# Reasoning about Entailment with Neural Attention

ROCKTÄSCHEL, GREFENSTETTE, HERMANN, KOČISKÝ, BLUNSOM

ICLR 2016

Presented by Gerui Wang

# Task

Recognizing textual entailment (RTE) is the task of determining whether two natural language sentences are

1. contradicting each other (CONTRADICTION)
2. not related (NEUTRAL)
3. the first sentence entails the second sentence (ENTAILMENT)

Notation: We call the first sentence *premise* and second sentence *hypothesis*.

# Example

Premise: A wedding party taking pictures.

ENTAILS

Hypothesis: Someone got married.

# Example

Premise: A girl is waring a **blue** jacket.

CONTRADICTS

Hypothesis: A young girl wearing a **pink** coat.

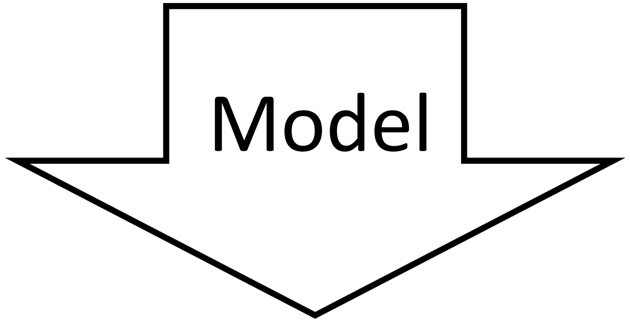
# A Classification Problem

Input: two natural language sentences (Premise, Hypothesis)

Output: one of 3 classes {ENTAILMENT, NEUTRAL, or CONTRADICTION}

# A Classification Problem

Input: two natural language sentences (Premise, Hypothesis)




Output: one of 3 classes {ENTAILMENT, NEUTRAL, or CONTRADICTION}

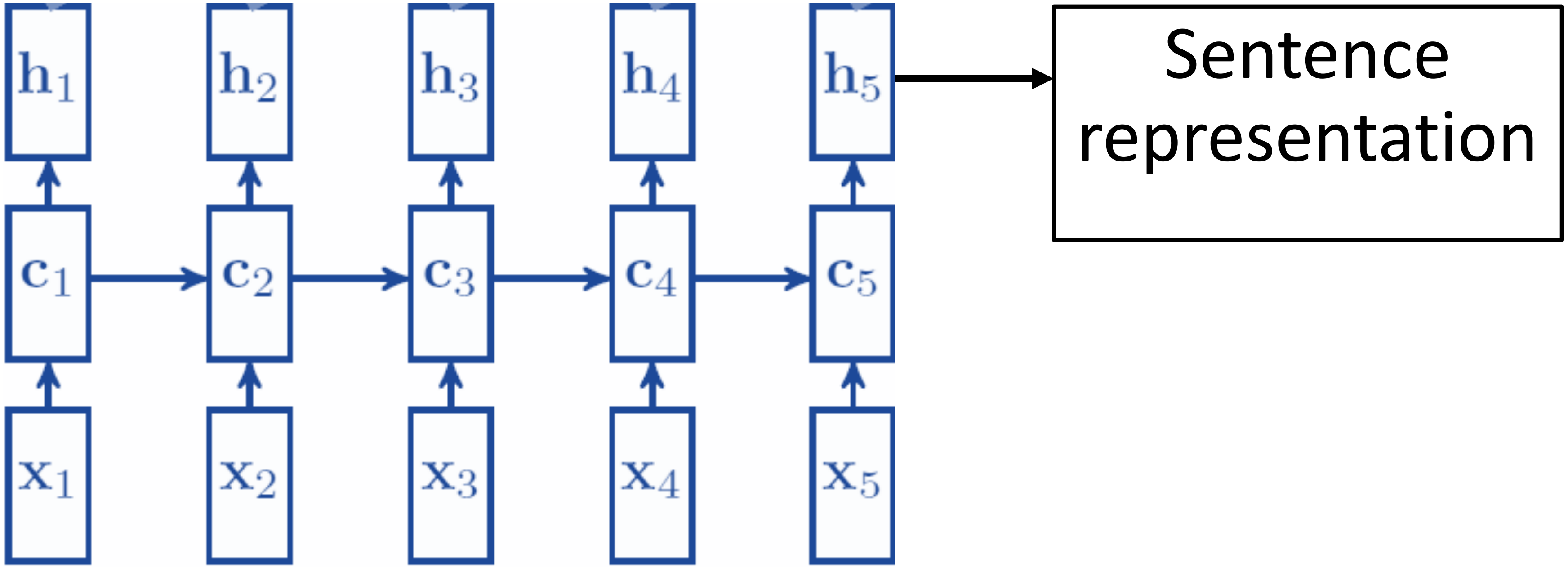
# Previous Models

State-of-the-art: Lexicalized classifier that heavily relies on hand-crafted features, various external resources, and other subcomponents such as negation detection.

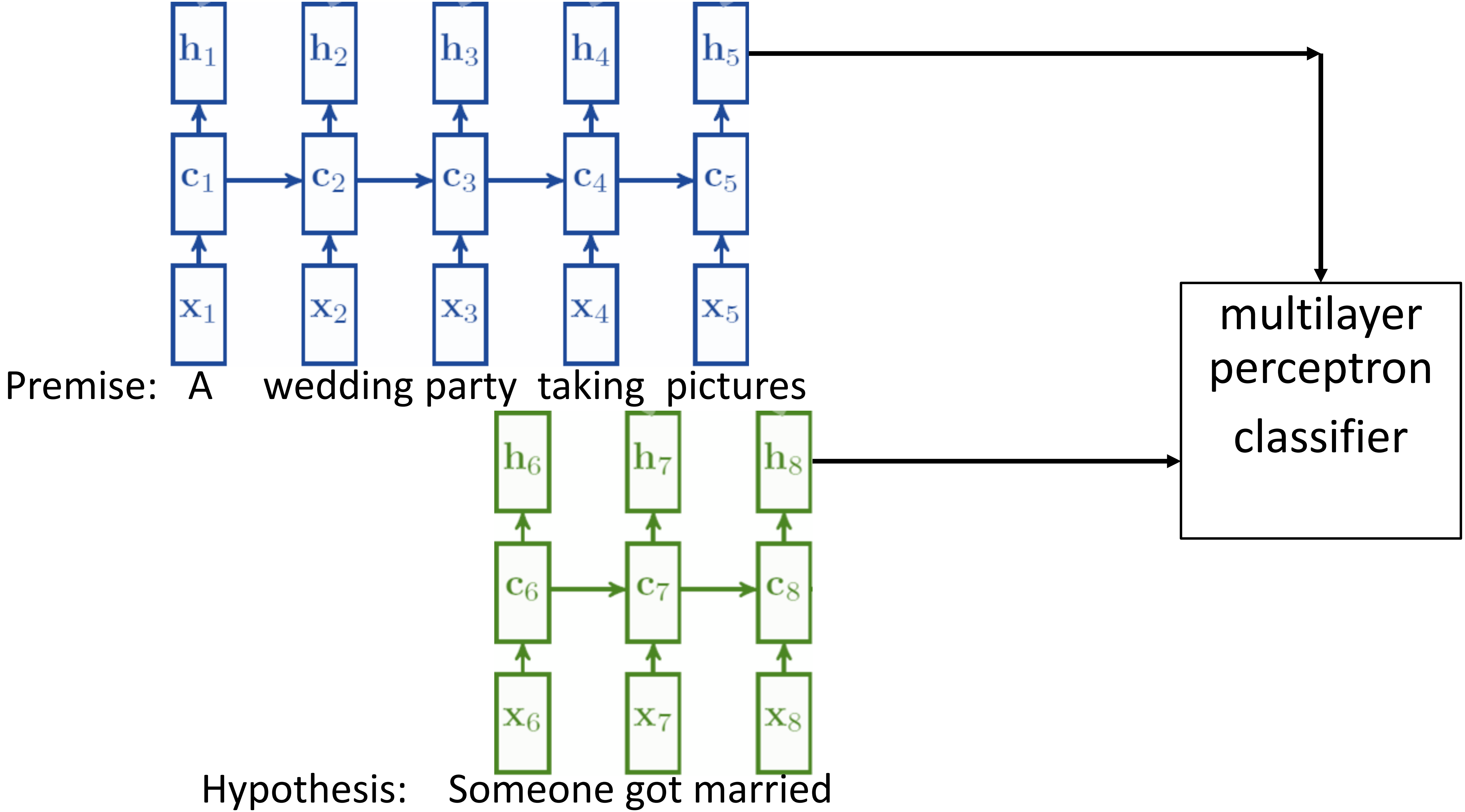
Neural Model:

- Pros: end-to-end differentiable and trainable, avoids assumptions and external resources about the underlying language.
- Cons: Previous LSTM model (Bowman 2015) didn't outperform state-of-the-art.

