

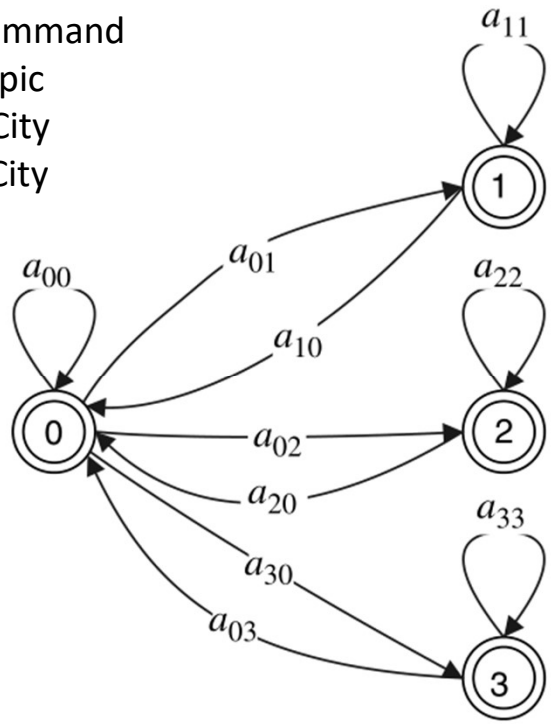
- Word sequence W
- Meaning representation M

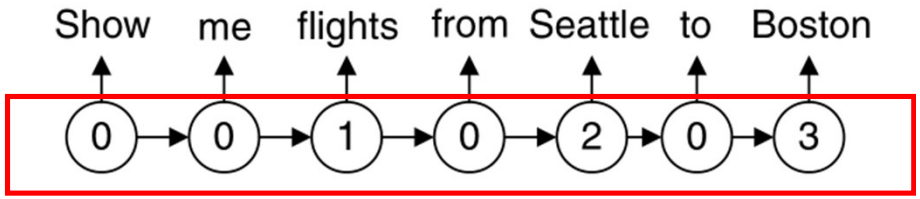
$$\hat{M} = \arg\max_{M} P(M \mid W) = \arg\max_{M} P(W \mid M) P(M)$$

- Generative Model
  - Generates the natural language from the underlying meaning
  - P(M): semantic prior model
  - P(W|M): lexicalization / lexical generation / realization model
- Discriminative Model
  - Predicts the underlying meaning on top of the natural language
  - P(M|W)

Hidden-Markov Model (HMM)

- State 0: command
- State 1: topic
- State 2: DCity
- State 3: ACity





$$\Pr(M) = \pi_0 a_{00} a_{01} a_{10} a_{02} a_{20} a_{03} a_{30}$$

$$\Pr(W \mid M) = b_0(\text{Show}) \times b_0(\text{me}) \times b_1(\text{flights}) \times b_0(\text{from}) \times b_2(\text{Seattle}) \times b_0(\text{to}) \times b_2(\text{Boston})$$

Figure from: Gokhan Tur and Renato De Mori (2011) – “Spoken Language Understanding”.

Conditional Random Field (CRF)

- Word sequence $x_1, \ldots, x_n$
- Meaning representation (state sequence) $y_1, \ldots, y_n$

$$P(\mathbf{y} \mid \mathbf{x}; \Lambda) = \frac{1}{Z(\mathbf{x}; \Lambda)} \exp\left\{\sum_k \lambda_k f_k(\mathbf{y}, \mathbf{x})\right\}$$

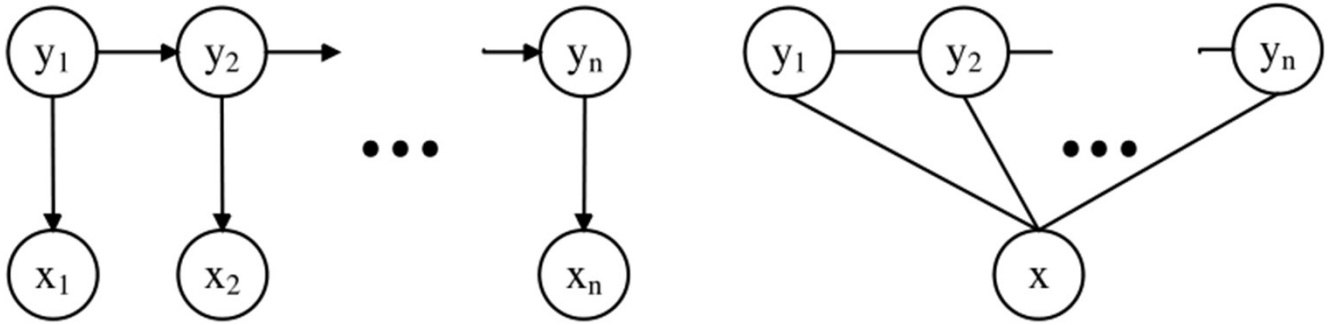


Figure from: Gokhan Tur and Renato De Mori (2011) – "Spoken Language Understanding".

Project

- Voice-based Bot is preferred and No task-oriented
  - English is the language if voice-based
  - Text-based bot might be a better for other languages
  - Multi-turn dialog
- Projects from last year
  - https://hao-fang.github.io/ee596_spr2018/syllabus.html
- Big picture vs concrete execution plan
  - Question answering in dialog regarding news article
  - Recommendation, movie/book/restaurant/job…
  - Debate and opinion sharing
  - Language acquisition
  - Psychology (probing personality)
  - Collaborative Storytelling

Intent Classification

System-initiative Systems

- Interaction is completely controlled by the machines.
  - *Please say collect, calling card, or third party.*
- Commonly known as Interactive Voice Response systems(IVR)
  - Now widely implemented using established and standardized platforms such as VoiceXML.
- A primitive approach, a great commercial success

Utterance Level Intents

- AT&T's **H**ow **M**ay **I** **H**elp **Y**ou system

HMIHY: How may I help you?
User: Hi, I have a question about my bill (*Billing*)
HMIHY: OK, what is your question?
User: May I talk to a human please? (*CSR*) (Customer Service Representative)
HMIHY: In order to route your call to the most appropriate department can you tell me the specific reason you are calling about?
User: There is an international call I could not recognize (*Unrecognized_Number*)
HMIHY: OK, I am forwarding you to the human agent. Please stay on the line.

Figure 4.2 A conceptual example dialogue between the user and the AT&T HMIHY system

Figure from: Gokhan Tur and Renato De Mori (2011) – "Spoken Language Understanding".

Intent Classification

- Task: Classify users' utterances into predefined categories
- Speech utterance X_r
- M semantic classes: $C_1, C_2, \ldots, C_M$

$$\hat{C}_r = \arg\max_{C_r} P(C_r|X_r).$$

- Similar intent with significant freedom in utterance variations
  - *I want to fly from Boston to New York next week*
  - *I am looking to fly from Boston to JFK in the coming week*

Intent Classification vs. Frame-based SLU

- Coarse (high-level) vs fine-grained (detail)
  - Less attention to the underlying message conveyed
- Heavily rely on statistical methods
- Fit nicely into spoken language processing
  - less grammatical and fluent
  - ASR errors
- Out-of-domain utterances are still challenging
  - *I want to book a flight to New York next week*
  - *I want to book a restaurant in New York next week*

Dialog Act

- A **Speech Act** is a primitive abstraction or an approximate representation of the illocutionary force of an utterance. (Austin 1962)
  - asking, answering, promising, suggesting, warning, or requesting
- Five major classes (Searle, 1969)
  - Assertive: commit the speaker to something is being the case
    - suggesting, concluding
  - Directive: attempts by the speaker to do something
    - ordering, advising
  - Commissive: commit the speaker to some future action
    - planning, betting
  - Expressive: express the psychological state of the speaker
    - thanking, apologizing
  - Declaration: bring about a different state of the world
    - *I name this ship the Titanic*

Named Entity Recognition

What is a Named Entity?

