Adaptive Knowledge Sharing in Multi-Task Learning: Improving Low-Resource Neural Machine Translation

Poorya ZareMoodi, Wray Buntine, Gholamreza (Reza) Haffari

Monash University

Slides:

Roadmap

- **Introduction & background**
- Adaptive knowledge sharing in Multi-Task Learning
- Experiments & analysis
- Conclusion

Improving NMT in low-Resource scenarios

- NMT is notorious!
- Bilingually low-resource scenario: large amounts of bilingual training data is not available
- IDEA: Use existing resources from other tasks and train one model for all tasks using multi-task learning
- This effectively injects inductive biases to help improving the generalisation of NMT
- Auxiliary tasks: Semantic Parsing, Syntactic Parsing, Named Entity Recognition

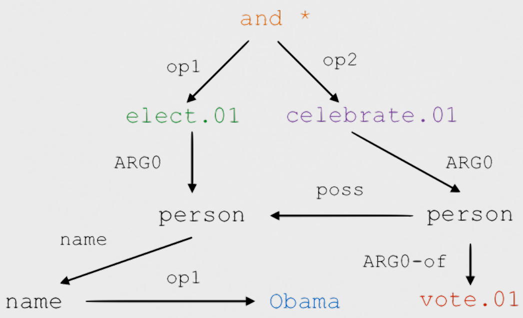
Encoders-Decoders for Individual Tasks

Machine Translation

I went home → Encoder → Decoder → من به خانه رفتم

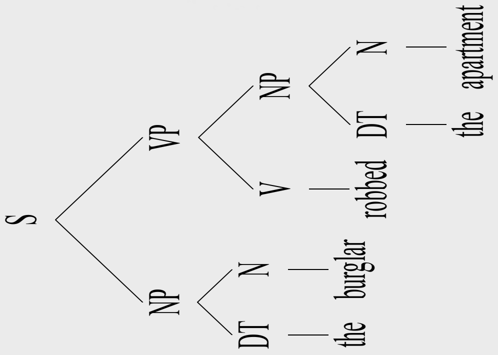
Semantic Parsing

Obama was elected and his voter celebrated → Encoder → Decoder



Syntactic Parsing

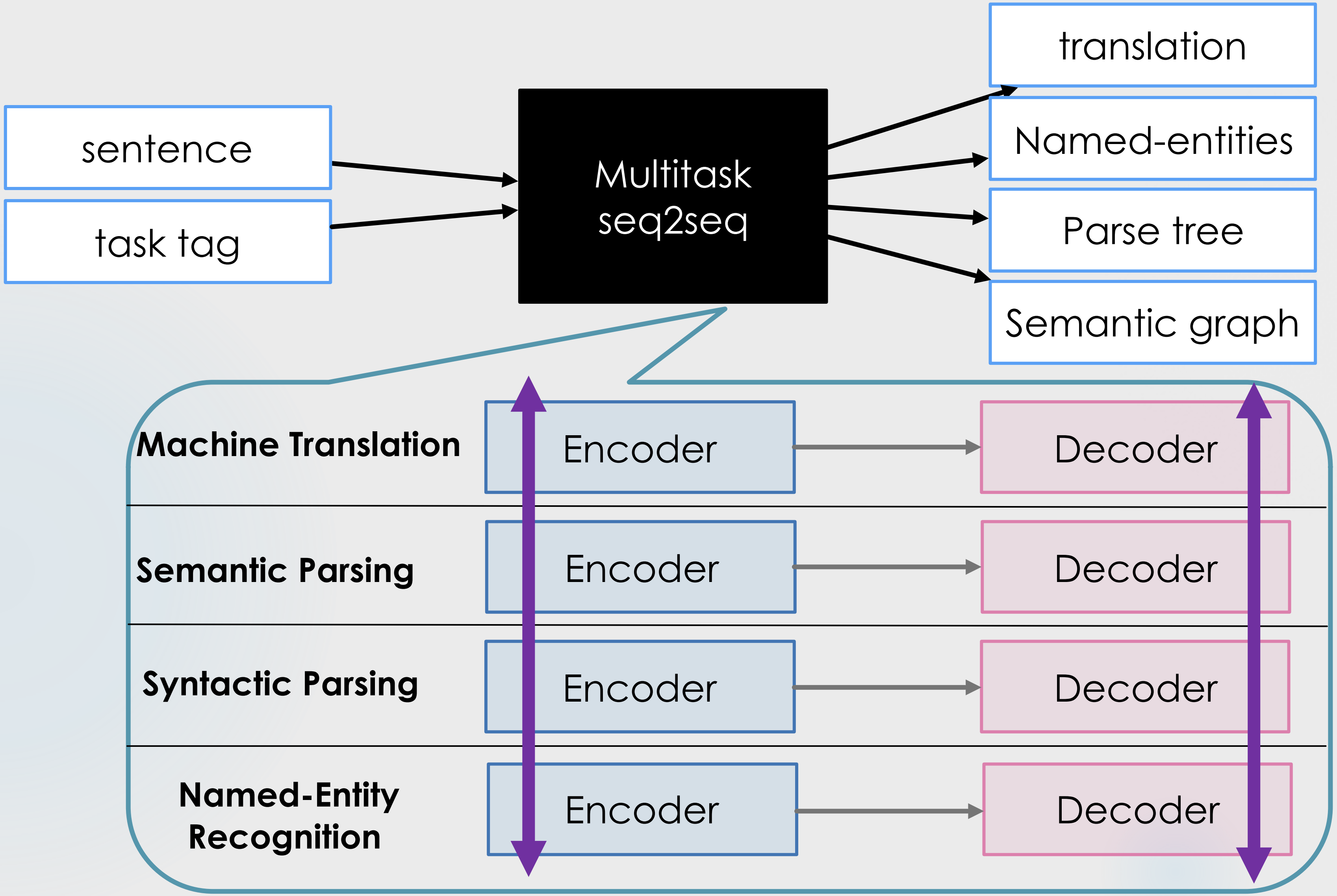
The burglar robbed the apartment → Encoder → Decoder



