Dialogue Generation from Structured Knowledge

presented by: Andrea Zugarini

Introduction

Recent advances in Natural Language Processing, due to big data and deep learning techniques, favoured the development of many virtual assistants, aka **conversational agents**.

There are two main lines of research:

- **Task-oriented** systems
- **Open-domain** models

Introduction

Task-oriented

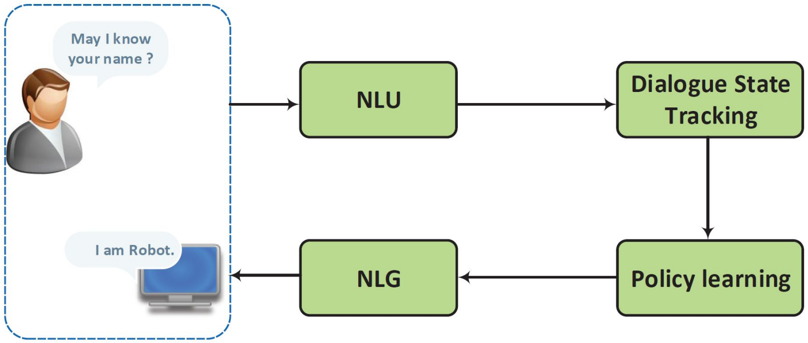
- Assist users in accomplish a specific task (booking a restaurant)
- Response is managed through a pipeline of modules

Open-domain

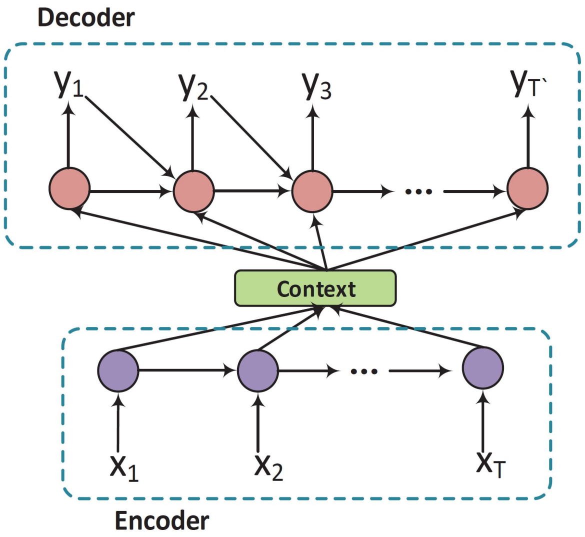
- Main purpose of those systems is user entertainment
- Perform **chit-chat** which is an important component in most conversation scenarios

Introduction

Task-oriented



Open-domain



Pros & Cons

Task-oriented

+ Good performances on the addressed task

- Domain dependent models are hard to transfer into other tasks
- Although statistical models are applied, **handcrafted** rules are often employed

Open-domain

+ Ease of development
+ Capture some dialogue properties

- Poor performances on factual knowledge or specific tasks
- Agents are black boxes that cannot be fixed
- *Response Diversity*: tendency of responding with little mean answers. E.g. **“I don’t Know”**

The Role of Knowledge

Prior knowledge is a crucial component in dialogues. Humans communicate on the base of the information they have.

Open-domain conversational agents rely only on dialogues examples, without making use of other knowledge.
Whereas **task-oriented** models usually have a (small) database to query in order to accomplish the task, knowledge is still very simple, domain dependent, and the dialogue generation is too driven by the problem.

So most of the dialogue systems do not make use of external knowledge.

How to link Knowledge with natural language?

