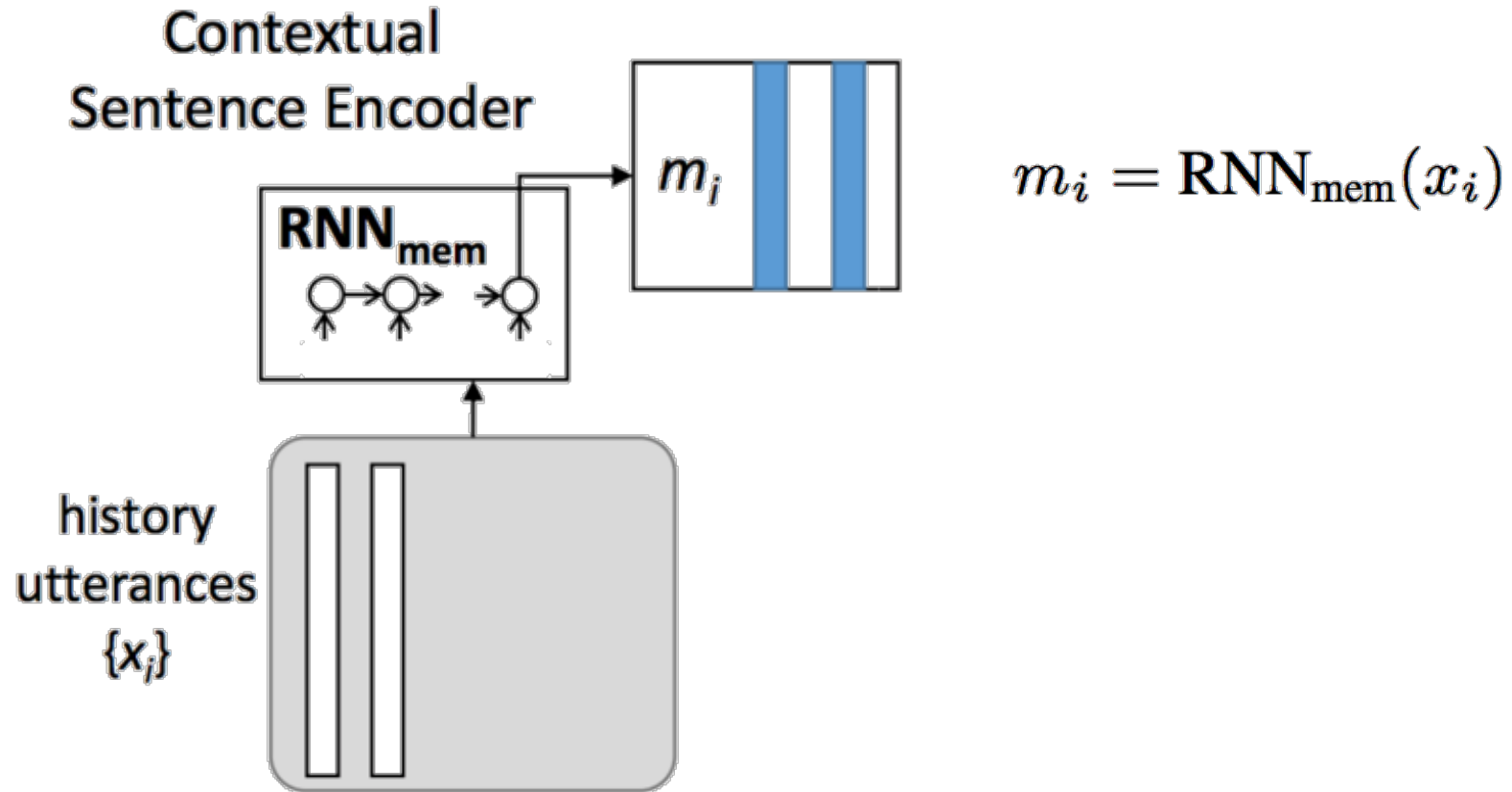# Utilizing context in our NLU

We need context to handle multi-turn dialogs

- User: **Give me directions from Los Angeles**
  - Intent classifier: nav.directions
  - Slot tagger: @FROM{Los Angeles}
  - Dialog manager: *required slot is missing, where to?*
- Agent (assistant): **Where do you want to go?**
- User: **San Francisco**
  - Intent classifier: nav.directions
  - Slot tagger: @TO{San Francisco}