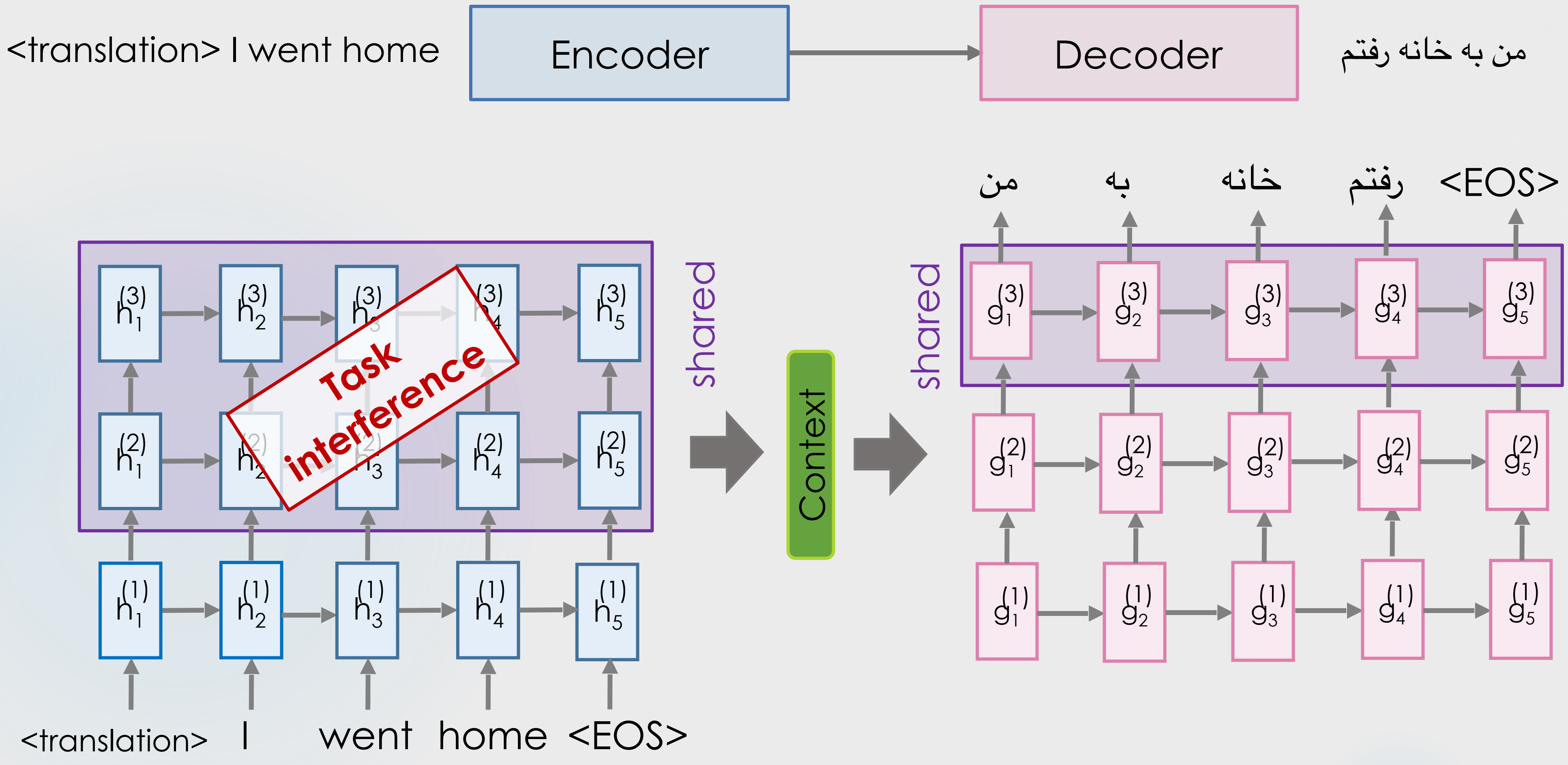
Named-Entity Recognition

Jim bought 300 shares of Acme Corp. in 2006 → Encoder → Decoder → B-PER 0 0 0 0 B-ORG I-ORG 0 B-MISC

Sharing Scenario



Partial Parameter Sharing



Zaremoodi & Haffari, NAACL, 2018

Roadmap

- Introduction & Background
- **Adaptive knowledge sharing in Multi-Task Learning**
- Experiments & analysis
- Conclusion

Adaptive Knowledge Sharing in MTL

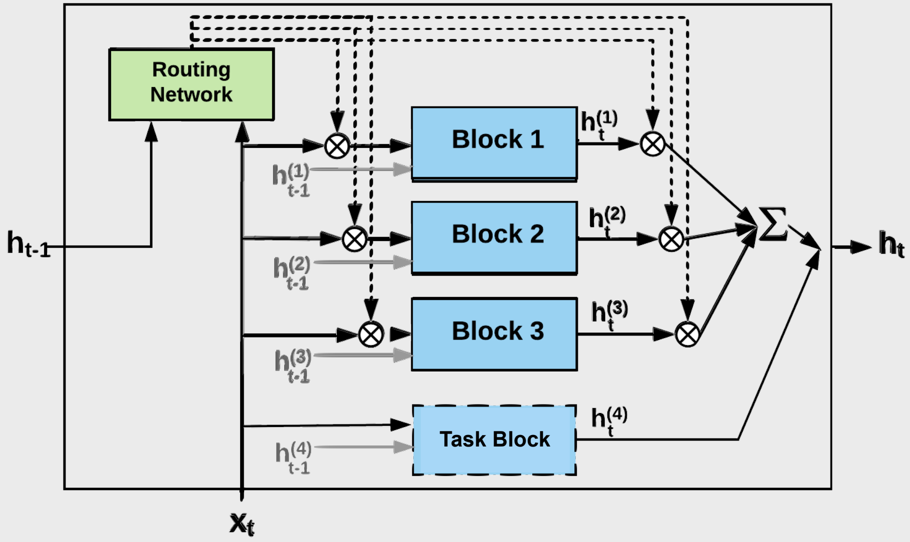
- Sharing the parameters of the recurrent units among all tasks
  - Task interference
  - Inability to leverage commonalities among *subsets* of tasks

sharing the *knowledge* for controlling the information flow in the hidden states

- IDEA
  - Multiple experts in handling different kinds of information
  - *Adaptively* share experts among the tasks

Adaptive Knowledge Sharing in MTL

- IDEA
  - Multiple experts in handling different kinds of information
  - Adaptively share experts among the tasks
  - Extend the recurrent units with multiple blocks
    - each block has its own information flow through the time
    - *Routing mechanism:* to softly direct the input to these blocks



Adaptive Knowledge Sharing

Routing:

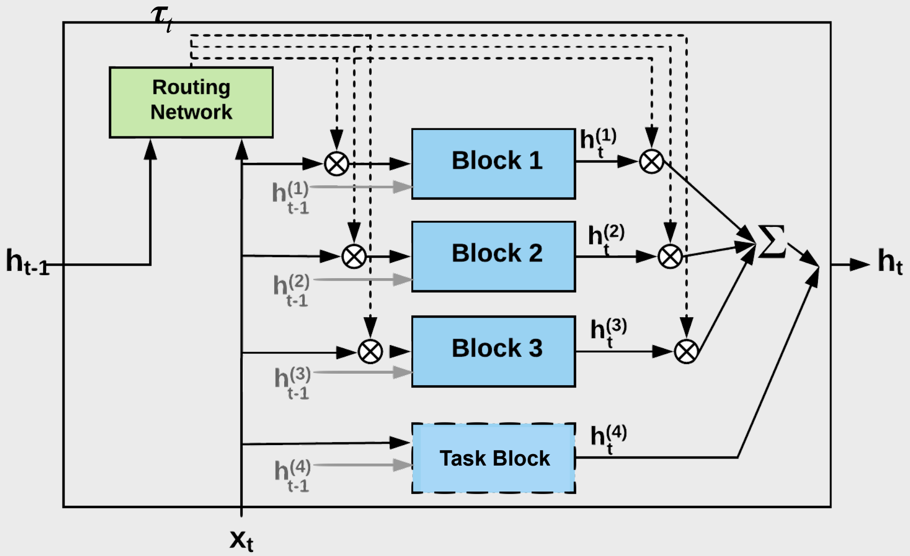
$$\begin{aligned} \boldsymbol{s}_t &= \tanh(\boldsymbol{W}_x \cdot \boldsymbol{x}_t + \boldsymbol{W}_h \cdot \boldsymbol{h}_{t-1} + \boldsymbol{b}_s), \\ \boldsymbol{\tau}_t &= \mathrm{softmax}(\boldsymbol{W}_\tau \cdot \boldsymbol{s}_t + \boldsymbol{b}_\tau), \end{aligned}$$

➡ $\tilde{\boldsymbol{x}}_t^{(i)} = \boldsymbol{\tau}_t[i]\boldsymbol{x}_t$ ➡ $\boldsymbol{h}_t^{(shared)} = \sum_{i=1}^{n} \boldsymbol{\tau}_t[i]\boldsymbol{h}_t^{(i)}$ ➡ $\boldsymbol{h}_t = [\boldsymbol{h}_t^{(shared)}; \boldsymbol{h}_t^{(task)}]$

Blocks:

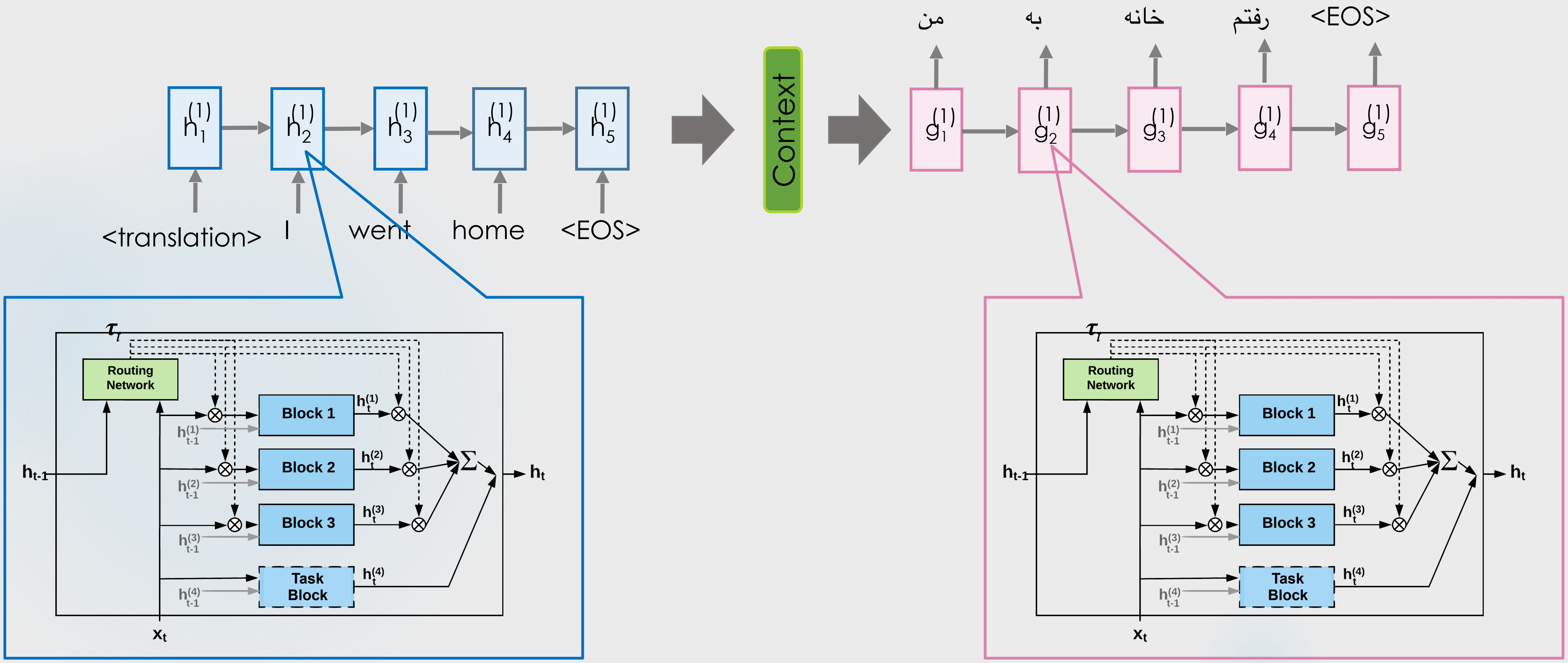
$$\begin{aligned} \boldsymbol{z}_t^{(i)} &= \sigma(\boldsymbol{W}_z^{(i)}\tilde{\boldsymbol{x}}_t^{(i)} + \boldsymbol{U}_z^{(i)}\boldsymbol{h}_{t-1}^{(i)} + \boldsymbol{b}_z^{(i)}), \\ \boldsymbol{r}_t^{(i)} &= \sigma(\boldsymbol{W}_r^{(i)}\tilde{\boldsymbol{x}}_t^{(i)} + \boldsymbol{U}_r^{(i)}\boldsymbol{h}_{t-1}^{(i)} + \boldsymbol{b}_r^{(i)}), \end{aligned}$$

$$\begin{aligned} \tilde{\boldsymbol{h}}_t^{(i)} &= \tanh(\boldsymbol{W}_h^{(i)}\tilde{\boldsymbol{x}}_t^{(i)} + \boldsymbol{U}_h^{(i)}\boldsymbol{h}_{t-1}^{(i)} + \boldsymbol{b}_h^{(i)}), \\ \boldsymbol{h}_t^{(i)} &= \boldsymbol{z}_t^{(i)} \odot \boldsymbol{h}_{t-1}^{(i)} + (1 - \boldsymbol{z}_t^{(i)}) \odot \tilde{\boldsymbol{h}}_t^{(i)}. \end{aligned}$$



Adaptive Knowledge Sharing

We use the proposed recurrent unit inside encoder and decoder.