Three approaches are presented here, all aiming to generate textual data from different kind of knowledge:

- **Knowledge-Driven Response Generation** exploits wikipedia data to provide context-sensitive utterances in dialogues
- **Generating Textual Summaries from KB Triples** Semantic Web data, made of triples, is converted into natural language text to describe an entity involved in such facts
- **Learning Dialogue Agents with KB** is a work in which two agents, with an internal structured knowledge, learn to dialogue by communicating to accomplish a common task

Knowledge-Driven Response Generation*

- Assumption that humans in a conversation base their utterances not only on previous dialog response, but also on their **background knowledge**
- Task is to generate a context-sensitive response to a sequence of comments
- Model exploits Wikipedia data as the agent's knowledge, which is aligned with Reddit comments
- RNN and CNN based approach

* See Ref. [2] in References slide

Knowledge-Driven Response Generation

Dataset

- Collected Wikipedia pages and Reddit comments about 35 topic-keywords in philosophy and literature
- A sequence of comments is associated with 20 sentences extracted from wikipedia page summary
- About 15k sequence of comments and 75k wikipedia sentences
- Vocabulary of 56k words

Knowledge-Driven Response Generation

Model

Sentence Modelling (Wikipedia's summaries)

background knowledge is incorporated through sentence-level CNN processing a group of sentences S

Sequence Modelling (Reddit's comments)

word-level RNN elaborate the sequence of comments

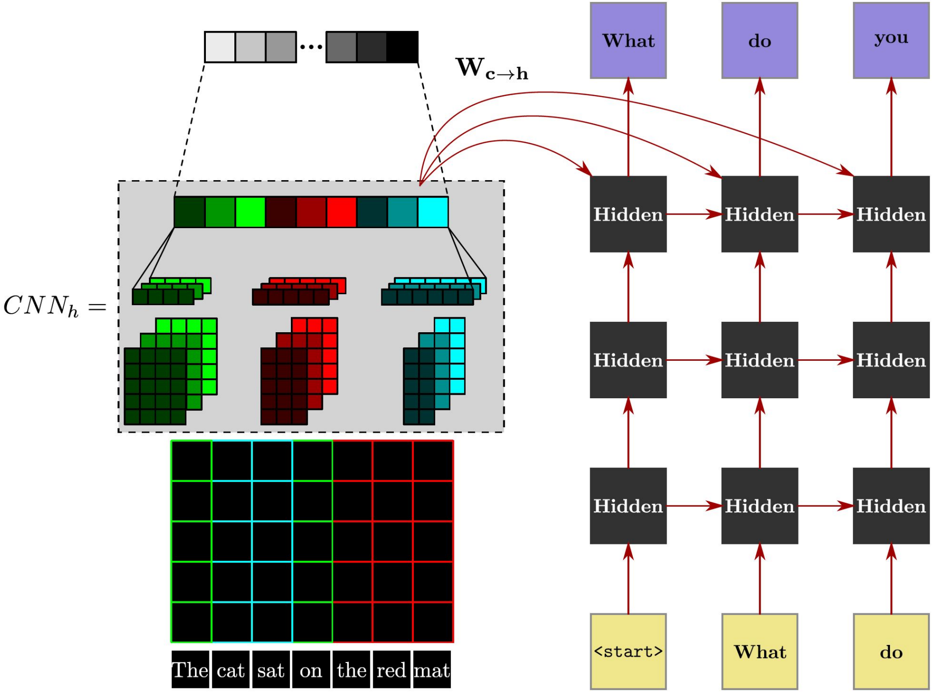
Coupling

context-sensitive information (CNN features) provided at each timestep on the RNN last layer

Knowledge-Driven Response Generation

Model

$$
\begin{aligned}
c_S &= \mathbf{W_{c\rightarrow h}} CNN_h(S) \quad , \\
h_t &= h_t^L \odot c_S \quad , \\
y_t &= softmax(\mathbf{W_y} h_t).
\end{aligned}
$$



Knowledge-Driven Response Generation

Experiments

CNN is pre-trained, with a subsample of 30k wikipedia sentences, to predict the class of the input sentence over the 35 topic-keywords.
Narrow convolutional filters of widths 3,4,5 and 6 with 300 feature maps each.

Experiments conducted with two recurrent models: **LSTM, GRU**. In both cases, they used 2 layers of 1000 cells each.

Model	LSTM	LSTM Coupled With ConvNet	GRU	GRU Coupled With ConvNet
Perplexity	4.301	**1.905**	3.749	2.051
Average Rating (σ)	**2.65** ($\pm$1.167)	2.4 ($\pm$1.27)	2.5 ($\pm$1.359)	**2.65** ($\pm$1.561)

Generating Textual Summaries from KB Triples*

- Semantic web data require textual or visual interfaces to allow their interpretation
- The task is to generate Natural language summaries from a set of triples
- Same approach may be useful for other applications (e.g. conversational agents)
- **Triples** are encoded into a vector of fixed dimensionality
- Wikipedia snippets are loosely aligned with DBpedia and Wikidata triples to train and evaluate the model

* See Ref. [3] in References slide

Generating Textual Summaries from KB Triples

Triples	dbr:Walt_Disney dbo:birthDate ''1901-12-05'' dbr:Walt_Disney dbo:birthPlace dbr:Chicago dbr:Mickey_Mouse dbo:creator dbr:Walt_Disney
Textual Summary	Walt Disney was born in Chicago, and was the creator of Mickey Mouse.

Generating Textual Summaries from KB Triples

Datasets

- Wikipedia summaries are aligned to a structured knowledge base
- Two different datasets are generated using different KBs: DBpedia and Wikidata

Triples	`dbr:Walt_Disney dbo:birthDate ``1901-12-05''` `dbr:Walt_Disney dbo:birthPlace dbr:Chicago` `dbr:Mickey_Mouse dbo:creator dbr:Walt_Disney`
Textual Summary	Walt Disney was born in Chicago, and was the creator of Mickey Mouse.

Table 4: Example of the alignment of our dataset. One Wikipedia summary is coupled with a set of triples from either DBpedia or Wikidata. Any reference to the main discussed entity of the summary (i.e.`dbr:Papa_Roach` or `wikidata:Q254371` respectively) is replaced by the special `<item>` token both in the text and the corresponding triples. Each other entity is stored along with its instance type. In the case of infrequent entities these are replaced with the special token of their instance types both in the text and the triples (e.g. "triple platinum" is replaced with `dbr:RIAA_certification`). When a rare entity in the text is matched to an entity of the corresponding triples' set, then it is replaced by a unique token, which consists from the predicate of the triple, a descriptor of the component of the triple that was matched, and the instance type of the entity (e.g. the reference to the music album "Infest (2000)" is replaced with the placeholder `[dbo:artist__sub__dbo:Album]`).

<item>	dbr:Papa_Roach and wikidata:Q254371
Original Wikipedia Summary	Papa Roach is an American rock band from Vacaville, California. Formed in 1993, their first major-label release was the triple-platinum album Infest (2000).
DBpedia Triples	<item> dbo:bandMember dbr:Jacoby_Shaddix [dbo:MusicalArtist] <item> dbo:bandMember dbr:Jerry_Horton [dbo:MusicalArtist] <item> dbo:genre dbr:Hard_rock [dbo:MusicGenre] ⋮ <item> dbo:hometown dbr:United_States [dbo:Country] <item> dbo:hometown dbr:Vacaville,_California [dbo:City] [dbo:Album] dbr:Infest_(album) dbo:artist <item> [dbo:Album] dbr:Metamorphosis_(Papa_Roach_album) dbo:artist <item>
Summary /w URIs	<start> <item> is an dbr:United_States dbr:Rock_music band from [dbo:hometown__obj__dbo:City] . Formed in <year> , their first major-label release was the dbr:RIAA_certification album [dbo:artist__sub__dbo:Album] (<year>) . <end>
Summary /w Surface Form Tuples	<start> <item> is an (dbr:United_States, American) (dbr:Rock_music, rock) band from [dbo:hometown__obj__dbo:City] . Formed in <year> , their first major-label release was the dbr:RIAA_certification album [dbo:artist__sub__dbo:Album] (<year>) . <end>
Wikidata Triples	<item> wikidata:P136 wikidata:Q11399 [dbo:MusicGenre] <item> wikidata:P495 wikidata:Q30 [dbo:Country] <item> wikidata:P571Month 1 [<unk>] <item> wikidata:P571Year <year> [<unk>] <item> wikidata:P31 wikidata:Q215380 [<unk>] <item> wikidata:P264 wikidata:Q212699 [dbo:RecordLabel] ⋮ [dbo:Album] wikidata:Q902353 wikidata:P175 <item>
Summary /w URIs	<start> <item> is an wikidata:Q30 wikidata:Q11399 band from dbo:City . Formed in <year> , their first <rare> release was the wikidata:Q2503026 album [wikidata:P175__sub__dbo:Album] (<year>) . <end>
Summary /w Surface Form Tuples	<start> <item> is an (wikidata:Q30, American) (wikidata:Q11399, rock) band from dbo:City . Formed in <year> , their first <rare> release was the <unk> album [wikidata:P175__sub__dbo:Album] (<year>) . <end>

