# -SOTA-
# Document-level Neural Machine Translation

by Lorenzo Lupo

April 2020

# Plan

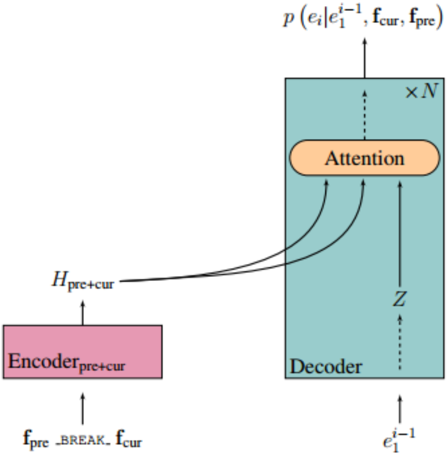
# Plan

# Overview

- gnee

# Concatenation Approaches

Concatenation approaches to DLNMT consist in feeding a standard encoder-decoder architecture with a concatenation of sentences.



# Concatenation Approaches

For instance:

- [Tiedemann and Scherrer, 2017] firstly introduced this approach proposing an RNN-based model that incorporate the preceding sentence by prepending it to the current one, separated by a <CONCAT> token. They propose two methods:
  - **2-TO-2**: the previous and the current sentences are translated together. The translation of the current sentence is then obtained by only retaining the tokens following the concatenation token.
  - **2-TO-1**: only the current sentence is translated.
- [Agrawal et al., 2018, Scherrer et al., 2019] investigated the concatenation approach with the Transformer as base model, extending the number of context sentences both on the source (s:-3,+1) and the target (t:-2) side.

# Separate Encoding Approaches

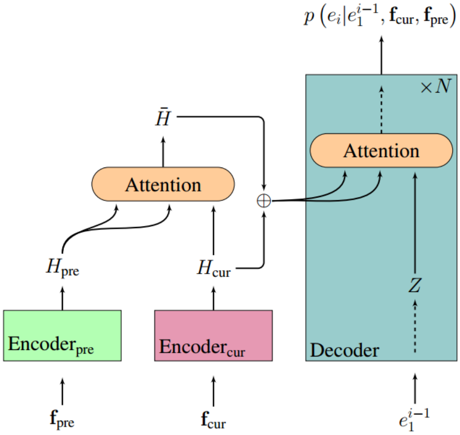
Separate encoding approaches to DLNMT consist in encoder-decoder models that encode the current and context sentences separately. This can be undertaken by:

- **Multiple encoders** working in parallel for the current and previous sentence. E.g. [Wang et al., 2017].
- **Multiple encoders with shared weights**. In this case, the parallel-working encoders not only have the same architecture, but also the same weights. E.g. [Voita et al., 2018].
- **Two-pass approaches**, in which the encoder makes a first sentence-level encoding pass of the source, and a second in which it encodes contextual information too. See Slide 13.
  - Remark: a powerful feature of two-pass approaches is their ability to exploit **future** target-side context.

# Separate Encoding Approaches

Once the encoding of the current and the context sentences has been carried out, they can be integrated in different ways:

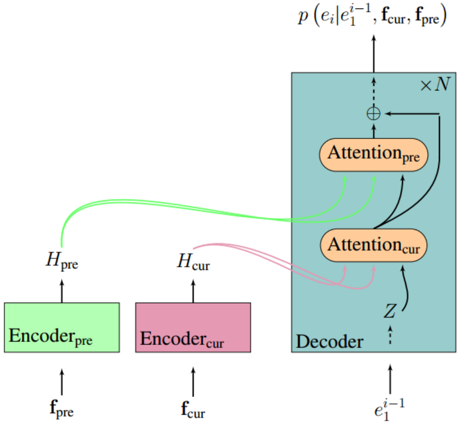
- **Outside** the decoder.
  - $(+)$ symbol represents a gate, a sum or a concatenation.



# Separate Encoding Approaches

Once the encoding of the current and the context sentences has been carried out, they can be integrated in different ways:

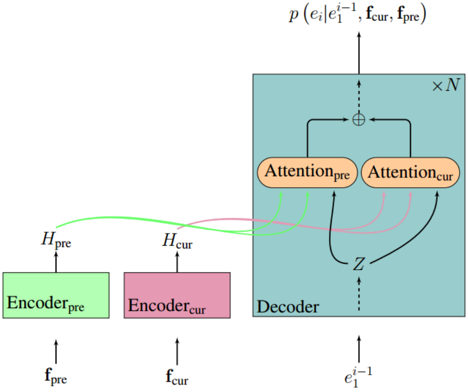
- **Outside** the decoder.
  - $(+)$ symbol represents a gate, a sum or a concatenation.
- **Inside** the decoder, **sequentially**.



# Separate Encoding Approaches

Once the encoding of the current and the context sentences has been carried out, they can be integrated in different ways:

- **Outside** the decoder.
  - $(+)$ symbol represents a gate, a sum or a concatenation.
- **Inside** the decoder, **sequentially**.
- **Inside** the decoder, **in parallel**.



# Separate Encoding Approaches

**Architecture**

The encoder-decoder architectures depicted above can be both RNN-based (until 2017) or Transfomer-based (after 2017), as for any approach to DLNMT. However, often some modifications are applied. For example:

- In the case of RNN-based architectures, integration inside the decoder can be undertaken without attention by simply concatenating context representations to the cell state of the deocdrr's RNN [Wang et al., 2017].
- beside contextual representation of words, the context encoder can also higher level representations such as sentence or document representations. This representations can also be attended by the decoder [Miculicich et al., 2018, Maruf et al., 2019] or added to the word-representations [Tan et al., 2019].
- Parallel integration inside the decoder can also happen within a single multi-head attention that takes as values and queries the concatenations of the current and context sentence representations [Voita et al., 2019b]

# Separate Encoding Approaches

**Including target-side context**

Despite some have considered including past target-side context harmful because of the *error propagation* problem [Zhang et al., 2018], most recent works have showed it to be of utmost importance for making the most out of context. Past works have successfully included target-side context information in different ways:

- Translating past sentences (usually 1) along with the current one, and then discarding them, as in concatenation approaches [Bawden et al., 2018].
- By making the decoder attend the target-side hidden representations or embeddings of previously decoded sentences [Miculicich et al., 2018, Voita et al., 2019b, Maruf et al., 2019, Zheng et al., 2020].

# Separate Encoding Approaches

| Reference | Context | Two-Pass Approach | Outside Integr. | Inside Integr. | Lang. Pair |
|---|---|---|---|---|---|
| [Wang et al., 2017] | s:-3 | | aut... | ...aut | Zh→En |
| [Voita et al., 2018] | s:-1 | | yes | | En→Ru |
| [Zhang et al., 2018] | s:-2 | | yes | sequential | Zh→En |
| [Miculicich et al., 2018] | s:-3; t:-3 | | yes | | Zh/Es→En |
| [Maruf et al., 2019] | s:all; t:all | optional | yes | | En→De |
| [Zheng et al., 2020] | s:all; t:all | yes | yes | | Zh/En→En/De |
| [Jean et al., 2017] | s:-1 | | | parallel | En→De/Fr |
| [Bawden et al., 2018] | s:-1; t:-1 | | | parallel | En→Fr |
| [Fu et al., 2019] | s:all | yes | | parallel | En/Zh→De/En |
| [Tan et al., 2019] | s:all | yes | | parallel | Zh/De→En |
| [Voita et al., 2019b] | s:-3; t:-3 | yes | | parallel* | En→Ru |

# Positional Embedding Schema

For separate encoding approaches as well as concatenation approaches, the standard positional encoding proposed by [Vaswani et al., 2017] is insufficient because the DNMT system needs to tell context sentences from the current one. For this reason, many strategies have been proposed in the literature, such as:

1. Adding a **sentence distance embedding** to context sentences, that tell the model how far away they are from the current sentence [Voita et al., 2019b].
2. Assign **positional embeddings progressively** to the current sentence, then to the previous one, and so on, so that far away sentences have high values of positional embedding [**?**].
3. Adding a **segment embedding**, similar to classical positional encoding but for the position of the sentence/segment within the document [Zheng et al., 2020].

Cache approaches to DLNMT consist in encoder-decoder models that are equipped with one or more caches that store context information. The information stored can belong to both **source side or target side, past and future**.

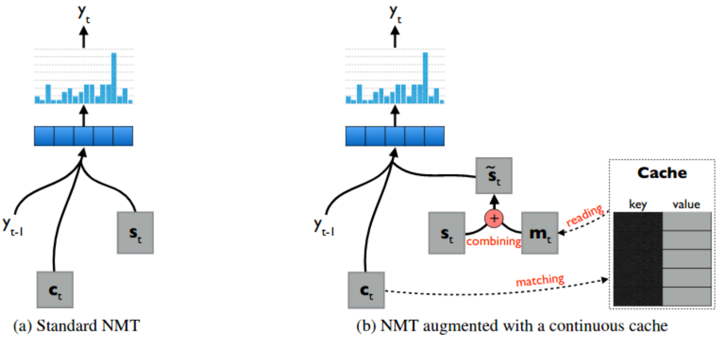
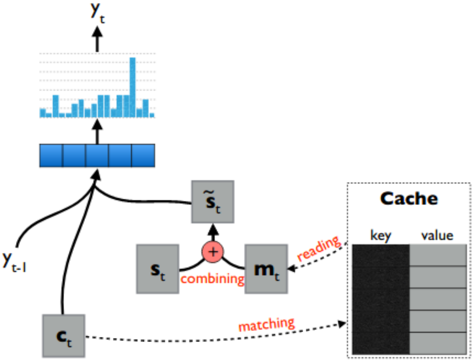



Figure: Continuous cache by [Tu et al., 2017]

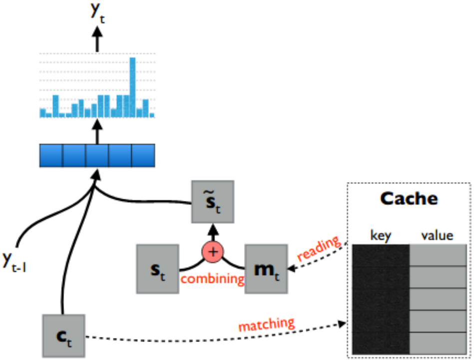
Every cache slot is a **key-value-indicator** triplet (the key and the indicator are often the same thing). With these variables, we can **read** or **write** caches.



# Cache Approaches

**Cache reading** involves:

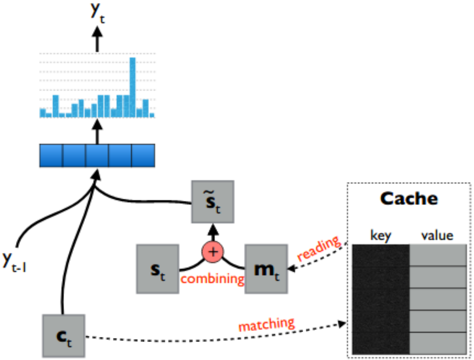
- Soft key matching
- Value reading
- Combining



# Cache Approaches

**Cache writing** can be undertaken after having translated one or more sentences. For every triplet:

- If the **indicator** already exists in the cache, we just update it's keys and values.
- Else, we write the triplet in an empty slot, after having emptied the oldest slot if the cache is full.



# Cache Approaches

| Reference | Caches | Size | Key (Indic.) | Value | Lang. Pair |
|---|---|---|---|---|---|
| [Tu et al., 2017] | single | $\leqslant 500$ | $c_t(\ y_{k<t})$ | $s_{k<t}$ | Zh→En |
| [Kuang et al., 2018] | dynamic<br>topic | 100<br>200 | $c_t$ | $y_{k<t}$<br>topic emb. | Zh→En |
| [Maruf and Haffari, 2018] | source<br>target | doc.size | $h_t$<br>$s_t$ | $sent.emb.$<br>$s_{k<t}$ | Fr/De/Et→En |

# On Parallel Corpora for Training

DLNMT systems require training on **document-level parallel corpora**. These corpora are usually released during workshops on machine translation like IWSLT and WMT, and hosted on open source web inventories. The most common ones, are extracted from:

- Movie subtitles (OpenSubtitles)
- TED talks (WIT3)
- News articles (LDC)
- Parliamentary interventions (Europarl)

# On Parallel Corpora for Training

Unfortunately, document-level parallel corpora are often insufficient to train DLNMT systems from scratch, although it is often possible to make them converge to a local optimum.

[Kim et al., 2019] pointed out that when constraining training on such small datasets, model comparison becomes misleading because gains in performance are mainly related to better regularization.

A popular solution to this problem is the **two-step training strategy**: [Tu et al., 2017, Voita et al., 2018, Miculicich et al., 2018]

1. Distinguish two integrated components in your model with params $\Theta = [\theta_S; \theta_D]$:
   - A self-standing sentence-level NMT system with parameters $\theta_S$.
   - Some context-handling modules with parameters $\theta_D$.
2. Train $\theta_S$ independently on a sentence-level parallel corpus $C_S$.
3. Train together $\theta_D$ and $\theta_S$ (finetune) on a document-level parallel corpus $C_D$.

# Exploiting Monolingual Corpora

Another solution to the lack of vast document-level parallel corpora is leveraging on huge *monolingual* document-level corpora like BookCorpus [Zhu et al., 2015] and PG-19 [Rae et al., 2019]. In the literature, we can find various approaches to leverage monolingual corpora:

- **Back-translate** target-side corpus to augment dl corpus [Sugiyama and Yoshinaga, 2019].
- Train **context-aware language models** on target/source-side corpus, then:
  - Generate translations by fusioning the decoder and the LM's scores to candidate words [Martnez Garcia et al., 2019].
  - Initialize the econder (or decoder) of a DLNMT model [Li et al., 2019].
- Train **Automatic Post Editing** systems on target-side corpus (See next slide).

# Exploiting Monolingual Corpora

**Automatic Post Editing** (APE)
[Voita et al., 2019a] devised an APE system called DocRepair, that turns a sentence-level translation into a context-aware translation. DocRepair can work on top of whatever sentence-level MT system and is lightweight compared to NMT systems.

# Cache Approaches

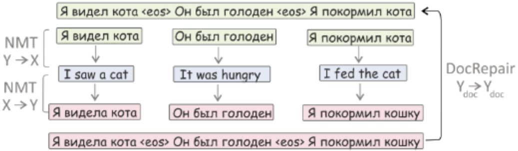



Figure 1: Training procedure of DocRepair. First, round-trip translations of individual sentences are produced to form an inconsistent text fragment (in the example, both genders of the speaker and the cat became inconsistent). Then, a repair model is trained to produce an original text from the inconsistent one.

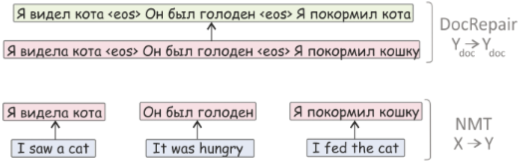



Figure 2: The process of producing document-level translations at test time is two-step: (1) sentences are translated independently using a sentence-level model, (2) DocRepair model corrects translation of the resulting text fragment.

# Others

**Approaches Including Additional Discourse Information as Input**

These approaches consist in concatenation approaches or separate encoding approaches that also integrate discourse-related information as additional input features. Examples of extra features are:

- Lexical chains of semantically similar words to promote word sense disambiguation [Rios Gonzales et al., 2017].
- Coreference chains to promote coreference resolution [Stojanovski and Fraser, 2018, Ohtani et al., 2019].

**Learning Approaches**

[Jean and Cho, 2019] looked at the problem from a learning perspective and designed a regularisation term to encourage a DLNMT model to exploit the additional context in a useful way . This regularisation term is applied at the token, sentence and corpus levels and is based on pair-wise ranking loss, that is, it helps to assign a higher log-probability to a translation paired with the correct context than to the translation without context.

**Possible Future Research Directions**

- build a large DL corpus for training systems, or find automatic approaches to generate synthetic data other than back-translation [Jean et al., 2019]
- design models exploiting full context;
- design models performing good for single-sentence translation [Zheng et al., 2020]
- design post-processing models that are lightweight and can be trained on little data [Kim et al., 2019]. Nonetheless, beware that While this kind of approach is easy to deploy, the two-stage generation process may result in error accumulation.
- pre-trained language model for DLNMT decoder.

Thank you for your attention!

# References I

Agrawal, R. R., Turchi, M., and Negri, M. (2018).
Contextual Handling in Neural Machine Translation: Look Behind, Ahead and on Both Sides.
pages 11–20.
00007 Accepted: 2018-08-08T15:15:28Z.

Bawden, R., Sennrich, R., Birch, A., and Haddow, B. (2018).
Evaluating Discourse Phenomena in Neural Machine Translation.
In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1304–1313, New Orleans, Louisiana. Association for Computational Linguistics.
00055.

# References II


Fu, H., Liu, C., and Sun, J. (2019).
Reference Network for Neural Machine Translation.
In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3002–3012, Florence, Italy. Association for Computational Linguistics.
00000.

Jean, S., Bapna, A., and Firat, O. (2019).
Fill in the Blanks: Imputing Missing Sentences for Larger-Context Neural Machine Translation.
*arXiv:1910.14075 [cs]*.
00000 arXiv: 1910.14075.

# References III


Jean, S. and Cho, K. (2019).
Context-Aware Learning for Neural Machine Translation.
*arXiv:1903.04715 [cs]*.
00003 arXiv: 1903.04715.

Jean, S., Lauly, S., Firat, O., and Cho, K. (2017).
Does Neural Machine Translation Benefit from Larger Context?
*arXiv:1704.05135 [cs, stat]*.
00038 arXiv: 1704.05135.

Kim, Y., Tran, D. T., and Ney, H. (2019).
When and Why is Document-level Context Useful in Neural Machine Translation?
*arXiv:1910.00294 [cs]*.
00001 arXiv: 1910.00294.

# References IV


Kuang, S., Xiong, D., Luo, W., and Zhou, G. (2018).
Modeling Coherence for Neural Machine Translation with Dynamic and Topic Caches.
In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 596–606, Santa Fe, New Mexico, USA. Association for Computational Linguistics.
00012.

Li, L., Jiang, X., and Liu, Q. (2019).
Pretrained Language Models for Document-Level Neural Machine Translation.
*arXiv:1911.03110 [cs]*.
00001 arXiv: 1911.03110.

# References V

Martnez Garcia, E., Creus, C., and Espaa-Bonet, C. (2019).
Context-Aware Neural Machine Translation Decoding.
In *Proceedings of the Fourth Workshop on Discourse in Machine Translation (DiscoMT 2019)*, pages 13–23, Hong Kong, China. Association for Computational Linguistics.
00000.

Maruf, S. and Haffari, G. (2018).
Document Context Neural Machine Translation with Memory Networks.
In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1275–1284, Melbourne, Australia. Association for Computational Linguistics.
00032.