# Previous LSTM model (Bowman 2015)




Premise: A wedding party taking pictures


h1
h2
h3
h4
h5
c1
c2
c3
c4
c5
x1
x2
x3
x4
x5
Premise: A wedding party taking pictures
h6
h7
h8
c6
c7
c8
x6
x7
x8
Hypothesis: Someone got married
multilayer perceptron classifier

# Observation

Fail to capture the asymmetry of premise and hypothesis.

Observation:

People may read the hypothesis in a different way conditioned on the semantic of the premise.

- Premise: A girl is waring a blue jacket.
- Hypothesis: A young girl wearing a pink coat plays with a yellow toy gold club.

# Observation

Fail to capture the asymmetry of premise and hypothesis.

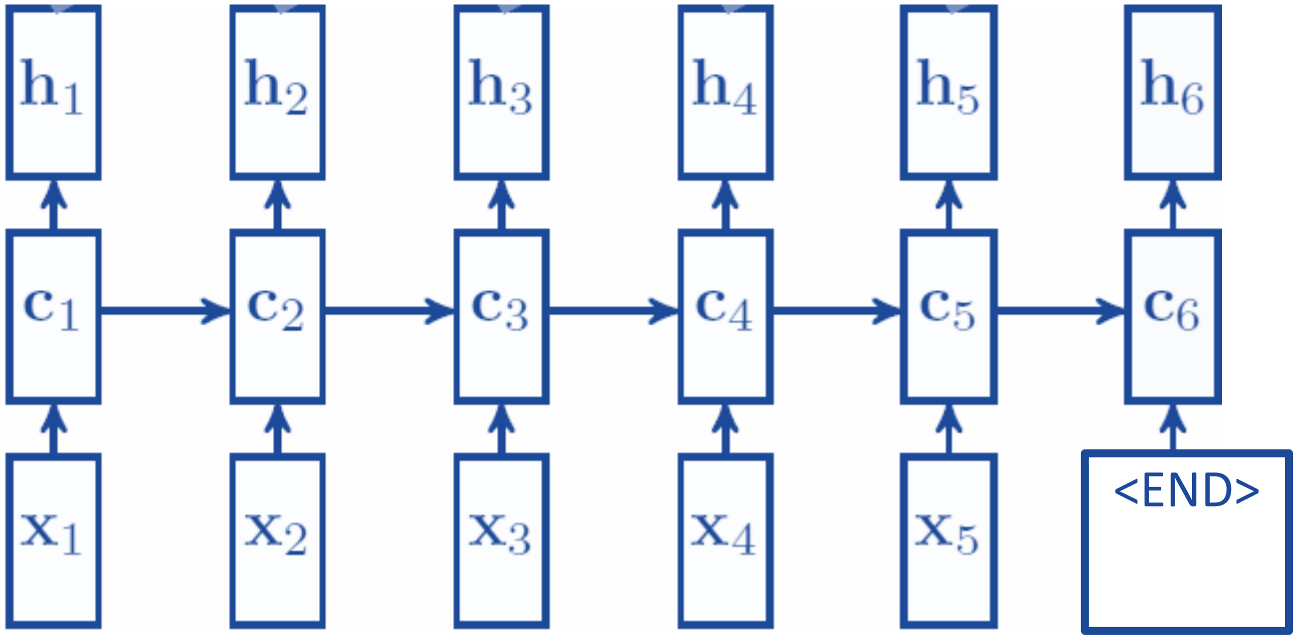
Observation:

People may read the hypothesis in a different way conditioned on the semantic of the premise.

- Premise: A girl is waring a blue jacket.
- Hypothesis: A young girl wearing a pink coat plays with a yellow toy gold club.