Generating Textual Summaries from KB Triples

Model

Given E facts involving an entity:

$$F = \{f_1, f_2, \ldots, f_E : f_i = (s_i, p_i, o_i)\}$$

Encoder

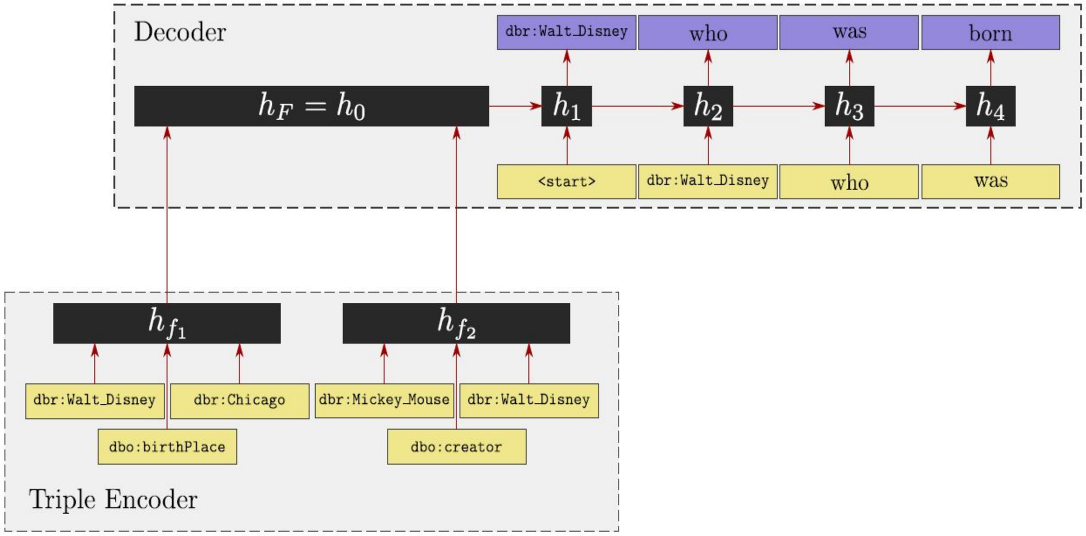
$$\widetilde{h_{f_i}} = [\mathbf{W}_{\mathbf{x}\to\tilde{\mathbf{h}}} s_i; \mathbf{W}_{\mathbf{x}\to\tilde{\mathbf{h}}} p_i; \mathbf{W}_{\mathbf{x}\to\tilde{\mathbf{h}}} o_i] \ ,$$

$$h_{f_i} = \mathrm{ReLU}(\mathbf{W}_{\tilde{\mathbf{h}}\to\mathbf{h}} \widetilde{h_{f_i}}) \ ,$$