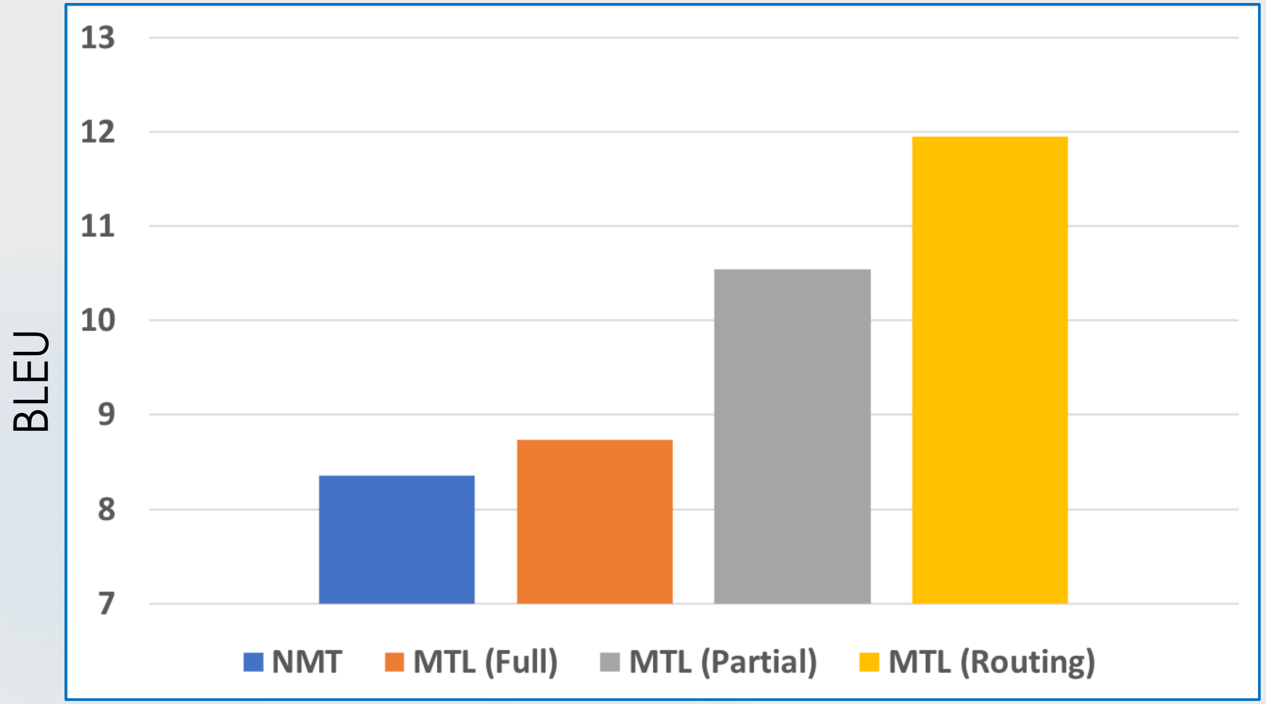
Roadmap

Experiments

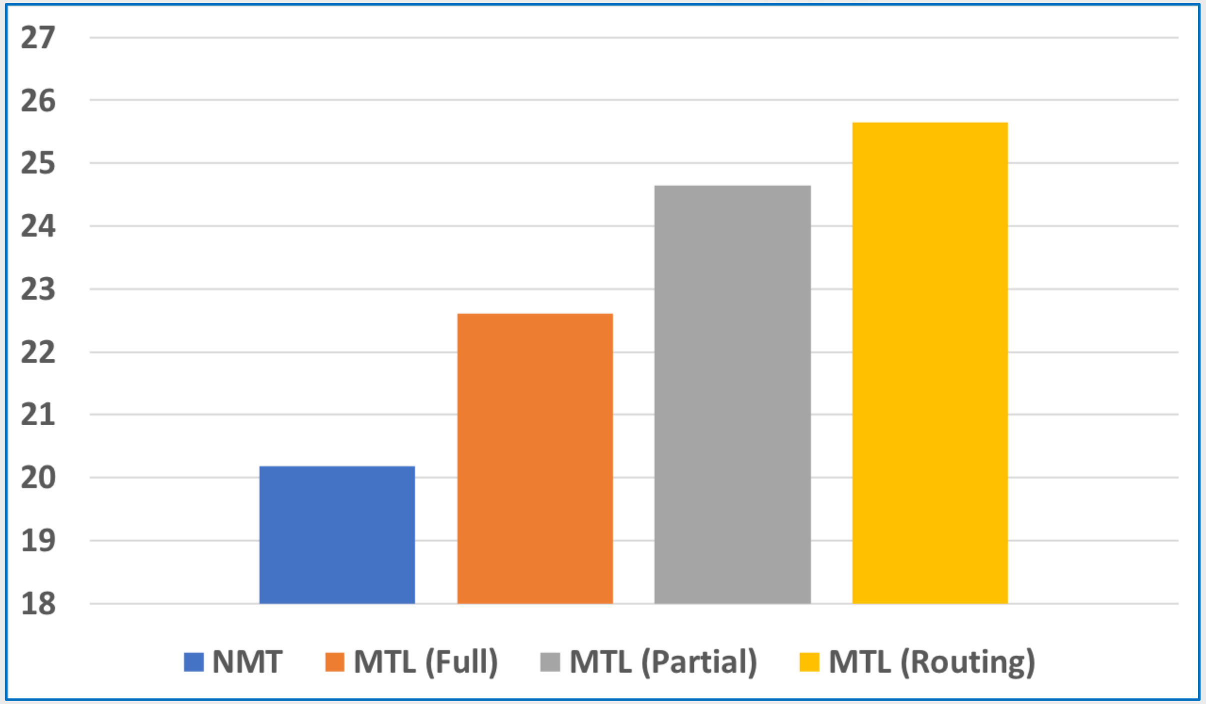
	Train	Dev	Test
En → Fa	98,158	3,000	4,000
En → vi	133,290	1,553	1,268