- Introduced at the MUC-6 evaluation program (Sundheim and Grishman, 1996) as one of the *shallow understanding* tasks.
- No formal definition from a linguistic point of view.
- Goal: extract from a text all the word strings corresponding to these kinds of entities and from which a unique identifier can be obtained without resolving any reference resolution process.
  - New York city: yes
  - the city: no
  - The White House: yes
  - the white house by the lake: no

Entity Categories

1. ENAMEX
   - ORGANIZATION: named corporate, governmental, or other organizational entity
   - PERSON: named person or family
   - LOCATION: name of politically or geographically defined location (cities, provinces, countries, international regions, bodies of water, mountains, etc.)
2. TIMEX
   - DATE: complete or partial date expression
   - TIME: complete or partial expression of time of day
3. NUMEX
   - MONEY: monetary expression
   - PERCENT: percentage

Technical Challenges

- Segmentation ambiguity
  - [University of California Berkeley]
  - [University of California] [Berkeley]
  - [Elon Musk, the CEO of Telsa]
  - [Elon Musk], the CEO of [Telsa]
- Classification ambiguity
  - John F. Kennedy (JFK): PERSON vs. AIRPORT

Approaches

- Rules and Grammars
- Word Tagging Problem

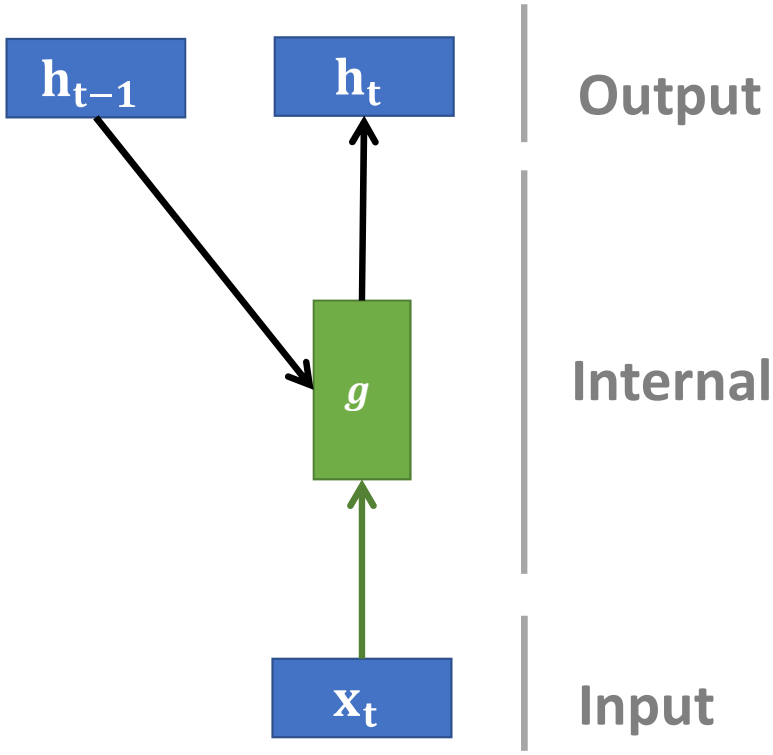
Sentence	*show*	*flights*	*from*	*Boston*	*To*	*New*	*York*	*today*
Slots/Concepts	O	O	O	B-dept	O	B-arr	I-arr	B-date
Named Entity	O	O	O	B-city	O	B-city	I-city	O

Break (15min)

Recurrent Neural Networks for SLU

RNN Unit

- Two inputs
  - Current input into network, $x_t \in \mathbb{R}^v$
  - Previous hidden state, $h_{t-1} \in \mathbb{R}^n$
- One output
  - Current hidden state, $h_t \in \mathbb{R}^n$



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

- Simple RNN Unit

$$h_t = g(Wx_t + Uh_{t-1}),$$

where $W \in \mathbb{R}^{n \times v}$ and $U \in \mathbb{R}^{n \times n}$ are model parameters, $g(\cdot)$ is a non-linear transformation function.

Simple RNN Unit

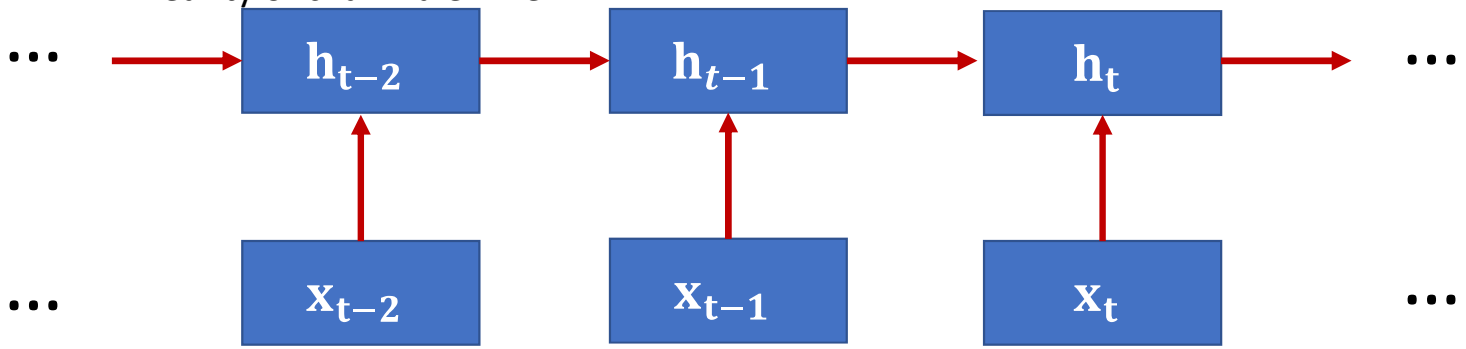
- Simple RNN Unit

$$h_t = g(Wx_t + Uh_{t-1})$$

- Hidden state is designed to memorize all history info
- Non-linear transformation
  - Sigmoid: [0, 1]
  - Hyperbolic Tangent: [-1, 1]
  - Sigmoid vs Hyperbolic Tangent
    - Sigmoid function has almost zero gradient almost everywhere
    - Sigmoid function is sensitive to the data centralization.

Training an RNN

- An RNN model with simple RNN unit
  - Stochastic gradient descent (SGD) with backpropagation through time (BPTT).
  - Difficult in optimization
    - Exploding gradient: careful clipping or truncating.
    - Vanishing gradient:
      - Result in the inability to capture dependencies over longer time span
      - Linearity of chain rule when BPTT



Long Short Term Memory (LSTM)

- h_t in RNN servers 2 purpose
  - make output predictions
  - represent the data sequence processed so far
- The LSTM cell split these two roles into two separate variables
  - h_t: make output predictions
  - C_t: save the internal state

$$\tilde{C} = \tanh(W_{cx} X_t + W_{ch} h_{t-1} + b_c)$$
$$C_t = gate_{forget} \cdot C_{t-1} + gate_{input} \cdot \tilde{C}$$
$$h_t = gate_{out} \cdot \tanh(C_t)$$

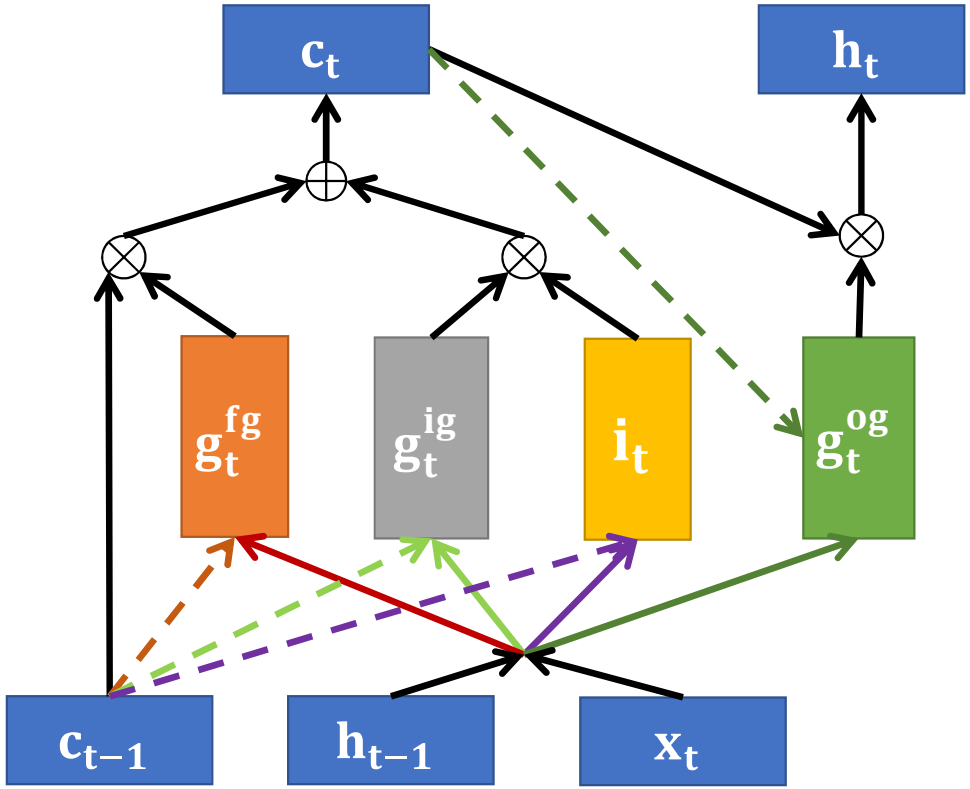
LSTM Gates

- Forget gate: what part of the previous cell state will be kept
- Input gate: what part of the new computed information will be added to the cell state C_t
- Output gate: what part of the cell state C_t will be exposed as the hidden state

$$\tilde{C} = \tanh(W_{cx}X_t + W_{ch}h_{t-1} + b_c)$$
$$C_t = gate_{forget} \cdot C_{t-1} + gate_{input} \cdot \tilde{C}$$
$$h_t = gate_{out} \cdot \tanh(C_t)$$

$$gate_{forget} = \sigma(W_{fx}X_t + W_{fh}h_{t-1} + b_f)$$
$$gate_{input} = \sigma(W_{ix}X_t + W_{ih}h_{t-1} + b_i)$$
$$gate_{out} = \sigma(W_{ox}X_t + W_{oh}h_{t-1} + b_o)$$