Decoder

$$\widetilde{h_F} = [h_{f_1}; h_{f_2}; \ldots; h_{f_{E-1}}; h_{f_E}] \ ,$$

$$h_F = \mathbf{W}_{\mathbf{h_F}\to\mathbf{h}_0^1} \widetilde{h_F} \ ,$$



Generating Textual Summaries from KB Triples

Experiments

Generating Textual Summaries from KB Triples

Results

<item>	dbr:Barbara_Flynn
Triples	<item> dbo:birthPlace dbr:England [owl#Thing] <item> dbo:birthPlace dbr:St_Leonards-on-Sea [dbo:Settlement] <item> dbo:birthPlace dbr:Sussex [owl#Thing] <item> dbo:occupation dbr:Actress [<unk>] <item> dbo:birthDateMonth 8 [<unk>] <item> dbo:birthDateYear <year> [<unk>] [dbo:TelevisionShow] dbr:Open_All_Hours dbo:starring <item> [dbo:TelevisionShow] dbr:A_Very_Peculiar_Practice dbo:starring <item> [dbo:TelevisionShow] dbr:The_Beiderbecke_Trilogy dbo:starring <item> ⋮ [dbo:TelevisionShow] dbr:Cracker_(UK_TV_series) dbo:starring <item>

Triples2LSTM w/ Surf. Form Tuples (Final)	<start> Barbara Flynn (born 0 August <year>) is an English actress . She is best known for her role as dbo:SoapCharacter in the BBC soap opera EastEnders . <end>

Learning Dialogue Agents with KB*

- Humans communicate through natural language
- Any person has its own knowledge, with different expertise
- Lead by these principles two agents, each with private knowledge, are trained to dialogue in order to achieve a common goal
- KB is structured and also dynamic, since it evolves during the conversation

* See Ref. [4] in References slide

Learning Dialogue Agents with KB

Scenario

- Two agents, A and B, must communicate to find their mutual friend
- Each agent has its own **structured Knowledge Base**, composed of a list of *items* made of several *attributes* values
- *Items* are friends, *attributes* are name, school, company etc…
- Agents KBs changes from dialogue to dialogue, varying randomly number of items, attribute types and values. There is only a single shared item
- Same task faced by people on AMT, where about 9k conversations were collected

Learning Dialogue Agents with KB

Model

The model is made of 3 main components:

A. Dynamic Knowledge Graph
B. Graph Embedding
C. Utterance Generator

Learning Dialogue Agents with KB

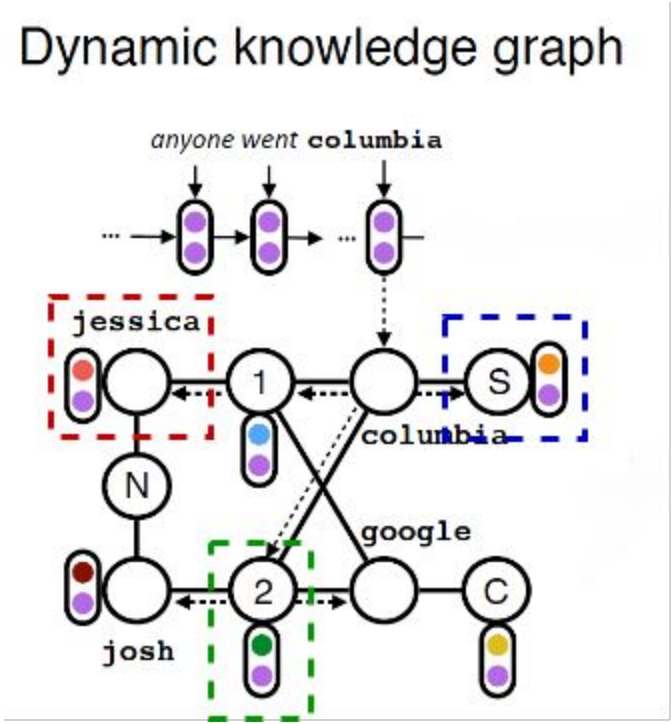
Dynamic Knowledge Graph

Three kinds of node:

item, **entity** (values), **attribute** (type)

Edges represent relations.

Given a dialogue of T utterances, G_t is the graph at time step t.



Learning Dialogue Agents with KB

Graph Embeddings

A vector $V_t(v)$ associated to each node v at time t.

$V_t(v)$ is the concatenation of several features:

- **Node features**: simple structural features of the graph codified as 1 hot vectors: e.g. high-frequency entity, node degree, entity type etc.. . All concatenated in $F_t(v)$
- **Mention vectors**: vector $M_t(v)$ contains unstructured context from utterances relevant to node v up to turn t.
  $$M_t(v) = \lambda_t M_{t-1}(v) + (1 - \lambda_t)\tilde{u}_t;$$
- **Recursive node embeddings**:
  $$V_t^k(v) = \max_{v' \in N_t(v)} \tanh \left(W^{\text{mp}} \left[V_t^{k-1}(v'), R(e_{v \to v'}) \right] \right),$$

Learning Dialogue Agents with KB

Graph Embeddings

A vector $V_t(v)$ associated to each node v at time t. $V_t(v)$ is the concatenation of several features:

- **Node features**: simple structural features of the graph codified as 1hot vectors: e.g. high-frequency entity, node degree, entity type etc.. . All concatenated in $F_t(v)$
- **Mention vectors**: vector $M_t(v)$ contains unstructured context from utterances relevant to node v up to turn t.

  $$M_t(v) = \lambda_t M_{t-1}(v) + (1 - \lambda_t)\tilde{u}_t;$$

- **Recursive node embeddings**:

  $$V_t^k(v) = \max_{v' \in N_t(v)} \tanh \left(W^{\mathrm{mp}} \left[V_t^{k-1}(v'), R(e_{v \to v'}) \right] \right),$$

  k is the depth-k node embedding at turn t and $N_t(v)$ is the set of nodes adjacent to v. Messages from all neighbors are aggregated by max.
- Finally $V_t(v) = \left[V_t^0(v), \ldots, V_t^K(v) \right],$

Learning Dialogue Agents with KB

Utterance Embedding and Generation

Embedding:

$$h_{t,j} = \mathrm{LSTM}_{\mathrm{enc}}(h_{t,j-1}, A_t(x_{t,j})),$$

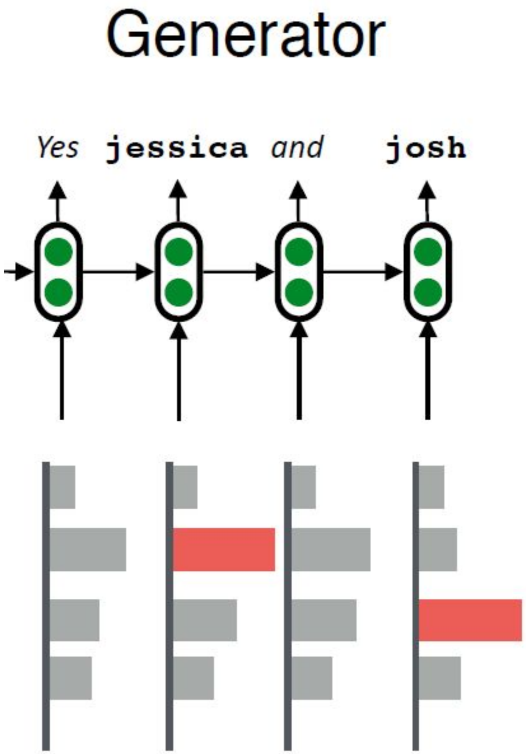
where if y is an entity $A_t(y) = [E_{\mathrm{type}(y)}, V_t(v)]$
otherwise is the word it self

Generation:

Decoder is another LSTM. The initial state is the final state of the step above.

$$h_{t,j} = \mathrm{LSTM}_{\mathrm{dec}}(h_{t,j-1}, [A_t(x_{t,j}), c_{t,j}]),$$
$$c_{t,j} = \sum_{v \in G_t} \alpha_{t,j,v} V_t(v),$$

$c_{t,j}$ is an attention mechanism



Learning Dialogue Agents with KB

Results

Friends of A

ID	Name	Company	Time	Location
1	Kathy	TRT Holdings	afternoon	indoor
2	Jason	Dollar General	afternoon	indoor
3	Johnny	TRT Holdings	afternoon	outdoor
4	Frank	SFN Group	afternoon	indoor
5	Catherine	Dollar General	afternoon	indoor
6	Catherine	Weis Markets	afternoon	indoor
7	Kathleen	TRT Holdings	morning	indoor
8	Lori	TRT Holdings	afternoon	indoor
9	Frank	L&L Hawaiian Barbecue	afternoon	outdoor

Friends of B

ID	Name	Company	Time	Location
1	Justin	New Era Tickets	morning	indoor
2	Kathleen	TRT Holdings	morning	indoor
3	Gloria	L&L Hawaiian Barbecue	morning	indoor
4	Kathleen	Advance Auto Parts	morning	outdoor
5	Justin	Arctic Cat	morning	indoor
6	Anna	Dollar General	morning	indoor
7	Steven	SFN Group	morning	indoor
8	Wayne	R.J. Corman Railroad Group	morning	indoor
9	Alexander	R.J. Corman Railroad Group	morning	indoor

A: Human B: Human

A: Hi
B: hey || i have one outdoor
A: I have 4 TRT Holdings || I have 2 outdoor one Johnny, other Frank
B: i only have one TRT Holdings - Kathleen
A: SELECT 7 B: SELECT 2

A: DynoNet B: Human

A: **hi**
B: Hello || all my friends prefer morning
A: **1 of my morning likes the outdoors**
B: and all like indoor except for one
A: **do they work for trt holdings?**
B: Kathleen?
A: **SELECT 7** B: SELECT 2

A: StanoNet B: Human

A: **Hello**
B: hi
A: **Hello || I have one morning person.**
B: all of my friends like mornings
A: **My friend prefers afternoon works at trt holdings.**
B: what is their name?
A: **Likes indoors.**
B: what is your fiend who likes morning name?
A: **They work for trt holdings.**
B: SELECT 2 A: **SELECT 7**

A: Human B: Rule

B: **hiya**
A: hEY
B: **I have 1 indoors and kathleen.**
A: Most of mine are indoors.
B: **SELECT 1**
A: I have one morning and rest afternoon.
B: **Do you have any friend working at l hawaiian?**
A: I don't know Justin
B: **I have 1 alexander.**
...

Conclusions

All the presented works aim at language generation conditioned to knowledge.

Despite their differences, language generation is always delegated to a sequence-to-sequence approach.

However, knowledge is still considered as an external resource. Even though in [4] knowledge is dynamically updated during the dialogue, its lifetime is only the conversation.

References

[1] Chen, Hongshen, et al. "A Survey on Dialogue Systems: Recent Advances and New Frontiers." *ACM SIGKDD Explorations Newsletter* 19.2 (2017): 25-35.

[2] Vougiouklis, Pavlos, Jonathon Hare, and Elena Simperl. "A neural network approach for knowledge-driven response generation." *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. 2016.

[3] Vougiouklis, Pavlos, et al. "Neural Wikipedian: Generating Textual Summaries from Knowledge Base Triples." *arXiv preprint arXiv:1711.00155* (2017).

[4] He, He, et al. "Learning Symmetric Collaborative Dialogue Agents with Dynamic Knowledge Graph Embeddings." *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vol. 1. 2017.