# Conditional Encoding

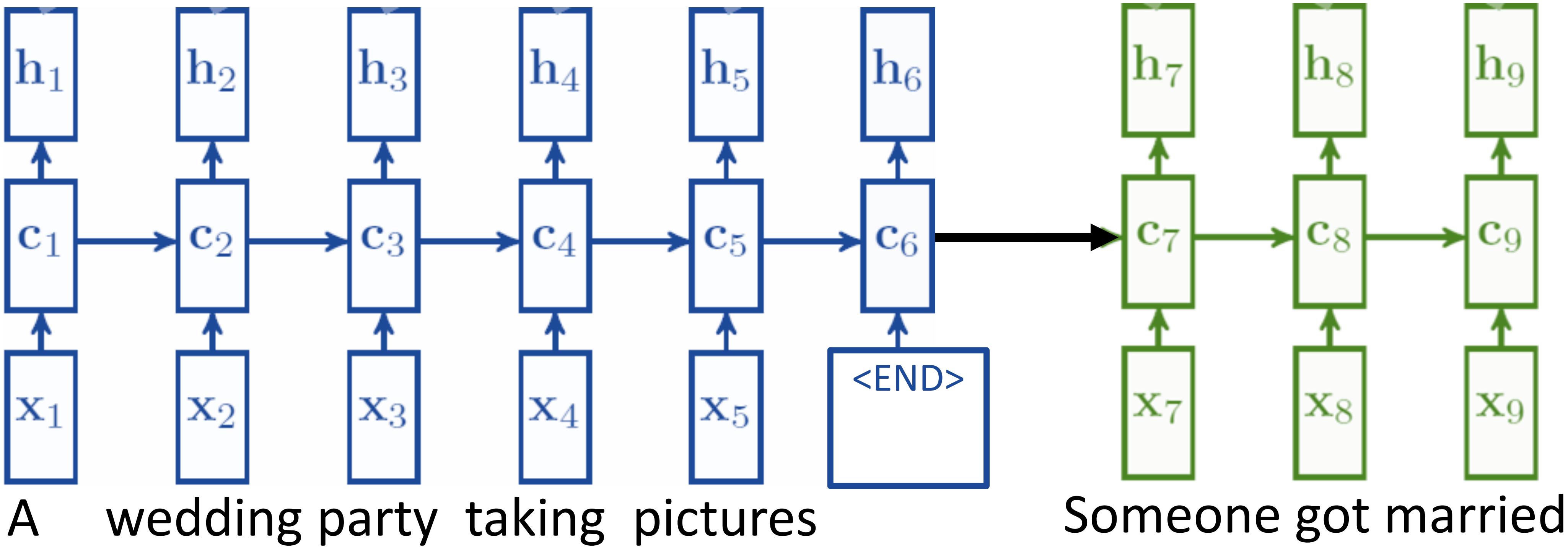
Use LSTM-1 to encode premise until a special token of <END> is encountered.




A wedding party taking pictures

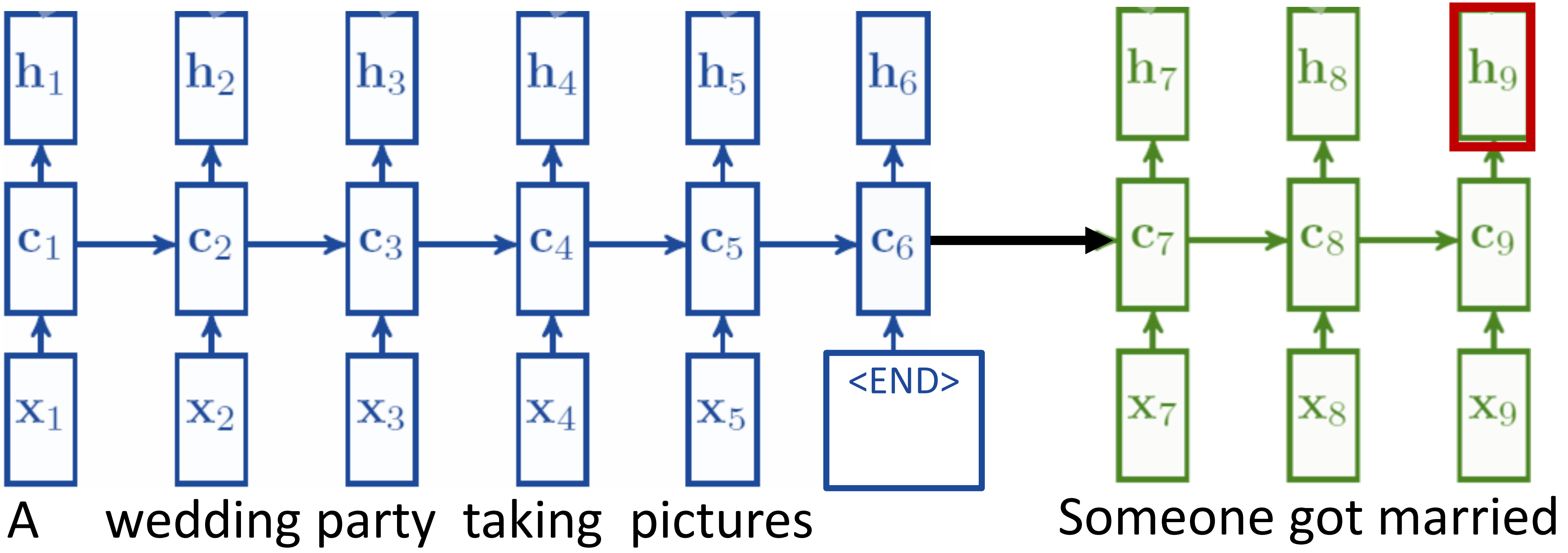
# Conditional Encoding (cont.)

Initialize LSTM-2 with the last cell vector of LSTM-1. Then use LSTM-2 to encode hypothesis.



# Conditional Encoding (cont.)

For classification, use a softmax layer over the output of a non-linear projection of the last output vector (h9 in the example).



# Observation 2

- People may read the hypothesis in a more focused way conditioned on the premise.
- Premise: A **wedding** party taking pictures.
- Hypothesis: Someone got **married**.
- (If we notice the word *wedding* in premise, we will focus on related words in hypothesis, i.e., *married*.)
- It’s natural to use attention mechanism over LSTM.

# Attention

Let $[h_1, h_2, \dots, h_L]$ be the output vectors of the premise, where every $h_i$ is a vector.

For the last output vector $h_N$ of the hypothesis, define attention $\alpha$ as follows,

$$m_i = \tanh(W^y h_i + W^h h_N)$$

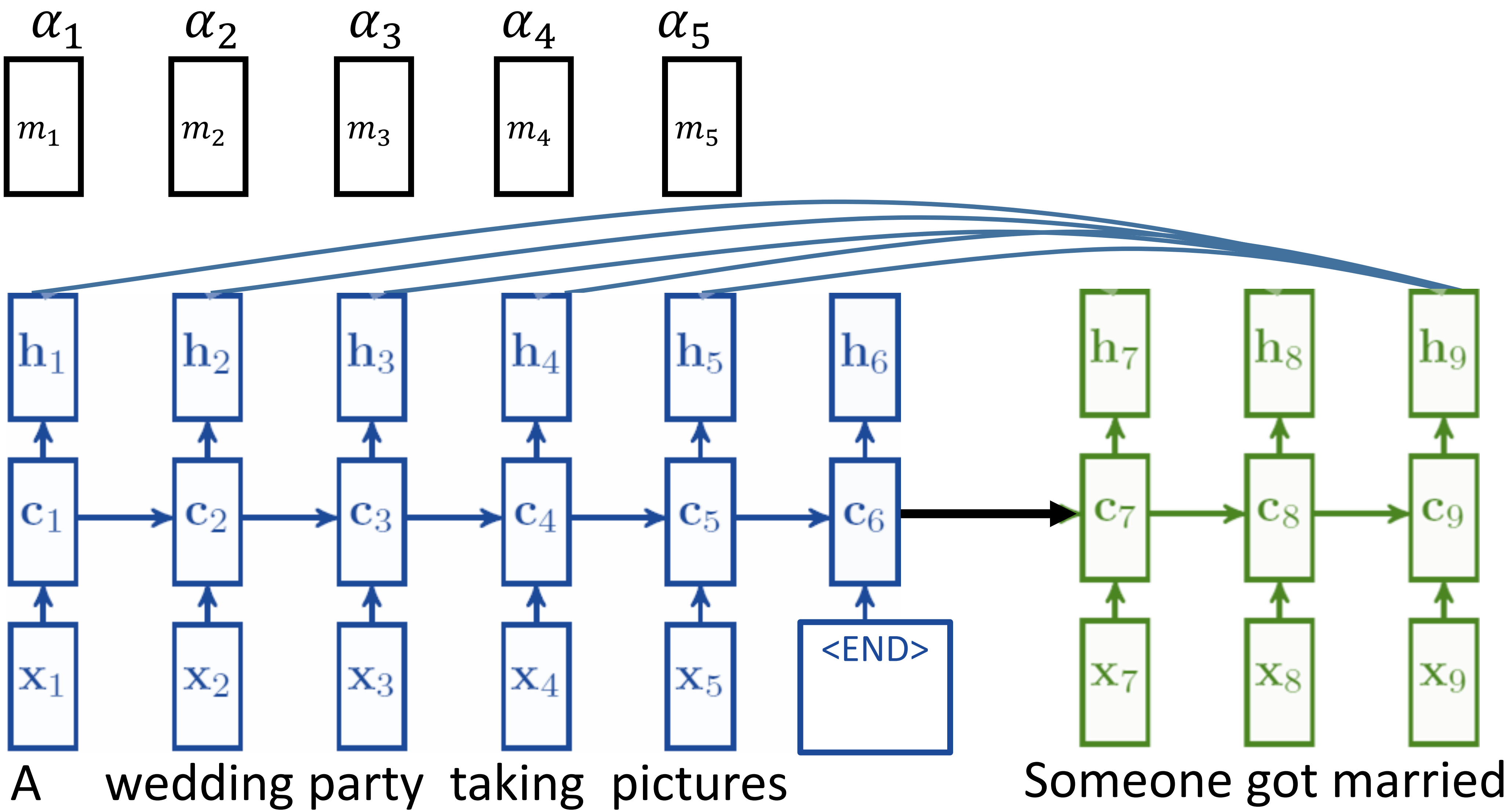
this result is a vector of same dimensionality as $h_i$ and $h_N$.

Then project $m_i$ to a scalar and put a softmax layer over them to compute attention $\alpha$.

$$\alpha_i = \text{softmax}(w^T m_i)$$

# Attention

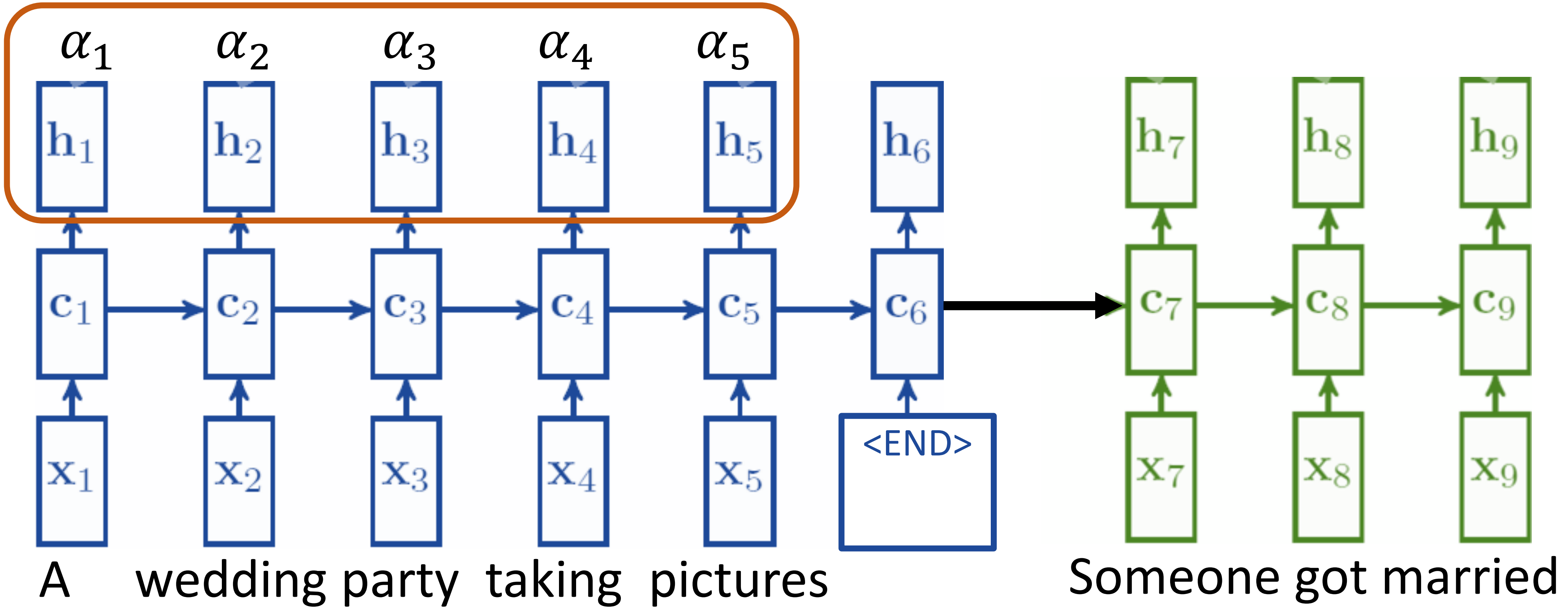
$$m_i = \tanh(W^y h_i + W^h h_N)$$
$$\alpha_i = \text{softmax}(w^T m_i)$$



# Attention (cont.)

Then compute the attention-weighted representation of premise

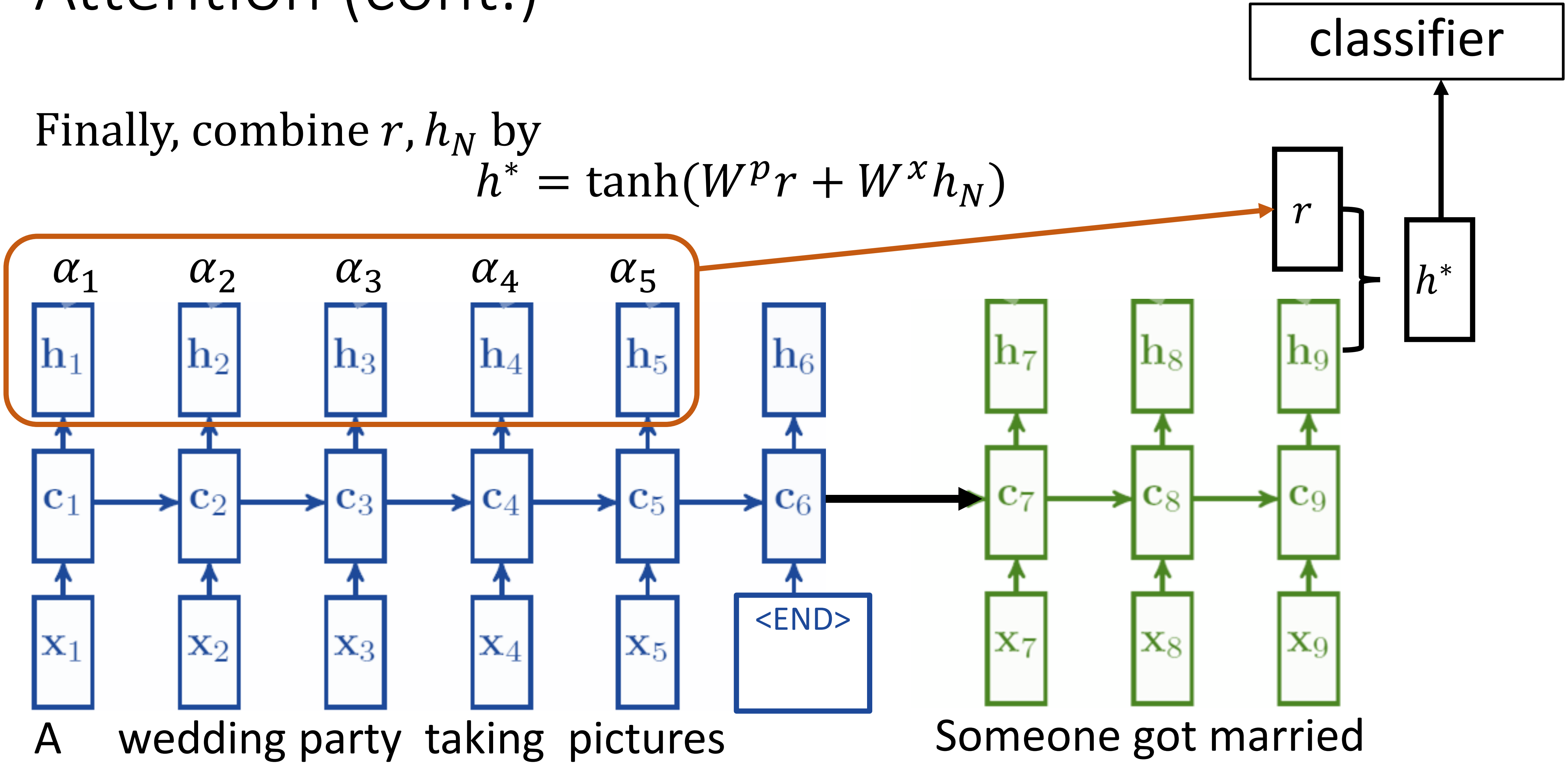
$$r := \sum_{i=1}^{L} \alpha_i \; h_i$$



# Attention (cont.)

Finally, combine $r, h_N$ by

$$h^* = \tanh(W^p r + W^x h_N)$$



# Word-by-word attention

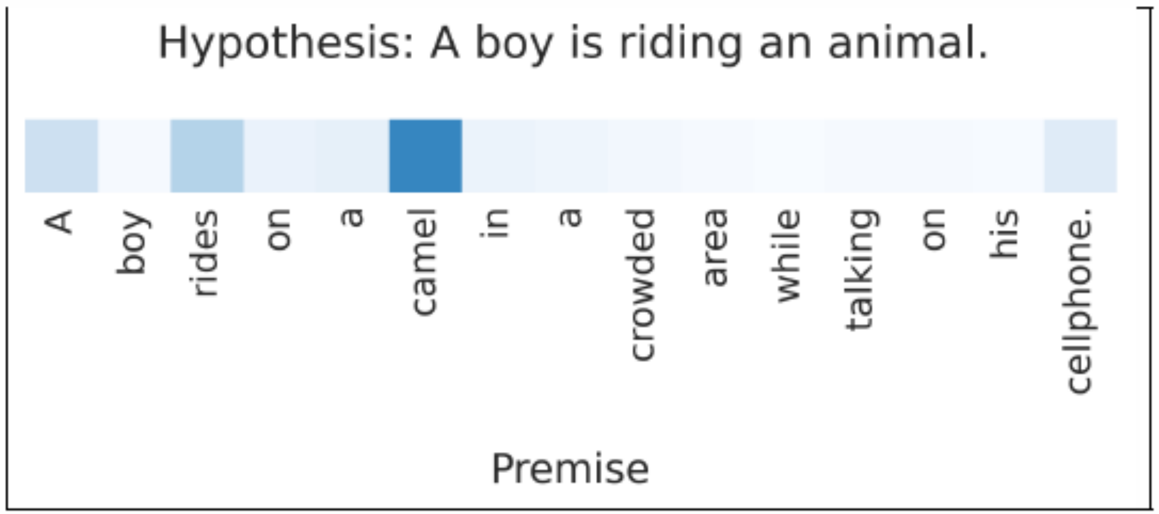
Compute attention for every premise word and every hypothesis word.

Very similar to previous slides, only with more computation.

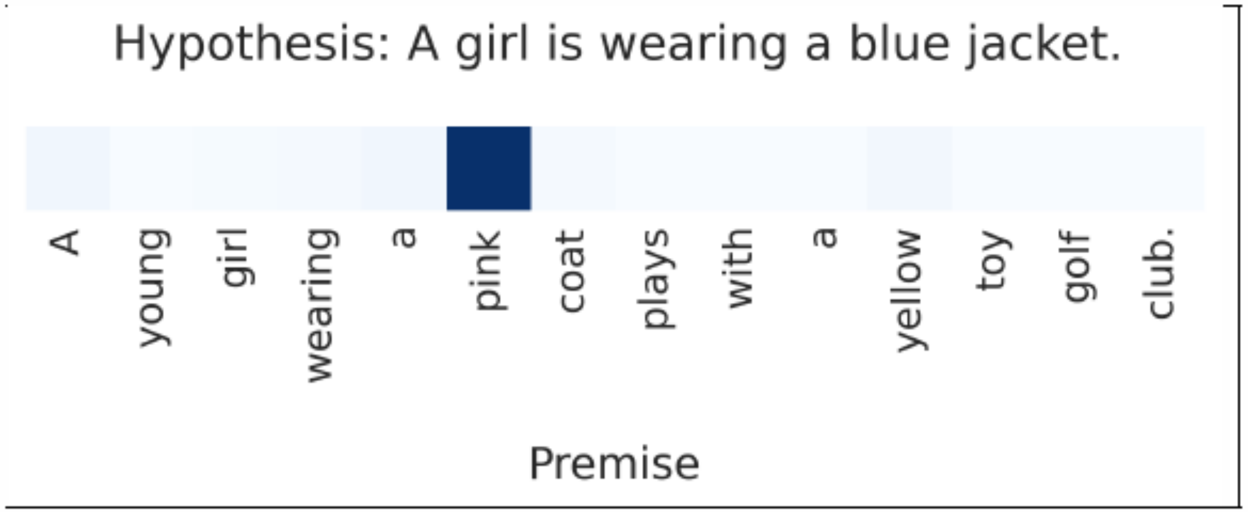
# Results of SNLI corpus

| Model | #Parameters | Dev Acc | Test Acc |
|---|---|---|---|
| Lexicalized classifier | - | - | 78.2 |
| LSTM (Bowman 2015) | 10M | - | 77.6 |
| Conditional encoding | 3.9M | 82.1 | 80.9 |
| Attention | 3.9M | 83.2 | 82.3 |
| Word-by-word attention | 3.9M | **83.7** | **83.5** |

# Results of attention (examples)




(a)




(b)

# Conclusion

For the problem of RTE, the authors use conditional encoding, attention, and word-by-word attention to improve the performance of the neural model.

For the first time, a neural model outperforms a feature-based system for RTE.

Since neural model is end-to-end differentiable and trainable, authors consider to use it to other sequential data other than natural language.

# Thanks!

# Experiment

Stanford Natural Language Inference corpus (SNLI).

Use cross-entropy loss.

ADAM optimizer with recommended coefficient.

Grid search on initial learning rate, dropout rate, and l2 regularization strength.

# Experiment Details

- Use word2vec vectors and add a linear layer to project them into proper dimensionality.
- Fix word2vec vectors during training. Out-of-vocabulary word vectors during training are initialized randomly and optimized during training.