  - Dialog manager: *okay, here's the route*
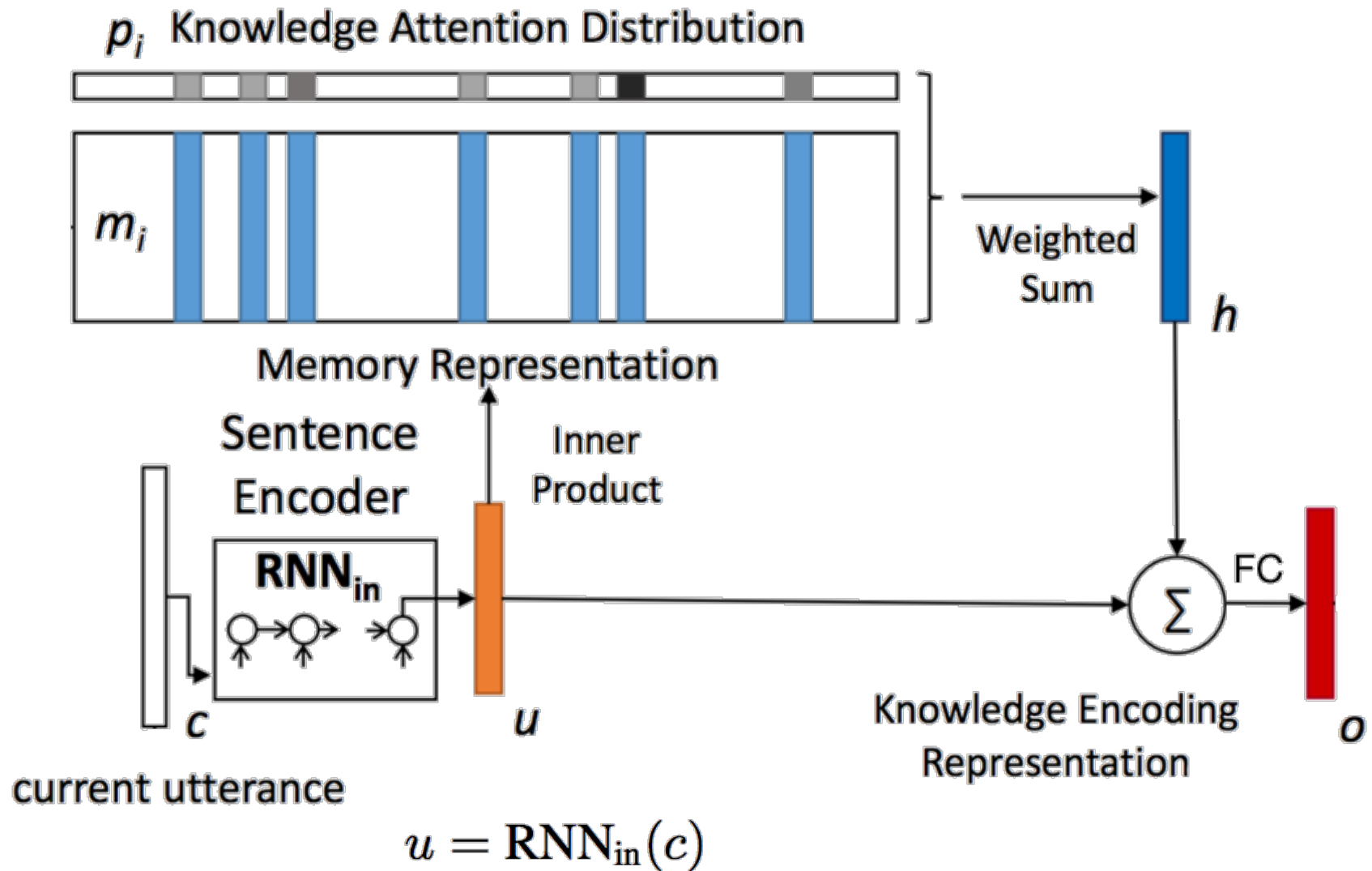- Agent (assistant): **Here's the route**

We need context here

Google Maps

Let's store all previous utterances in "memory"

Contextual Sentence Encoder

RNN_{mem}

m_i

history utterances $\{x_i\}$

$$m_i = \text{RNN}_{\text{mem}}(x_i)$$

https://www.microsoft.com/en-us/research/wp-content/uploads/2016/06/IS16_ContextualSLU.pdf

What knowledge is relevant to new utterance?
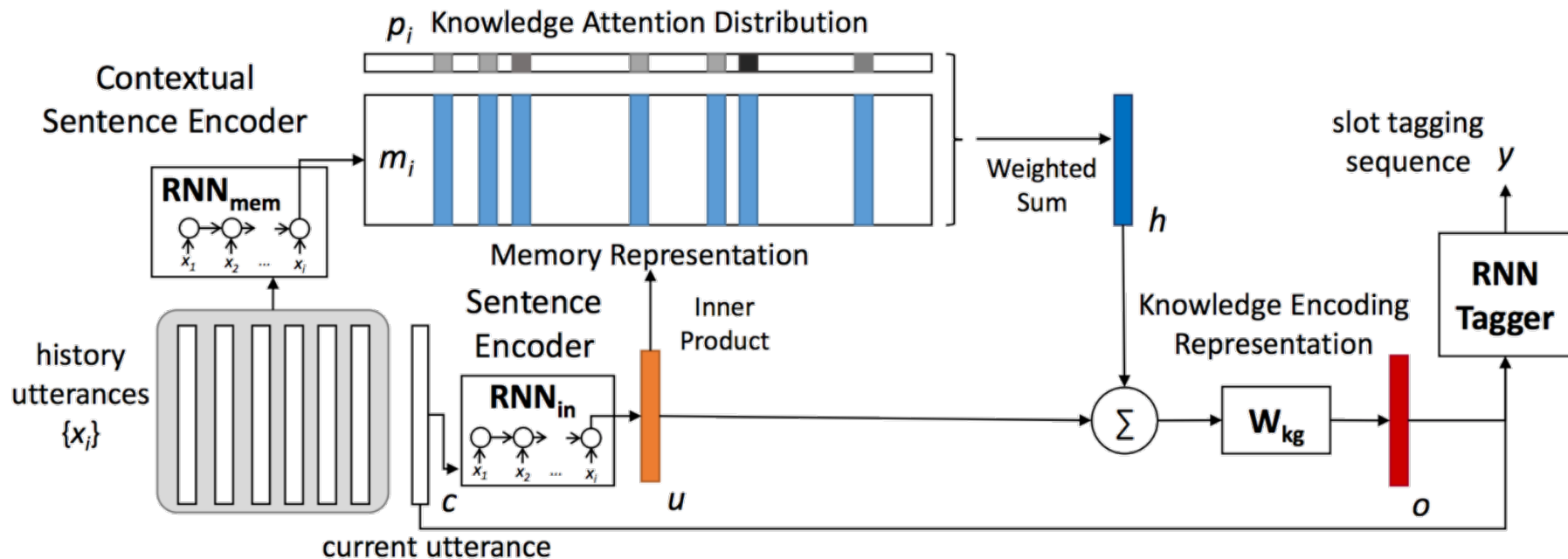

$$u = \mathrm{RNN}_{in}(c)$$

Tagging current utterance with knowledge

- We add knowledge representation in final RNN tagger:
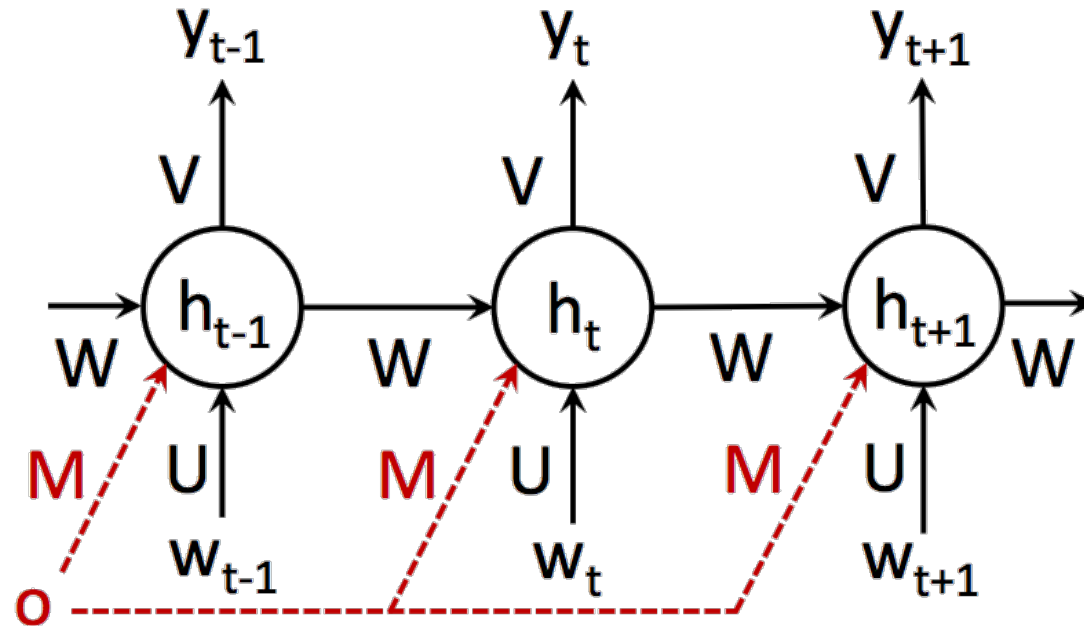
How to track context (with memory networks)

Chen et al, 2016

- We encode previous utterances to store them in **"memory"** as dense vectors
- We use **attention** mechanism to retrieve relevant prior knowledge about the conversation

https://www.microsoft.com/en-us/research/wp-content/uploads/2016/06/IS16_ContextualSLU.pdf

How to track context (with memory networks)

- Evaluation results for slot tagger:
  - Multi-turn dataset
  - F1-measure

Model	First turn	Other turns	Overall
RNN tagger wo context	55.8	45.7	47.4
Memory Network	73.2	65.7	67.1

Summary

- You can make your NLU context-aware with memory networks
- In the next video we'll take a look at lexicon utilization in our NLU (e.g. a list of music artists)