- Language Pairs: English to Farsi/Vietnamese
- Datasets:
  - English to Farsi: TED corpus & LDC2016E93
  - English to Vietnamese: IWSLT 2015 (TED and TEDX talks)
  - Semantic parsing: AMR corpus(newswire, weblogs, web discussion forums and broadcast conversations)
  - Syntactic parsing: Penn Treebank
  - NER: CONLL NER Corpus (newswire articles from the Reuters Corpus)
- NMT Architecture: GRU for blocks, 400 RNN hidden states and word embedding
- NMT best practice:
  - Optimisation: Adam
  - Byte Pair Encoding (BPE) on both source/target
  - Evaluation metrics: PPL, TER and BLEU

Experiments

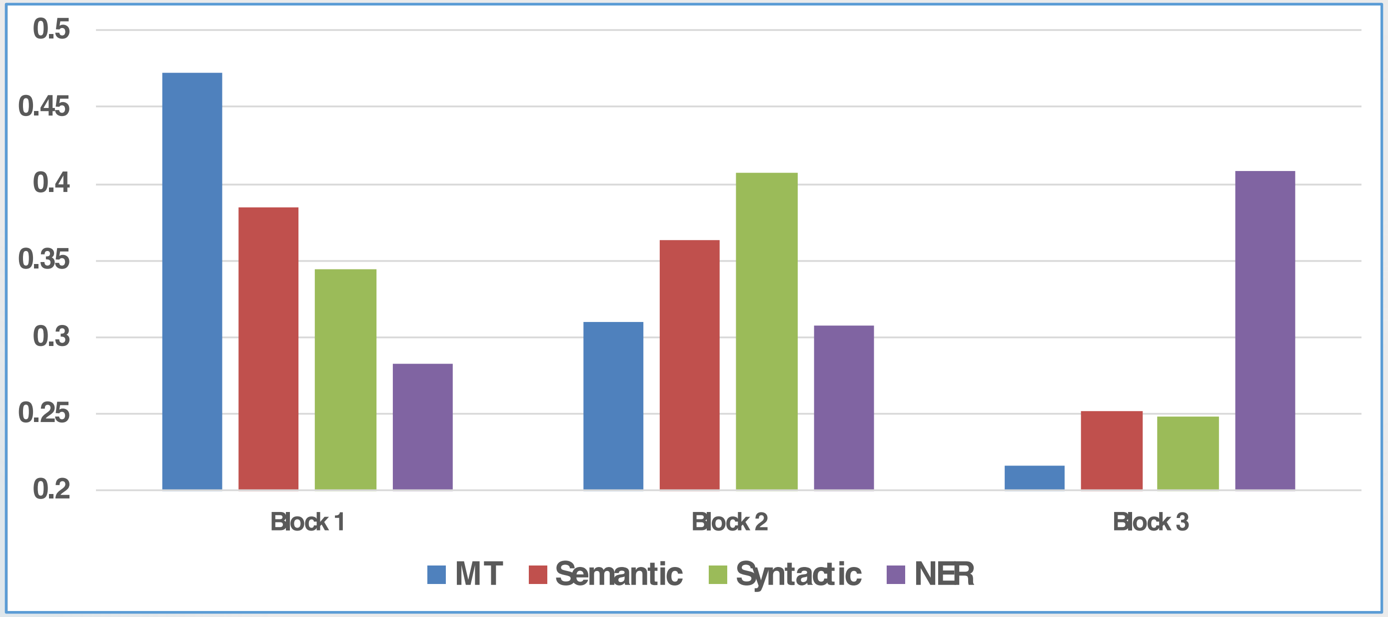


English ➔ Farsi



English ➔ Vietnamese

Experiments (English to Farsi)



- Average block usage.
- Blocks specialisation: Block 1: MT, Semantic Parsing, Block 2: Syntactic/Semantic Parsing, Block 3: NER

Conclusion

- Address the task interference issue in MTL
  - extending the recurrent units with multiple *blocks*
  - with a trainable *routing network*

Questions?

Paper: