Lecture 10

Multilingual Virtual Assistants

1. Introduction to Machine Translation
2. Localization for Single Commands
3. Localization for Dialogues

By Mehrad Moradshahi and Monica Lam
Motivation

• Most technologies are restricted to a few widely spoken languages
• Data collection for other languages is the main challenge
• Zero-shot cross-lingual approaches are promising but not quite there yet
• Companies have low business motivation to cover low-resource languages
Motivation

• Even speakers of major languages need multilingual
  • Local entities in their languages (songs, books, locations, cuisines, etc.)
  • Use case: TuneIn
    • 100,000 radio channels in 100 languages; Alexa speaks 10 languages
    • Mixed code: multiple languages in the same sentence
Motivation for Voice

• 6500 languages spoken in the world!
• Inclusion
  • “Why AI Needs to Be Able to Understand All the World’s Languages”
  The benefits of mobile technology are not accessible to most of the world’s 700 million illiterate people
• Use case: African languages
• Use case: Covid appointments for non-English speakers

https://www.scientificamerican.com/article/why-ai-needs-to-be-able-to-understand-all-the-worlds-languages/
Objective

• To democratize Virtual Assistant Technology for all languages by mostly automatic techniques!

• Can we do this cost and time efficiently, perhaps in a day?!
Traditional Localization for Software

• Localization: changing a product to a different language and adapting to a country or a region
• Expensive even for traditional graphical interface
• Professional translation services
• Pros:
  • High quality
  • More natural (handles colloquial text better)
• Cons:
  • Slow and expensive
  • Lack of consistency in style and terminology
  • Require in-depth knowledge of the content
Rule-Based Machine Translation

- Relies on built-in linguistic rules and bilingual dictionaries
- Text → a transitional representation → text in target language
- Transfers the grammatical structure of the source language into the target language
- Requires extensive lexicons with morphological, syntactic, semantic info & large sets of rules.
Statistical Machine Translation

• Use statistical models
  • Warren Weaver in 1949
  • Parameters are derived from analyzing bilingual text corpora
  • Relations and features are hand-crafted
Rosetta Stone

• First known piece of translation
• Inscribed with a decree Memphis, Egypt, 196 BC on behalf of King Ptolemy V.
• In three scripts:
  • Ancient Egyptian hieroglyphics:
    • Literate Egyptian priesthood
  • Demotic script
    • Language of documents
  • Ancient Greek
    • The government of Egypt had been Greek-speaking ever since the conquests of Alexander the Great.
• Key to understanding Egyptian hieroglyphics
Neural Machine Translation

• Approach: Example based
  • Deep learning
  • Representation learning
    • Use of vector representations ("embeddings", "continuous space representations") for words and internal states.

• Benefit
  • Trained directly on source and target text

Seq2Seq Network with Recurrent Encoder and Decoder
Neural Machine Translation: Pros / Cons

Quiz: Pros/ Cons for Neural vs. Statistical NMT systems
Neural Machine Translation: Pros / Cons

• Pros
  • End-to-end systems
  • NMT system can handle Word ordering, Morphology, Syntax, and Agreements
  • State-of-the-art results

• Cons
  • NMT needs a larger amount of corpus and resources
  • English centric
Free and Open-Source Translation Models

- Pretrained on sentence pairs in many different languages

**Marian (Helsinki Lab)**
- Mid quality
- Low Inference Latency
- High coverage (555 languages)

**MBART (Meta)**
- High quality
- Mid Inference Latency
- Low coverage (50 languages)

**M2M100/ NLLB (Meta)**
- High quality
- Mid Inference Latency
- Mid coverage (100-200 languages)

Seq2Seq with encoder-decoder transformers
Problem with Metrics

• BLEU (Bilingual Evaluation Understudy)
  • Computes ngram overlap between input and output
  • Measures surface syntax-level similarity not semantics
  • Doesn’t correlate well with human judgment
• More recent metrics (e.g. BertScore) address these
  • Not widely adopted in NMT community
  • Still measures token level similarity
  • Needs finetuning for a new language pair
Precision Needed for Semantic Parsing

• Strict need for accuracy
  • The right ThingTalk construct
  • The right parameters!

Quiz: Can we use semantic parsing as a metric for translation?
Outline

1. Introduction to Machine Translation
2. Localization for Single Commands
3. Localization for Dialogues
Localization for Single Commands
<table>
<thead>
<tr>
<th>Language</th>
<th>Translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>English</td>
<td>look for 5 star restaurants that serve burgers.</td>
</tr>
<tr>
<td>Arabic</td>
<td>ابحث عن مطاعم 5 نجوم التي تقدم الشاورما.</td>
</tr>
<tr>
<td>German</td>
<td>suchen sie nach 5 sterne restaurants, die mautaschen servieren.</td>
</tr>
<tr>
<td>Spanish</td>
<td>busque restaurantes de 5 estrellas que sirvan paella valenciana.</td>
</tr>
<tr>
<td>Persian</td>
<td>به دنبال رستوران های 5 ستاره باشید که جوجه کباب سرو می کنند.</td>
</tr>
<tr>
<td>Finnish</td>
<td>etsi 5 tähden ravintoloita, joissa tarjoillaan karjalanpiirakka.</td>
</tr>
<tr>
<td>Italian</td>
<td>cerca ristoranti a 5 stelle che servono bruschette.</td>
</tr>
<tr>
<td>Japanese</td>
<td>寿司を提供する5つ星レストランを探す。</td>
</tr>
<tr>
<td>Polish</td>
<td>poszukaj 5 gwiazdkowych restauracji, które serwują kotlet.</td>
</tr>
<tr>
<td>Turkish</td>
<td>köfte servis eden 5 yıldızlı restoranları arayın.</td>
</tr>
<tr>
<td>Chinese</td>
<td>搜索卖北京烤鸭的5星级餐厅。</td>
</tr>
</tbody>
</table>
Data Collection Challenges

• **Cost:** data acquisition is expensive
• **Labor:** repeating the same work done for English is inefficient
• **Inclusion:** languages with small user bases do not get attention
• **Localization:** people want to ask about local locations, cuisines ..
Few Multi-Lingual Agent Datasets

• WoZ: Two crowdsource workers converse
  • Costly, error prone
• M2M: synthesize dialogues using a state-machine
  • Done in only one/ two languages (usually English or Chinese)
  • Domain and target language knowledge is needed
Can We Leverage MT & Translate on the Fly?

- Build the system fully in one language (e.g. English)
- Translate any user-facing text to target language at run time

<table>
<thead>
<tr>
<th>System</th>
<th># Languages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alexa</td>
<td>10</td>
</tr>
<tr>
<td>Google Assistant</td>
<td>19</td>
</tr>
<tr>
<td>Siri</td>
<td>21</td>
</tr>
<tr>
<td>Google Translate</td>
<td>109</td>
</tr>
<tr>
<td>MarianMT (Open-source)</td>
<td>555</td>
</tr>
</tbody>
</table>

Quiz: What are the pros and cons?
Can We Leverage MT & Translate on the Fly?

- Build the system fully in one language (e.g. English)
- Translate any user-facing text to target language at run time

Pros:
- Easy & fast integration

Cons:
- Relies on accurate translations
- No way to recover from translation mistakes
- Risky for NLG – cannot verify agent responses

<table>
<thead>
<tr>
<th>System</th>
<th># Languages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alexa</td>
<td>10</td>
</tr>
<tr>
<td>Google Assistant</td>
<td>19</td>
</tr>
<tr>
<td>Siri</td>
<td>21</td>
</tr>
<tr>
<td>Google Translate</td>
<td>109</td>
</tr>
<tr>
<td>MarianMT (Open-source)</td>
<td>555</td>
</tr>
</tbody>
</table>
Experiment: English Schema2QA Dataset

- Schema2QA Restaurants and Hotels
  - English test/validation are collected through crowdsourcing (MTurks)
  - Questions that refer to two fields in schemas
  - Validation: 528  Test set: 524
- Training set: synthetic 508K/2% manual paraphrases
Experiment: Multilingual Schema2QA Dataset

- Validation and Test set: Professional translated from English
- Parameters are replaced with local entities
- ThingTalk is the same as English, except for the parameters

<table>
<thead>
<tr>
<th>English</th>
<th>Italian</th>
</tr>
</thead>
<tbody>
<tr>
<td>I am looking for a burger place near woodland pond.</td>
<td>sto cercando un posto da lasagna vicino a colosseo.</td>
</tr>
<tr>
<td>(@org.schema.Restaurant.Restaurant()), (geo == new Location(&quot;woodland pond&quot;)) &amp;&amp; servesCuisine =~ &quot;burger&quot;)</td>
<td>(@org.schema.Restaurant.Restaurant()), (geo == new Location(&quot;colosseo&quot;)) &amp;&amp; servesCuisine =~ &quot;lasagna&quot;)</td>
</tr>
</tbody>
</table>
Leveraging Neural MT Translation

ITALIAN AGENT

SOTA
Run Time

ThingTalk

English Semantic Parser

English Utterance

Neural Machine Translator

Italian Utterance
(a) Translate at Test Time (BackTranslate)

Inference Time

Italian sentence → NMT → English sentence → Trained English SP → Logical form

Stanford University

FAIL: Absolutely not acceptable!
Translations are not always correct!
Outline

1. Introduction to Machine Translation
2. Localization for Single Commands
   a. Translate on the Fly
   b. Train with Machine-Translated Data
   c. On the Fly with Entity-Aware Translation
   d. Train with automatically generated data with local entities
3. Localization for Dialogues
(b) Let’s Train with Translated Data

Italian Training Data → Train → Italian Semantic Parser

Italian Utterance

Neural Machine Translator

English Training Data

Genie

Run Time

ITALIAN AGENT

ThingTalk
Bootstrap: Train Parser with MT Data

• Translate training data using commercial translators (e.g. Google NMT)
• Crowdsourced translations for validation and testing
• Originally tested on closed ontology datasets with simple formal language constructs (Overnight and ATIS)
• +English: include English training data as well

(b) Train on Machine-Translated Data (Bootstrap)

- Results usually improve although not much

---

**Training Time**

<table>
<thead>
<tr>
<th>Language</th>
<th>English sentence</th>
<th>NMT</th>
<th>Italian sentence (+ English sentence)</th>
<th>Italian SP</th>
<th>Logical form</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arabic</td>
<td>@org.schema.Restaurant.Restaurant(), geo == new Location(&quot;stagno bosco&quot;) &amp; servesCuisine == &quot;burger&quot;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>German</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spanish</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Persian</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Finnish</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Italian</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Japanese</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Polish</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Turkish</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chinese</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

FAIL: Training does not help! Why?
Common Problems with Naive Translation

- **Literal translation of entities:**
  - **I am looking for a burger place near woodland pond.**
  - **Sto cercando un posto da hamburger vicino a stagno bosco.**

- **Transliteration of entities:**
  - **I am looking for a starbucks which is open before 13:30.**
  - **من به دنبال یک استارباکس هستم که قبل از ساعت ۱۳:۳۰ باید باشد.**

- **Dropping entities:**
  - **Find a hotel nearby that serves breakfast, has more than 9 reviews, and is located near palo alto.**
  - **Yakınılarda kahvaltı servisi yapan, 9'dan fazla yorumu olan ve bulunduğu bir otel bulun.**

- **Mistranslating entities:**
  - **What seasons did steph curry have only 2 blocks?**
  - **斯蒂芬·库里什么季节只有2个街区?**

---

A perfect parser will provide a good metric for translation!
## Handling Local Entities

I want a hotel near times square that has at least 1000 reviews

<table>
<thead>
<tr>
<th>Arabic</th>
<th>Arabic Translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>أريد فندقًا بالقرب من مسجد الحرم يحتوي على 1000 مراجعة على الأقل.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>German</th>
<th>German Translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ich möchte ein hotel in der nähe von marienplatz, das mindestens 1000 bewertungen hat</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Spanish</th>
<th>Spanish Translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Busco un hotel cerca de puerto banús que tenga al menos 1000 comentarios.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Persian</th>
<th>Persian Translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>من هتلی در نزدیکی باخ ارم می‌خواهم که حداقل 1000 بازیبینی داشته باشد.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Finnish</th>
<th>Finnish Translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Haluan paikan helsingin tuomiokirkko läheeltä hotellin, jolla on vähintään 1000 arvostelua.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Italian</th>
<th>Italian Translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voglio un hotel nei pressi di colosseo che abbia almeno 1000 recensioni.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Japanese</th>
<th>Japanese Translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>東京スカイツリー周辺でに1000件以上のレビューがあるホテルを見せて。</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Polish</th>
<th>Polish Translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Potrzebuję hotelu w pobliżu zamek w malborku, który ma co najmniej 1000 ocen.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Turkish</th>
<th>Turkish Translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kapalı carşı yakınlarında en az 1000 yorumu sahip bir otel istiyorum.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chinese</th>
<th>Chinese Translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>我想在天安门广场附近找一家有至少1000条评论的酒店。</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Annotations are the same (modulo entities)</th>
</tr>
</thead>
</table>

( @org.schema.Hotel.Hotel ) filter
param:aggregateRating.reviewCount: Number >= 1000 and
param:geo:Location
== location: " [entity] "

Stanford University
Outline

1. Introduction to Machine Translation
2. Localization for Single Commands
   a. Translate on the Fly
   b. Train with Machine-Translated Data
   c. On the Fly with Entity-Aware Translation
   d. Train with automatically generated data with local entities
3. Localization for Dialogues
(c) Entity-Aware Translation

- Preserve the entity across translation

**English**

I am looking for a **burger** place near **woodland pond**.

**Naïve Italian Translation**

Sto cercando un posto da "**hamburger**" vicino a "**stagno bosco**".

**ThingTalk**

```c
@restaurant(),
geo == near("Woodland Pond")
& servesCuisine =~ "burger"
```

**Italian Entity-Aware Translation**

Sto cercando un posto da "**burger**" vicino a "**woodland pond**".
Translate with Alignment

A(x,y): token y aligns with token x

-10Less

0More

I am looking for a burger place near woodland pond.

Sto cercando un posto da "burger" vicino a "woodland pond".

Translator
Adding Alignment to NMT

Goal: Preserve entities during translation

• Assumptions:
  • Entities are continuous spans of text
  • Entity spans in the sentence and annotation match

• Method:
  • Use cross-attention weights
    These are learned weights between decoder and encoder layers
  • For each input span,
    retrieve a unique output span with highest cross-attention value
  • Finally, replace the output spans with input spans
(c) Translate & Align at Run-time (BackTranslation)

- Results improve across the board by 25% to 40%
- Highlights importance of having aligned entities in the sentence and the logical form

Inference Time

Promising, entities are fixed now, but how do we do better?
Outline

1. Introduction to Machine Translation
2. Localization for Single Commands
   a. Translate on the Fly
   b. Train with Machine-Translated Data
   c. On the Fly with Entity-Aware Translation
   d. Train with automatically generated data with local entities
3. Localization for Dialogues
(d) SPL: Semantic Parser Localizer

- Train with local entities!

- Translate English data sets into target language and use entities local to the target language
Generating Training Data with Local Entities

ENGLISH
I am looking for a burger place near woodland pond.

ITALIAN
Sto cercando un posto da "hamburger" vicino a "stagno bosco".

THINGTALK
@restaurant()
geo == near("Woodland Pond")
&& servesCuisine =~ "burger"

Parameter Replacement
Sto cercando un posto da "hamburger" vicino a "stagno bosco".

Aligner
Sto cercando un posto da "burger" vicino a "woodland pond".

Neural Machine Translator
Sto cercando un posto da "bruschette" vicino a "venezia".

@restaurant()
geo == near("venezia")
&& servesCuisine =~ "bruschette"
(d) Train on Translated & Aligned Data (SPL)

- Model trained on translated data is more robust to noise
- Model learns to copy entities and can handle unseen ones
- Adding English usually helps by exposing parser to a larger vocabulary and different grammars

Training Time

English sentence

I am looking for a burger place near woodland pond.

SPL

substitutes entities in NL & TT

Italian sentence (+ English sentence)

Sto cercando un posto da lasagna vicino a colosseo.

Italian SP

Logical form

(@org.schema.Restaurant.Restaurant()).
(geo == new Location("colosseo") &&
ervesCuisine == "lasagna")
Add a Few-Shot of Human Translated Data

- Add human-translated validation data as few-shot
- Close to few-shot English; beats zero-shot

---

**Training Time**

- **English sentence**: I am looking for a burger place near woodland pond.
- **SPL**
  - substitutes entities in NL & TT
- **Italian sentence**
  - + a few human-translated sentences
  - Sto cercando un posto da lasagna vicino a colosseo.
- **Italian SP**
  - (@org.schema.Restaurant.Restaurant(),
    (geo == new Location("colosseo")) &&
    servesCuisine =~ "lasagna")
Similar Trend for the Hotels Domain

This looks good!
Outline

1. Introduction to Machine Translation
2. Localization for Single Commands
   a. Translate on the Fly
   b. Train with Machine-Translated Data
   c. On the Fly with Entity-Aware Translation
   d. Train with automatically generated data with local entities
3. Localization for Dialogues
Localization for Dialogues
Dialogues

👤: Hi, can you help me find a place to eat?
👉: Sure! How much do you want to spend and how high of a rating would you prefer?
👤: I'd like to eat at an expensive restaurant rated at least 9.
👉: Got it. What kind of food do you want?
👤: Any type of food is fine, but I want a place with Vegan Options.
Challenges Unique to Dialogues

- Utterances are generally longer and more complex containing multiple entities.
Challenges Unique to Dialogues

• Utterances are generally longer and more complex containing multiple entities.

Quiz: What else?
Challenges Unique to Dialogues

• Utterances are generally longer and more complex containing multiple entities.
• Translation errors accumulate over turns and can prevent a correct parse for the rest of the dialogue.
• There are logical dependencies between slot values across different turns.
  • Cannot arbitrarily substitute entities in Semantic Parser Localizer (SPL).
• Agent utterances are shown to the user and must be of higher quality.
Translating the Dialogue Dataset

• User and Agent utterances are translated separately
• Some slot values are logically dependent
  (e.g. price-range for a “fast-food” restaurants should be “cheap”)
  • These are translated by dictionary lookup
• Alignment works poorly for numbers/ dates/ time slots
  • We use heuristics to retrieve those spans using a dictionary
• NMT quality is better on shorter texts with fewer entities
  • Break down utterances into individual sentences before translation.
Neural Alignment

我想找个中等价位的餐厅

• Alignment maps input entity to an incorrect span in the output

"中等": “with a medium price”

Correct Translation:
I am looking for a restaurant with a medium price.
I am looking for a restaurant with a 中等 price.

- Use beam search to generate multiple translations for each entity
- Perform string matching using any of possible translations
- If unsuccessful, resort to neural alignment
End-to-End Dialogue Systems

- Has 3 separate subtasks: DST, Policy, NLG
  - DST for natural language understanding
  - Policy for managing agent actions
  - NLG to generate agent response

- Traditional approaches train a single model for each subtask
SOTA Approach

- Recent approaches train one model for all subtasks (e.g. SimpleTOD, BiToD)
- More robust to noise
- Better error recovery
- Transfer learning between different subtasks

Challenges

• How to get multilingual dialogue datasets?
  • Translate original dataset to other languages
• How to ensure robust translation for task-orientated dialogue?
  • Translation is noisy
  • Entities will be translated
  • Annotation and translation errors compound over multiple turns
Proposal: Concise data representation

• Reduce amount of natural language encoded at each turn:
  • Only input necessary information for correct prediction for each subtask
  • Replace agent responses with formal dialogue acts
• Replace history with dialogue state:
  • In practice, because of annotation errors, user’s change of mind, etc. state doesn’t always include all relevant slots
  • To mitigate include last two formal agent responses
Dialogue Loop for one turn

- **DST**: Parse user utterance
- **ACD**: Make an API call if needed
- **DAG**: Generate agent dialogue acts
- **RG**: Generate agent response
## Dataset (DST only)

- **MultiWOZ**
  - Originally in English
  - Translations in Chinese available
  - Bad annotations quality

- **CrossWOZ**
  - Originally in Chinese
  - Translations in English available

- **RisaWOZ**
  - Released in Chinese
  - Longer conversations
  - More complex dialogue goals
  - Good annotation quality
  - We translated it (with alignment) into English and German

<table>
<thead>
<tr>
<th></th>
<th>RiSAWOZ</th>
<th>CrossWOZ</th>
<th>MultiWOZ 2.1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Languages</strong></td>
<td>ZH, EN*, DE*</td>
<td>ZH, EN</td>
<td>EN, ZH</td>
</tr>
<tr>
<td><strong># Domains</strong></td>
<td>12</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td><strong># Dialogues</strong></td>
<td>10,000</td>
<td>5,012</td>
<td>8,438</td>
</tr>
<tr>
<td><strong># Turns</strong></td>
<td>134,580</td>
<td>84,692</td>
<td>115,424</td>
</tr>
<tr>
<td><strong># Slots</strong></td>
<td>159</td>
<td>72</td>
<td>25</td>
</tr>
<tr>
<td><strong># Values</strong></td>
<td>4,061</td>
<td>7,871</td>
<td>4,510</td>
</tr>
</tbody>
</table>

*: auto-translated using our approach
Our Semantic Parser: BART-CSP

- Fine-tune MBART
- Contextual Semantic Parser (CSP)
- Input is:
  - Current user utterance
  - Last agent response
  - Latest Belief state
- Output is
  - the updated belief state
- Everything is encoded directly as text to make the most use of Seq2Seq model pretraining

```
User utterance: Is that so? What is the recommended dish?
Agent response: Old Maple Garden is a good one.
State: restaurants Price = "cheap" restaurants Name = "Old Maple Garden" restaurants Cuisine = "western"
```

```
New state: restaurants Price = "cheap" restaurants Name = "Old Maple Garden" restaurants Cuisine = "western"
```

```
Seq2Seq Model
```

```
```

Stanford University
Baseline Models

- **TRADE**
  - Seq2Seq with pointer-generator
  - Encodes fill dialogue history

- **MLCSG**
  - Extends TRADE by improving modeling of long contexts

- **SOM**
  - Uses state operation to selectively update slot values at each turn

- **SUMBT**
  - First model to use Transformers (BERT) as Encoder
  - Showed improvement over previous RNN-based models

- **STAR**
  - SOTA on MultiWOZ.
  - Uses two BERT models for encoding context and slot values
  - Encodes both the previous belief state and history of dialogue.

- **BART**
  - Sequence-to-sequence denoising auto-encoder
Metrics

• **Joint Goal Accuracy (JGA):**
  • Standard metric for DST evaluation
  • Average accuracy of predicting all slot assignments correctly for any given turn
  • Predicted belief state in previous turn is used as input for the current turn

• **Gold Joint Goal Accuracy (GJGA):**
  • Similar to JGA but ground-truth belief state is used as input
  • Removes the compounding effect of errors from previous turns in calculation

Quiz: Which metric is more realistic and should be used in practice?
Results

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Model</th>
<th>Context</th>
<th>Dialogue</th>
<th>Encodes</th>
<th>Predefined</th>
<th>GJGA</th>
<th>JGA</th>
<th>GJGA</th>
<th>JGA</th>
</tr>
</thead>
<tbody>
<tr>
<td>RiSAWOZ</td>
<td>MLCG</td>
<td>Bi-GRU</td>
<td>Full</td>
<td>x</td>
<td></td>
<td>66.2</td>
<td>90.4</td>
<td>76.9</td>
<td></td>
</tr>
<tr>
<td>RiSAWOZ-EN-auto</td>
<td>MLCG</td>
<td>Bi-GRU</td>
<td>Full</td>
<td>x</td>
<td></td>
<td>36.2</td>
<td>88.8</td>
<td>68.6</td>
<td></td>
</tr>
<tr>
<td>(-alignment)</td>
<td>MLCG</td>
<td>Bi-GRU</td>
<td>Full</td>
<td>x</td>
<td></td>
<td>15.6</td>
<td>65.2</td>
<td>22.9</td>
<td></td>
</tr>
<tr>
<td>RiSAWOZ-DE-auto</td>
<td>MLCG</td>
<td>Bi-GRU</td>
<td>Full</td>
<td>x</td>
<td></td>
<td>13.4</td>
<td>86.7</td>
<td>65.9</td>
<td></td>
</tr>
<tr>
<td>CrossWOZ</td>
<td>TRADE</td>
<td>Bi-GRU</td>
<td>Full</td>
<td>x</td>
<td>✓</td>
<td>71.3¹</td>
<td>36.1</td>
<td>80.2</td>
<td>53.6</td>
</tr>
<tr>
<td>CrossWOZ-EN</td>
<td>SOM</td>
<td>BERT</td>
<td>Partial</td>
<td>✓</td>
<td>x</td>
<td>32.3²</td>
<td>81.1</td>
<td>52.3</td>
<td></td>
</tr>
<tr>
<td>MultiWOZ 2.1</td>
<td>MinTL</td>
<td>BART-large</td>
<td>Partial</td>
<td>✓</td>
<td>x</td>
<td>53.6³</td>
<td>81.2</td>
<td>53.7</td>
<td></td>
</tr>
<tr>
<td>MultiWOZ 2.1</td>
<td>STAR</td>
<td>BERT</td>
<td>Full</td>
<td>✓</td>
<td>✓</td>
<td>56.7</td>
<td>81.2</td>
<td>53.7</td>
<td></td>
</tr>
<tr>
<td>MultiWOZ-ZH 2.1</td>
<td>SUMBT</td>
<td>BERT</td>
<td>Full</td>
<td>✓</td>
<td>✓</td>
<td>46.0⁴</td>
<td>75.9</td>
<td>46.3</td>
<td></td>
</tr>
</tbody>
</table>

• Alignment is useful even for closed ontology and dialogue datasets (22.9% to 68.6%)
• We can create high-quality large-scale dialogue datasets in other languages using machine translation with alignment.
  • only 8% drop in accuracy compared to the original
• Accumulation of translation errors across turns is mitigated with CSP
  • 11-30% JGA improvements on CrossWOZ and RiSAWOZ
Results

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Model</th>
<th>Context Encoder</th>
<th>Dialogue History</th>
<th>Encodes State</th>
<th>Predefined Slots/ Ont.</th>
<th>GJGA</th>
<th>JGA</th>
<th>GJGA</th>
<th>JGA</th>
</tr>
</thead>
<tbody>
<tr>
<td>RiSAWOZ</td>
<td>MLCGR</td>
<td>Bi-GRU</td>
<td>Full</td>
<td>✓</td>
<td></td>
<td>66.2</td>
<td>90.4</td>
<td>76.9</td>
<td></td>
</tr>
<tr>
<td>RiSAWOZ-EN-auto</td>
<td>MLCGR</td>
<td>Bi-GRU</td>
<td>Full</td>
<td>✓</td>
<td></td>
<td>36.2</td>
<td>88.8</td>
<td>68.6</td>
<td></td>
</tr>
<tr>
<td>(-alignment)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RiSAWOZ-DE-auto</td>
<td>MLCGR</td>
<td>Bi-GRU</td>
<td>Full</td>
<td>✓</td>
<td></td>
<td>15.6</td>
<td>65.2</td>
<td>22.9</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td>13.4</td>
<td>86.7</td>
<td>65.9</td>
<td></td>
</tr>
<tr>
<td>CrossWOZ</td>
<td>TRADE</td>
<td>Bi-GRU</td>
<td>Full</td>
<td>✓</td>
<td>71.3†</td>
<td>36.1</td>
<td>80.2</td>
<td>53.6</td>
<td></td>
</tr>
<tr>
<td>CrossWOZ-EN</td>
<td>SOM</td>
<td>BERT</td>
<td>Partial</td>
<td>✓</td>
<td></td>
<td>32.3‡</td>
<td>81.1</td>
<td>52.3</td>
<td></td>
</tr>
<tr>
<td>MultiWOZ 2.1</td>
<td>MinTL</td>
<td>BART-large</td>
<td>Partial</td>
<td>✓</td>
<td>-</td>
<td>53.6‡</td>
<td>81.2</td>
<td>53.7</td>
<td></td>
</tr>
<tr>
<td>MultiWOZ 2.1</td>
<td>STAR</td>
<td>BERT</td>
<td>Full</td>
<td>✓</td>
<td>78.7*</td>
<td>56.7</td>
<td>81.2</td>
<td>53.7</td>
<td></td>
</tr>
<tr>
<td>MultiWOZ-ZH 2.1</td>
<td>SUMBT</td>
<td>BERT</td>
<td>Full</td>
<td>✓</td>
<td></td>
<td>46.0‡</td>
<td>75.9</td>
<td>46.3</td>
<td></td>
</tr>
</tbody>
</table>

- MultiWOZ has poor annotations
  - History offers a chance to recover
    - SOTA is better by 3% on MultiWOZ (53.7% → 78.7%)
  - When the past turns of a dialogue have been predicted correctly CSP does better
    - GJGA improves by 2.5% on MultiWOZ (78.7% → 81.2%)
- Annotation errors can misguide research direction!
Dataset (End-to-End experiments)

• BiToD
  • In Chinese and English
  • Including 5 domains
  • Constructed using simulation (M2M)
  • Then Human paraphrased for fluency
  • Data examples are not parallel

<table>
<thead>
<tr>
<th></th>
<th>BiToD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Language(s)</td>
<td>EN, ZH</td>
</tr>
<tr>
<td>Number of dialogues</td>
<td>5,787</td>
</tr>
<tr>
<td>Number of domains</td>
<td>5</td>
</tr>
<tr>
<td>Number of APIs</td>
<td>7</td>
</tr>
<tr>
<td>Total number of turns</td>
<td>115,638</td>
</tr>
<tr>
<td>Average turns / dialogues</td>
<td>19.98</td>
</tr>
<tr>
<td>Slots</td>
<td>68*</td>
</tr>
<tr>
<td>Values</td>
<td>8,206*</td>
</tr>
<tr>
<td>Deterministic API</td>
<td>✓</td>
</tr>
<tr>
<td>Complex User Goal</td>
<td>✓</td>
</tr>
<tr>
<td>Mixed-Language Context</td>
<td>✓</td>
</tr>
<tr>
<td>Provided KB</td>
<td>✓</td>
</tr>
</tbody>
</table>
Metrics

• **Joint Goal Accuracy (JGA):** 1 if all slot-relation-value triplets are correct
• **Task Success Rate (TSR):** Agent correctly provides all the user-requested information for that task.
• **Dialogue Success Rate (DSR):** 1 for a dialogue if all user requests are completed successfully
• **API:** 1 if the model correctly predicts to make an API call, and all the constraints provided for the call are correct
• **BLEU:** Measures the natural language response fluency based on n-gram matching with gold response
• **Slot Error Rate (SER):** 1 if the response contains all entities present in the gold response
Experiment Settings

• Access to full training data in the source language

• Full-shot:
  • Access to full training data in the target language

• Zero-shot:
  • No training data in the target language

• Few-shot:
  • Access to X% of training data in the target language
Full-shot results

- Use all of the training data in English
- 54% DSR
Pre-train on Chinese data

- Train a model on fewshot data in the target language
Use few-shot data

- Finetune on different amounts of training data; the more data the better
- Zero-shot achieves 0% accuracy! Understanding of target language entities is a must
Canonicalize slots, domains, and acts

- Brings training data closer to the vocabulary of test data
- No cost since translation is automatic using a dictionary
Canonicalize slots, domains, and acts

<table>
<thead>
<tr>
<th></th>
<th>JGA</th>
<th>TSR</th>
<th>DSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>+ Language</td>
<td>2.13%</td>
<td>1.20%</td>
<td>0.0</td>
</tr>
<tr>
<td>Pretraining</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+ Canonicalization</td>
<td>14.73%</td>
<td>3.52%</td>
<td>1.58</td>
</tr>
</tbody>
</table>

- No cost since translation is automatic using a dictionary
- Improves metrics significantly for zero-shot
Pre-train on large amount of translated data

Naive translation might even hurt accuracy!
Pre-train on Translated data

Quiz: In general, naive translation performs much worse for dialogues than for single turn commands. What do you think is the reason?
Use alignment for translation

- Alignment improves accuracy across the board
- Much needed for zero-shot!
Filter poor agent responses

- Increases data quality for RG
- Affects other subtasks too
Summary

• Building agents in new languages is possible using machine translation
  • We have created the first usable end-to-end dialogue agent in a zero-shot manner
  • Concise data representation is key to improved performance and data efficiency
  • In fact, our 10% few-shot models beats previous SOTA train on 100% training data!
Building high quality dialogue dataset

• How to curate validation/ test data for languages?
  • we want to measure performance of different techniques

• Extend RiSAWOZ dataset to new languages
• First effort to build end-to-end dialogue datasets;
  • previous work focused only on DST
• Automatic translation
  • + human post-editing for few-shot, validation, and test data
• Lots of manual effort needed to refine and improve quality of the dataset
Building high quality dialogue datasets

- Extend RiSAWOZ dataset to new languages
- Automatic translation
  - + human post-editing for few-shot, validation, and test dat
- Partnering with several universities and institutions to make this happen!
Conclusions

• Entities are important!  
  Machine Translation with Alignment is necessary
• Human fewshots can improve accuracy by closing data gap
• Dialogues:
  • Efficient data representation is important
  • Replace natural language with formal language where possible
• A toolkit to build question-answering and dialogue systems  
  for a new languages