LSTM Unit



- LSTM Unit
  - Memory cell
  - Three gates
- Dynamically control the “input”, “forget” and “output” of historical information
- Simplify peephole connections for less complex models

Gated Recurrent Unit (GRU)

- No separate cell
- Two gates
  - Reset gate: what part of the previous state will be kept
  - Update gate: how much the unit updates the state

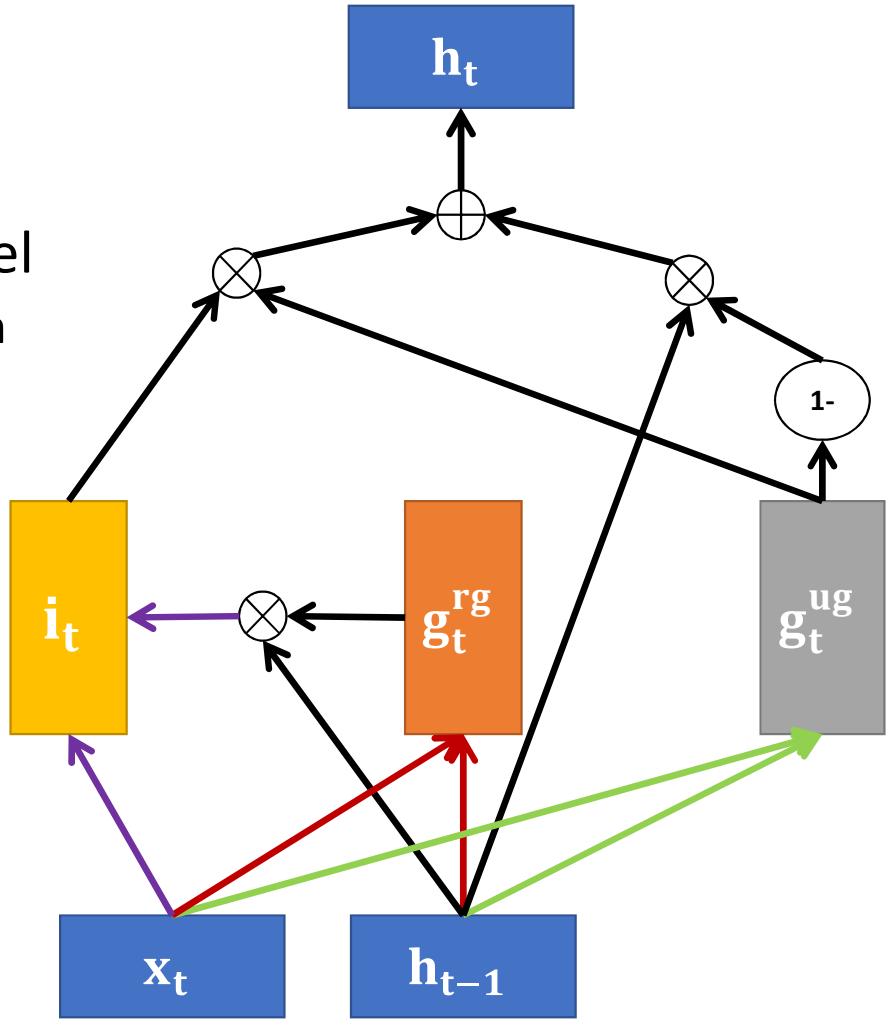
$$h_t = (1 - gate_{update}) \cdot h_{t-1} + gate_{update} \cdot \tilde{h}_t$$

$$\tilde{h}_t^j = \tanh\left(W\mathbf{x}_t + U\left(\mathbf{r}_t \odot \mathbf{h}_{t-1}\right)\right)^j$$

$$gate_r = \sigma(W_{rx}X_t + W_{rh}h_{t-1} + b)$$
$$gate_{update} = \sigma(W_{ux}X_t + W_{uh}h_{t-1} + b)$$

GRU

- Less complex model
  - Trade-off between
  - LSTM unit
  - Simple RNN unit
  - Leaky integration

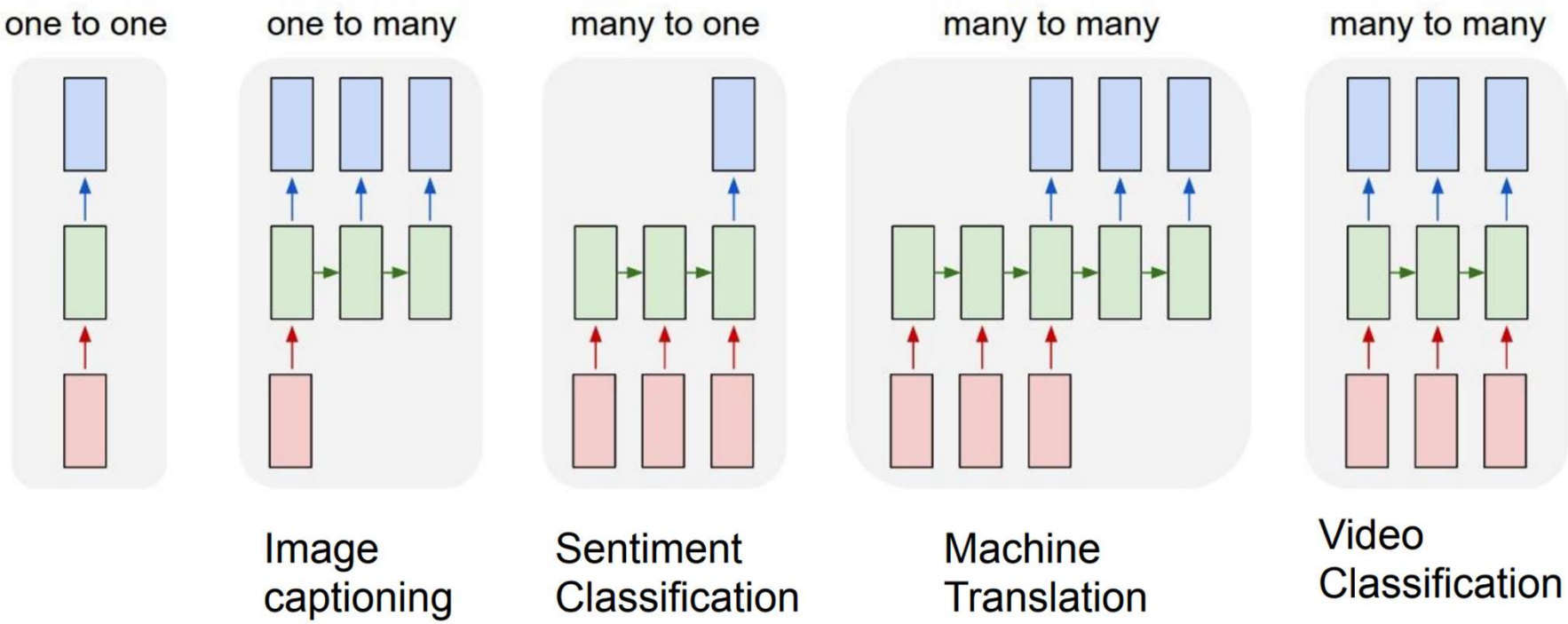


Output

Internal

Input

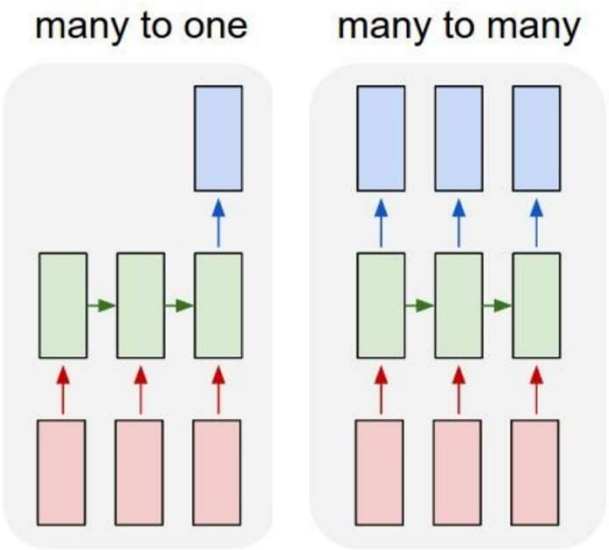
Recurrent Neural Networks



Intent Classification

HMIHY: How may I help you?
User: Hi, I have a question about my bill (*Billing*)
HMIHY: OK, what is your question?
User: May I talk to a human please? (*CSR*)
HMIHY: In order to route your call to the most appropriate department can you tell me the specific reason you are calling about?
User: There is an international call I could not recognize (*Unrecognized_Number*)
HMIHY: OK, I am forwarding you to the human agent. Please stay on the line.

Figure 4.2 A conceptual example dialogue between the user and the AT&T HMIHY system

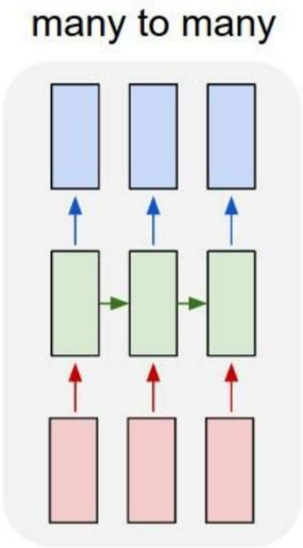


Slot Filling Task

- in/out/begin (IOB) representation

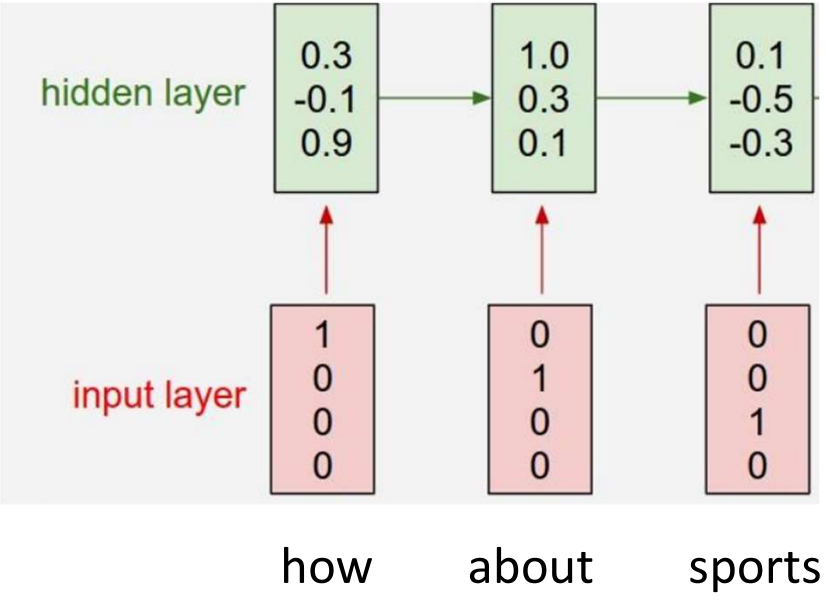
Sentence	*show*	*flights*	*from*	*Boston*	*To*	*New*	*York*	*today*
Slots/Concepts	O	O	O	B-dept	O	B-arr	I-arr	B-date
Named Entity	O	O	O	B-city	O	B-city	I-city	O
Intent	*Find_Flight*							
Domain	*Airline Travel*							

ATIS utterance example IOB representation



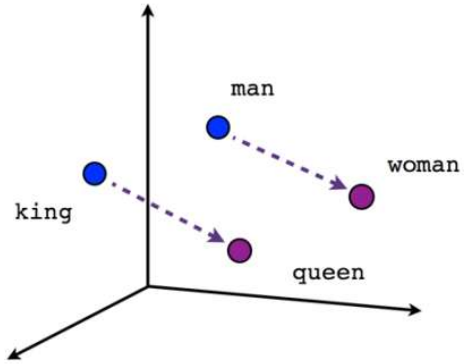
How to represent a word?

- Vocabulary: [how, about, sports, <unk>]
- One-hot encoding (local representation)
  - Representation dimension same as vocab size
  - No semantic relatedness: man vs woman
  - No functional relatedness: he vs him
  - Less friendly to shallow and feature-rich data-drive model (data sparsity)

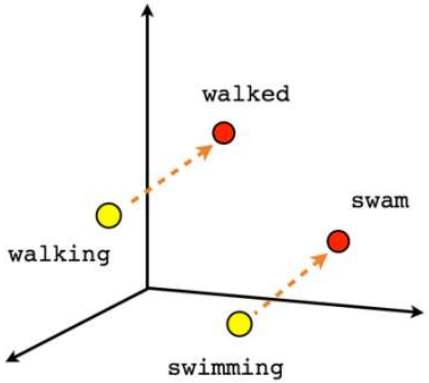


Word Embedding

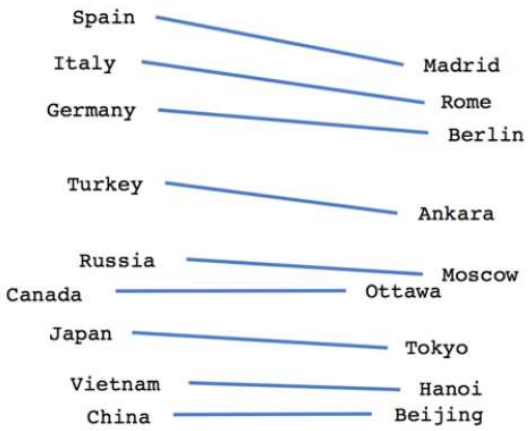
- Distributed Representation based on Distributional Hypothesis
  - Distributed Representation: continuous vector vs one-hot
  - Distributional Hypothesis: the semantics and syntactic function can be defined using the surrounding text (context)



Male-Female



Verb tense



Country-Capital

Figure from: https://www.tensorflow.org/tutorials/word2vec

SLU in Alexa Skills Kit

Creating an Alexa Skill

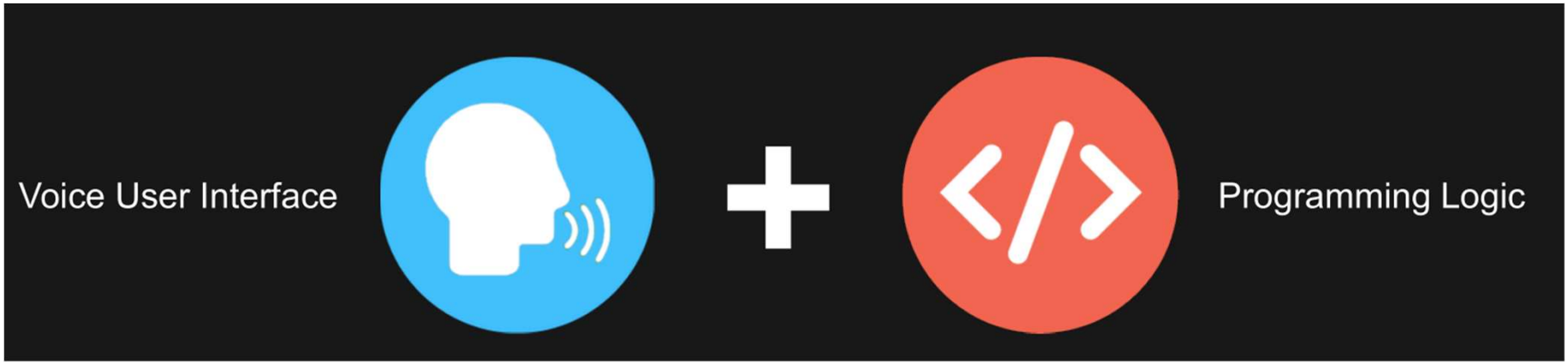


Figure from: Jeff Blankeburg and Alexa Evangelist (2017) – “Build an Alexa Skill using AWS Lambda”.

Creating an Alexa Skill

Figure from: Jeff Blankeburg and Alexa Evangelist (2017) – “Build an Alexa Skill using AWS Lambda”.

Alexa Skills Kit

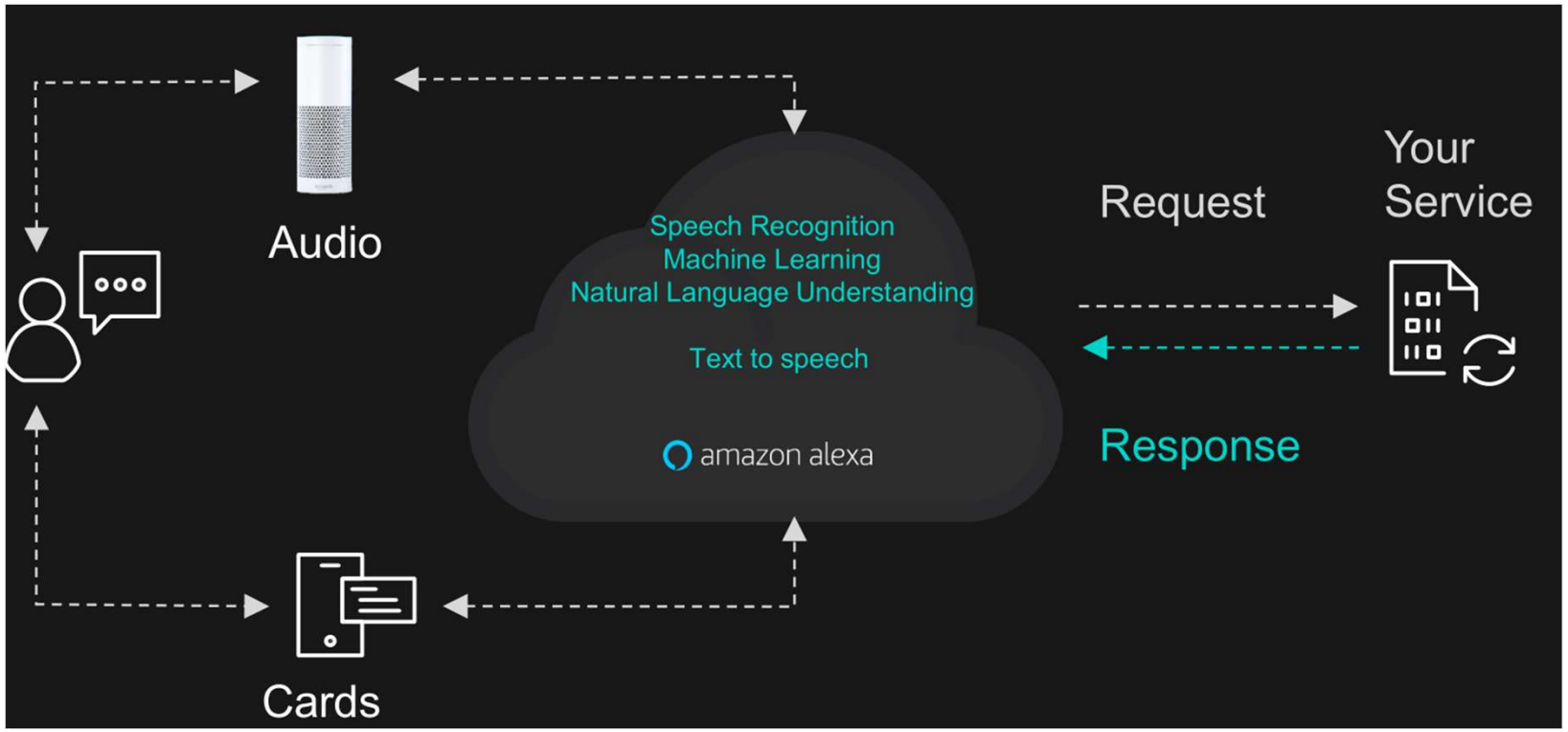


Figure from: Jeff Blankeburg and Alexa Evangelist (2017) – “Build an Alexa Skill using AWS Lambda”.

Alexa Skills Kit: Signal Processing

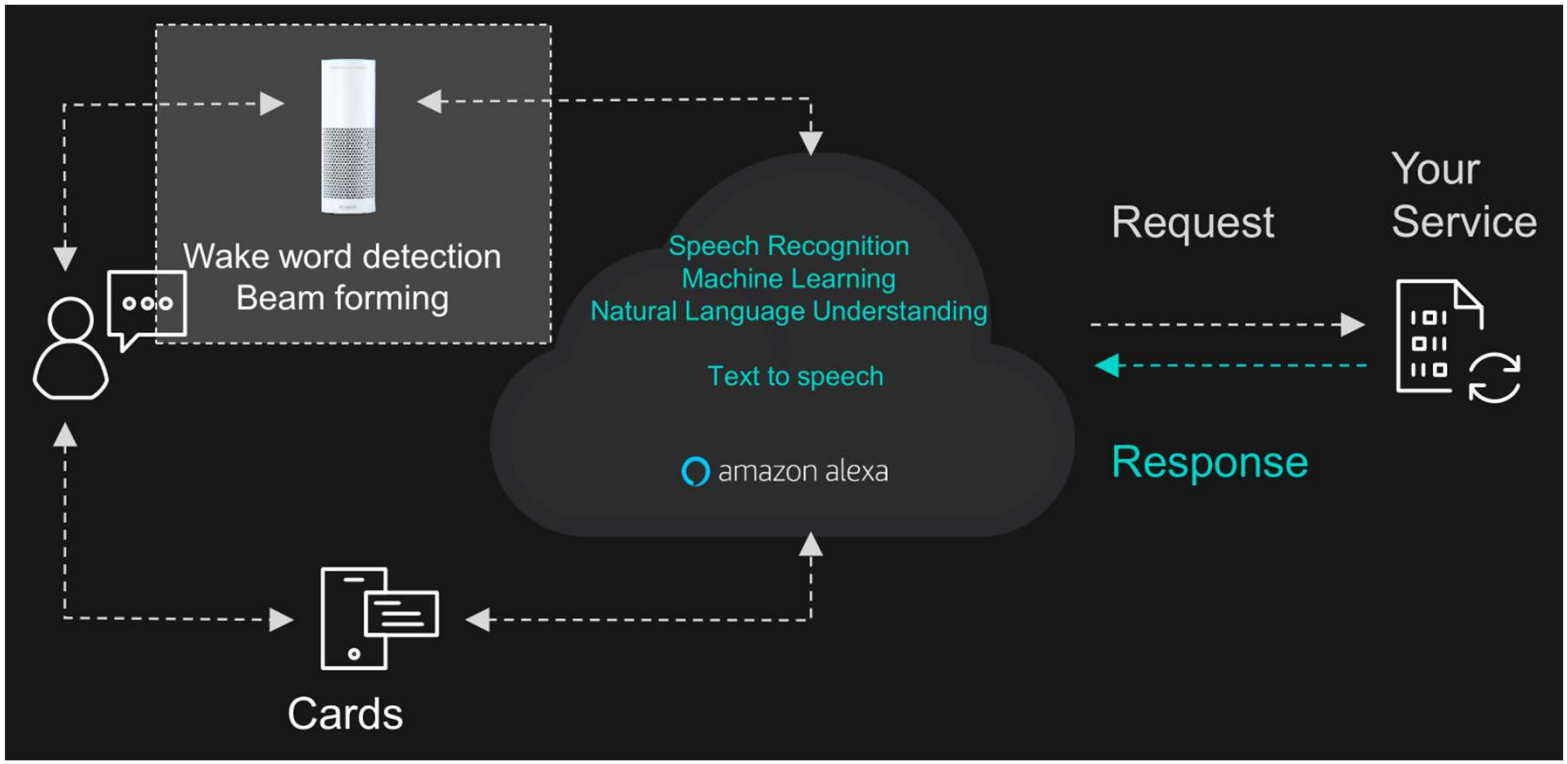


Figure from: Jeff Blankeburg and Alexa Evangelist (2017) – “Build an Alexa Skill using AWS Lambda”.

Alexa Skills Kit: Interaction Model

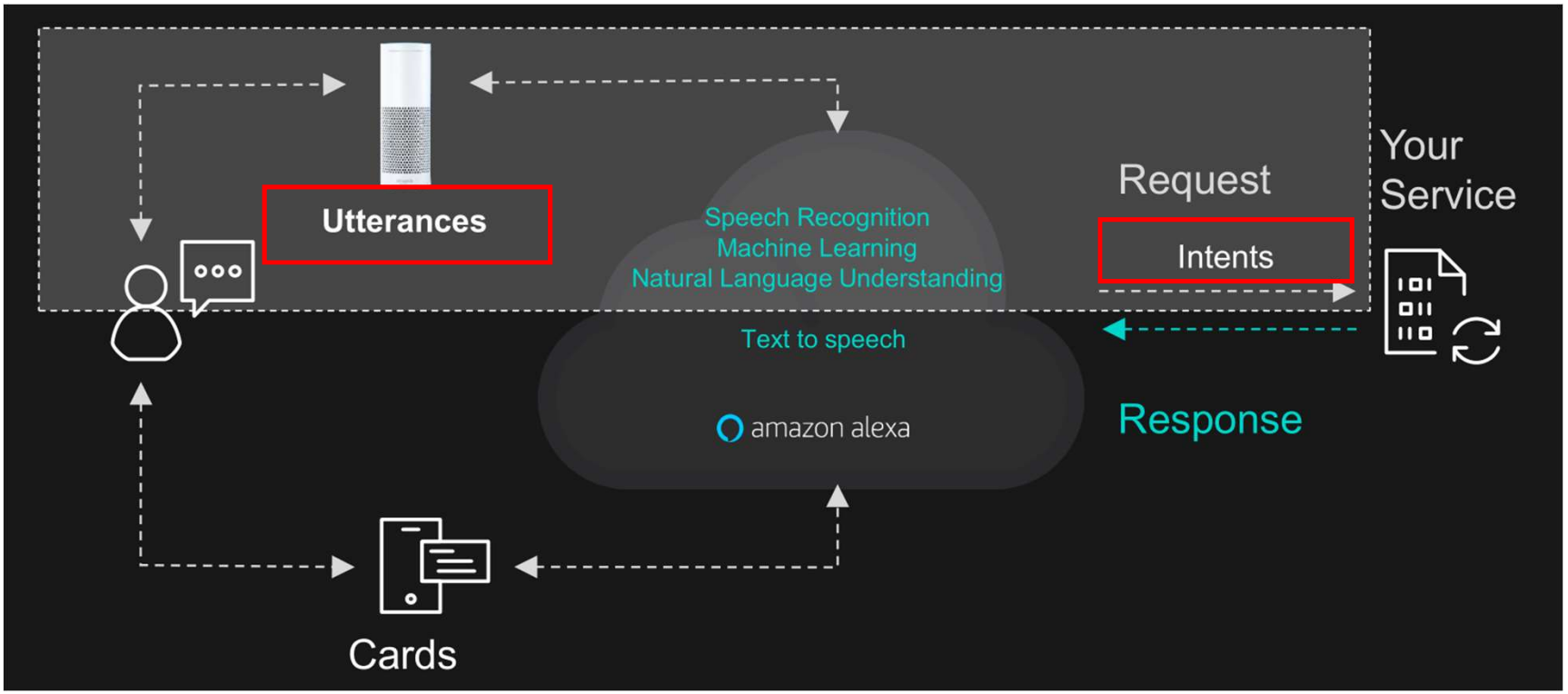


Figure from: Jeff Blankeburg and Alexa Evangelist (2017) – “Build an Alexa Skill using AWS Lambda”.

Intents

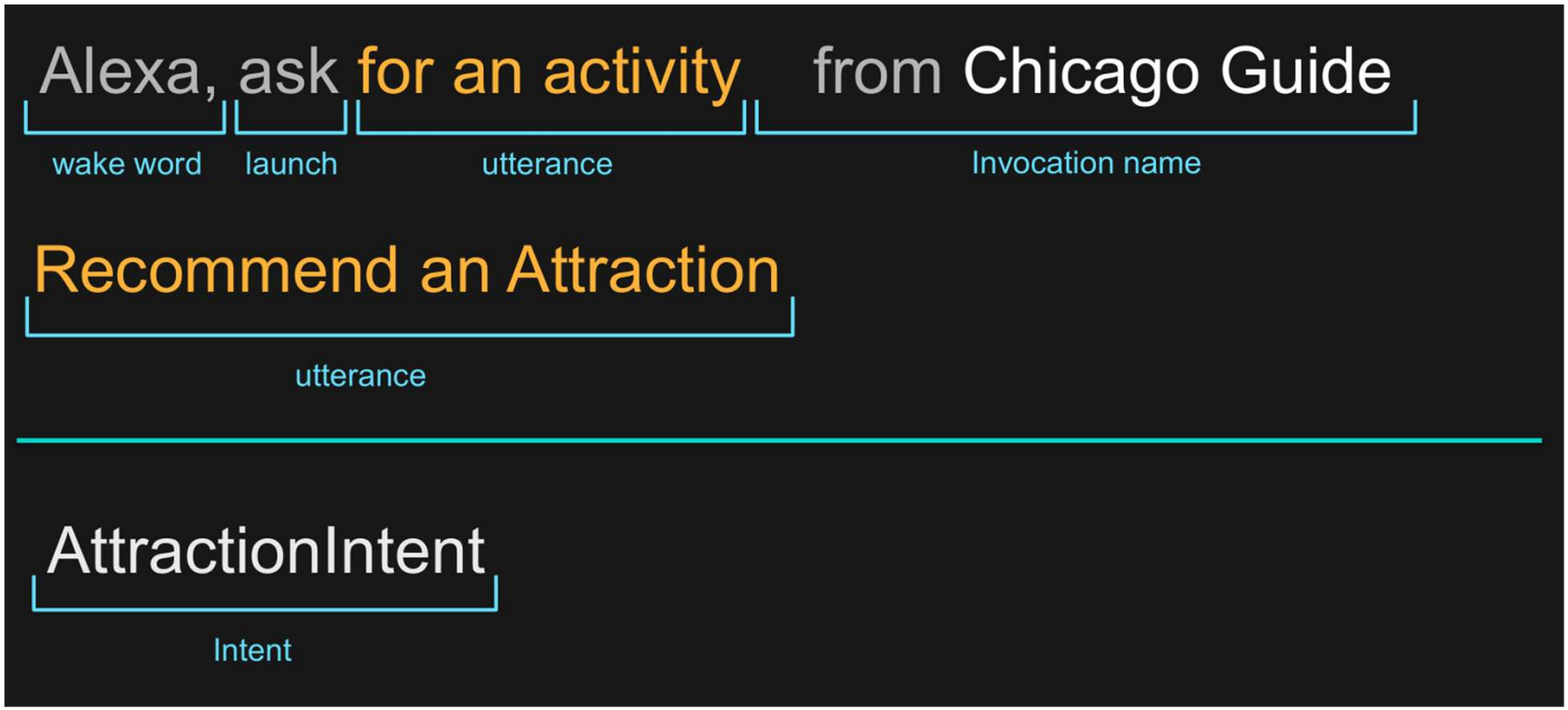


Figure from: Jeff Blankeburg and Alexa Evangelist (2017) – “Build an Alexa Skill using AWS Lambda”.

Built-in Slots

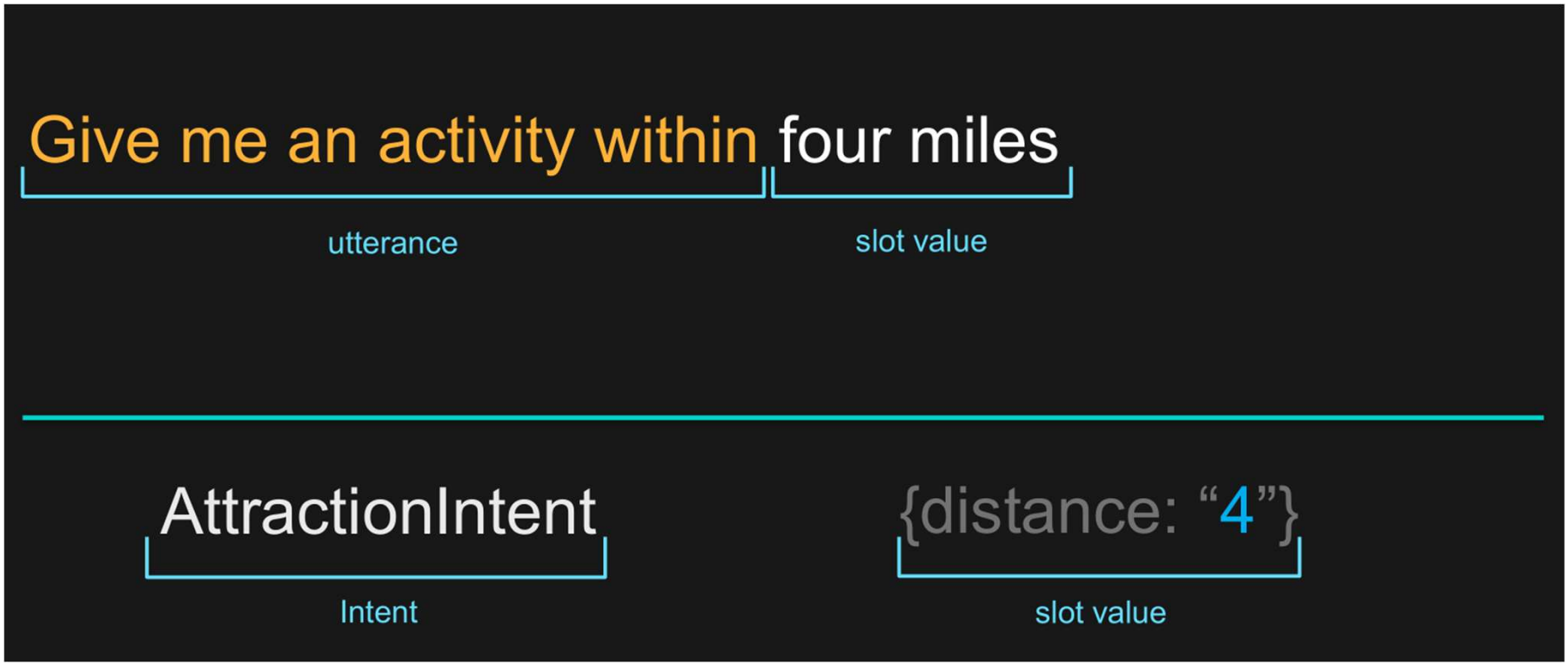


Figure from: Jeff Blankeburg and Alexa Evangelist (2017) – "Build an Alexa Skill using AWS Lambda".

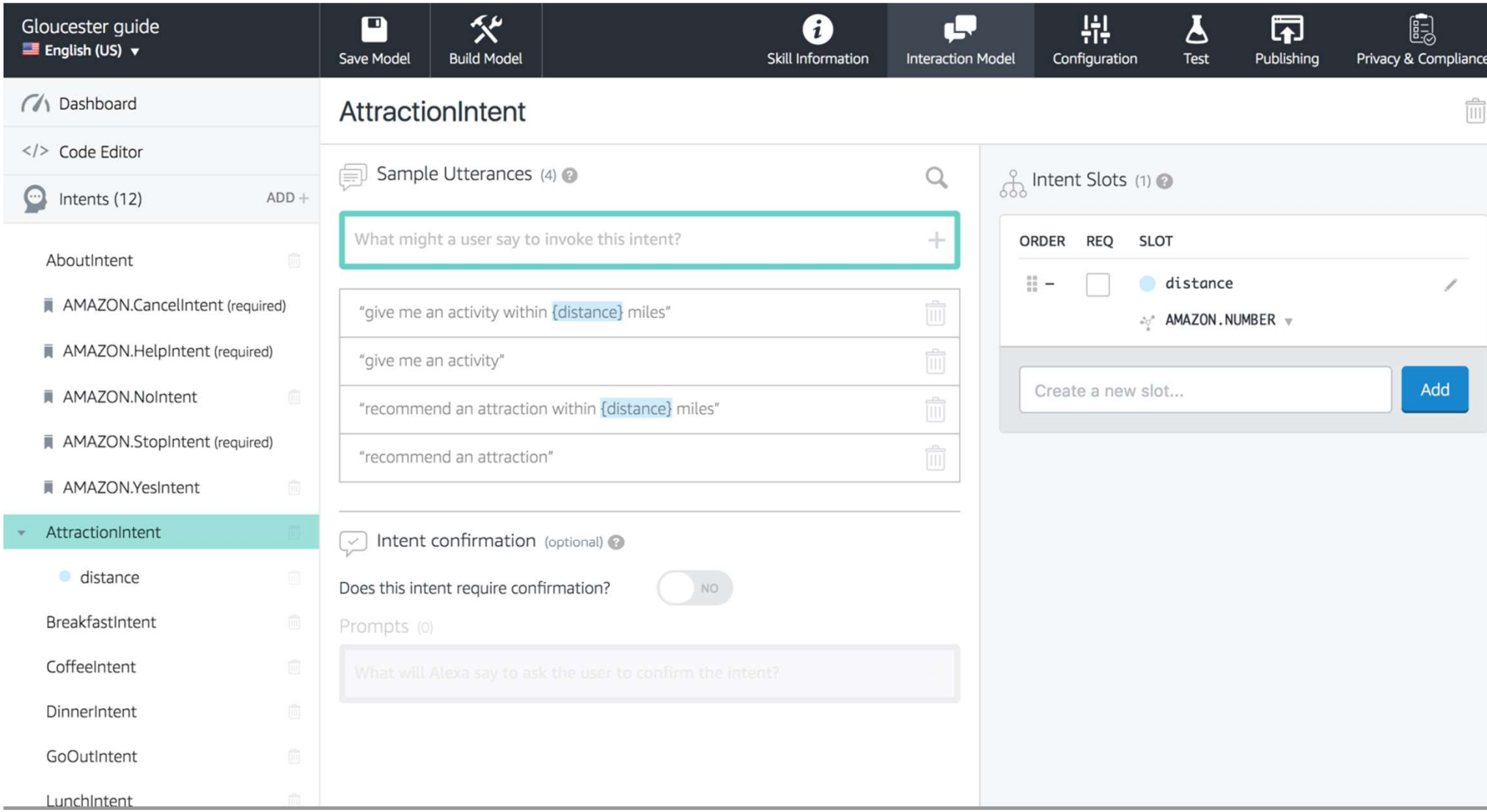


Figure from: Jeff Blankeburg and Alexa Evangelist (2017) – “Build an Alexa Skill using AWS Lambda”.

Custom Slots

Figure from: Jeff Blankeburg and Alexa Evangelist (2017) – "Build an Alexa Skill using AWS Lambda".

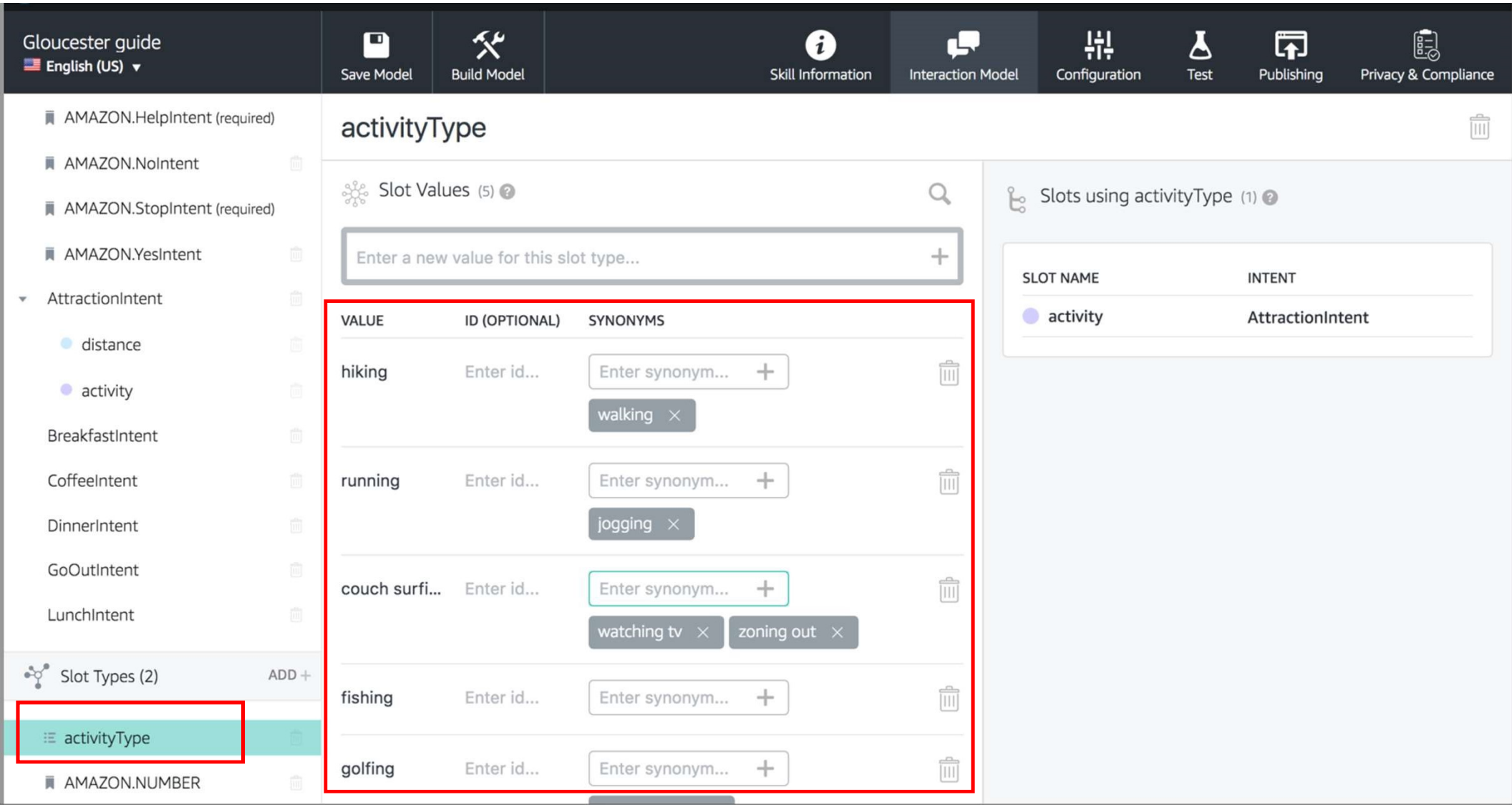


Figure from: Jeff Blankeburg and Alexa Evangelist (2017) – “Build an Alexa Skill using AWS Lambda”.

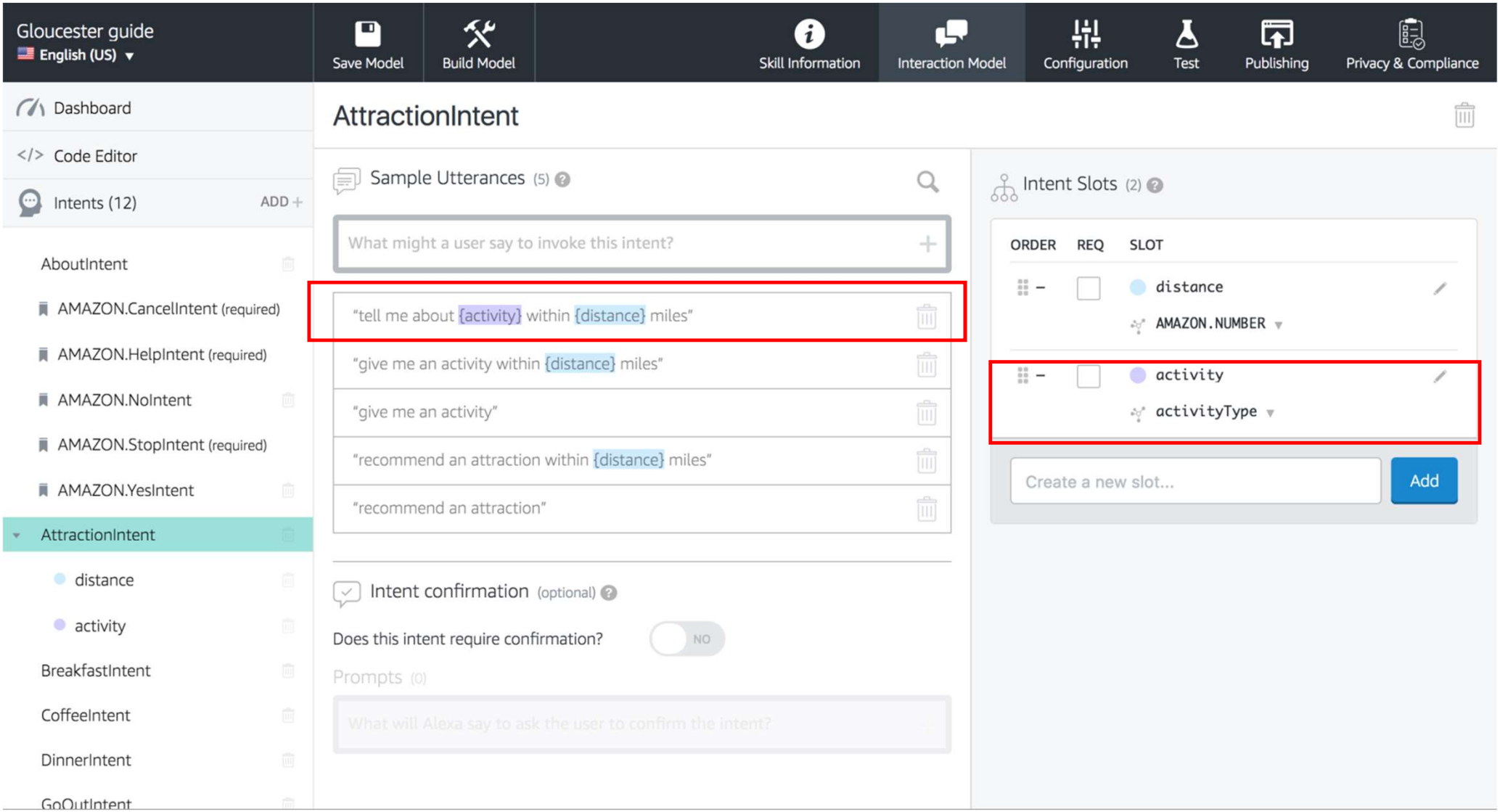


Figure from: Jeff Blankeburg and Alexa Evangelist (2017) – “Build an Alexa Skill using AWS Lambda”.

How Do I Receive My Slot?

```
myDistance = this.event.request.intent.slots.distance.value

myActivity = this.event.request.intent.slots.activity.value
```

Figure from: Jeff Blankeburg and Alexa Evangelist (2017) – “Build an Alexa Skill using AWS Lambda”.

Alexa Skills Kit: Requests and Responses

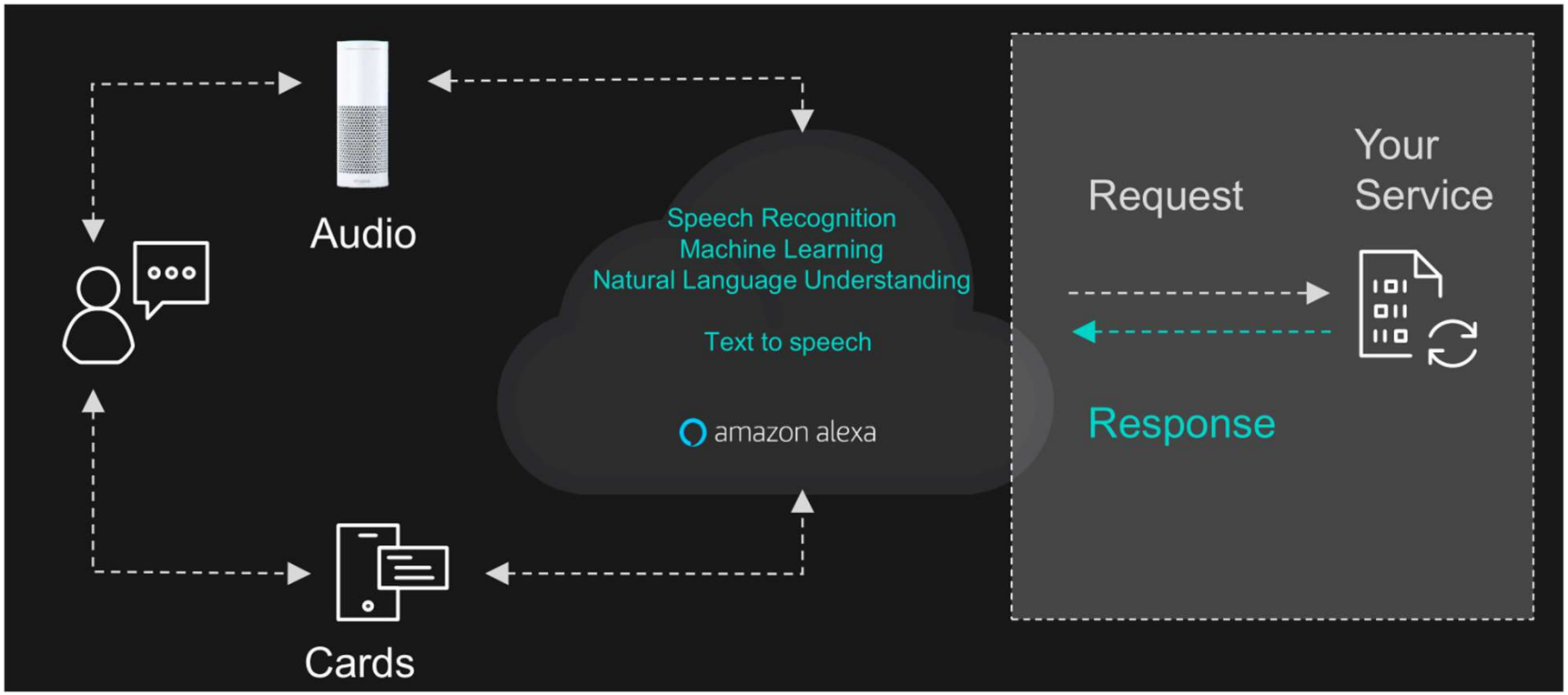


Figure from: Jeff Blankeburg and Alexa Evangelist (2017) – “Build an Alexa Skill using AWS Lambda”.

Alexa Skills Kit: Output

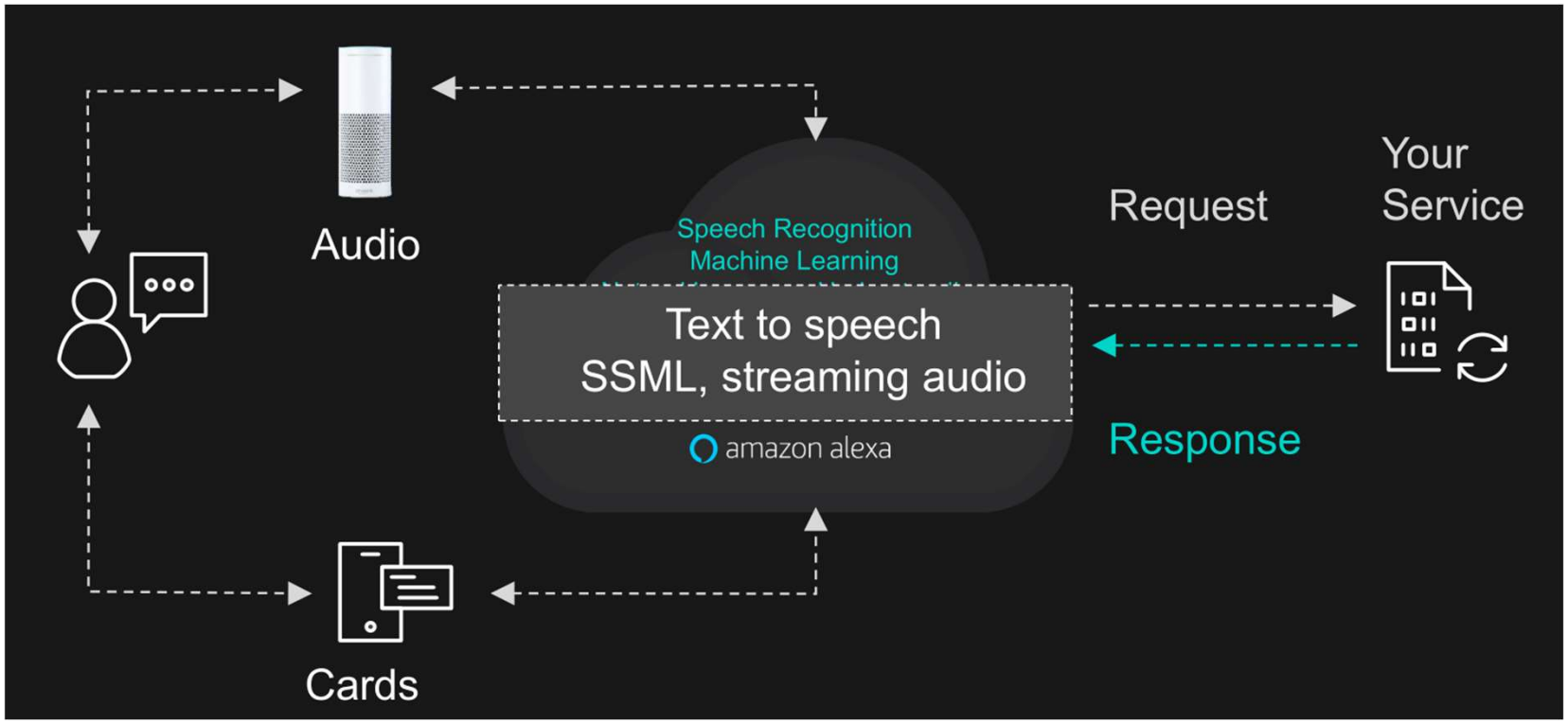


Figure from: Jeff Blankeburg and Alexa Evangelist (2017) – “Build an Alexa Skill using AWS Lambda”.