# References VI


Maruf, S., Martins, A. F. T., and Haffari, G. (2018).
Contextual Neural Model for Translating Bilingual Multi-Speaker Conversations.
In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 101–112, Brussels, Belgium. Association for Computational Linguistics.
00002.

Maruf, S., Martins, A. F. T., and Haffari, G. (2019).
Selective Attention for Context-aware Neural Machine Translation.
*arXiv:1903.08788 [cs]*.
00012.

Miculicich, L., Ram, D., Pappas, N., and Henderson, J. (2018).
Document-Level Neural Machine Translation with Hierarchical Attention Networks.
*arXiv:1809.01576 [cs]*.
00024 arXiv: 1809.01576.

Ohtani, T., Kamigaito, H., Nagata, M., and Okumura, M. (2019).
Context-aware Neural Machine Translation with Coreference Information.
In *Proceedings of the Fourth Workshop on Discourse in Machine Translation (DiscoMT 2019)*, pages 45–50, Hong Kong, China. Association for Computational Linguistics.
00000.

Rae, J. W., Potapenko, A., Jayakumar, S. M., and Lillicrap, T. P. (2019).
Compressive Transformers for Long-Range Sequence Modelling.
*arXiv:1911.05507 [cs, stat]*.
00000 arXiv: 1911.05507.

# References VIII

Rios Gonzales, A., Mascarell, L., and Sennrich, R. (2017).
Improving Word Sense Disambiguation in Neural Machine Translation with Sense Embeddings.
In *Proceedings of the Second Conference on Machine Translation*, pages 11–19, Copenhagen, Denmark. Association for Computational Linguistics.
00030.

Scherrer, Y., Tiedemann, J., and Loiciga, S. (2019).
Analysing concatenation approaches to document-level NMT in two different domains.
In *Proceedings of the Fourth Workshop on Discourse in Machine Translation (DiscoMT 2019)*, pages 51–61, Hong Kong, China. Association for Computational Linguistics.
00001.

# References IX


Stojanovski, D. and Fraser, A. (2018).
Coreference and Coherence in Neural Machine Translation: A Study Using Oracle Experiments.
In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 49–60, Brussels, Belgium. Association for Computational Linguistics.
00003.

Sugiyama, A. and Yoshinaga, N. (2019).
Data augmentation using back-translation for context-aware neural machine translation.
In *Proceedings of the Fourth Workshop on Discourse in Machine Translation (DiscoMT 2019)*, pages 35–44, Hong Kong, China. Association for Computational Linguistics.
00000.

# References X


Tan, X., Zhang, L., Xiong, D., and Zhou, G. (2019).
Hierarchical Modeling of Global Context for Document-Level Neural Machine Translation.
In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1576–1585, Hong Kong, China. Association for Computational Linguistics.
00002.

Tiedemann, J. and Scherrer, Y. (2017).
Neural Machine Translation with Extended Context.
In *Proceedings of the Third Workshop on Discourse in Machine Translation*, pages 82–92, Copenhagen, Denmark. Association for Computational Linguistics.
00038.

# References XI


Tu, Z., Liu, Y., Shi, S., and Zhang, T. (2017).
Learning to Remember Translation History with a Continuous Cache.
*arXiv:1711.09367 [cs]*.
00037 arXiv: 1711.09367.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017).
Attention Is All You Need.
*arXiv:1706.03762 [cs]*.
05728 arXiv: 1706.03762.

Voita, E., Sennrich, R., and Titov, I. (2019a).
Context-Aware Monolingual Repair for Neural Machine Translation.
*arXiv:1909.01383 [cs]*.
00003 arXiv: 1909.01383.

# References XII


Voita, E., Sennrich, R., and Titov, I. (2019b).
When a Good Translation is Wrong in Context: Context-Aware Machine Translation Improves on Deixis, Ellipsis, and Lexical Cohesion.
In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1198–1212, Florence, Italy. Association for Computational Linguistics.
00007.

Voita, E., Serdyukov, P., Sennrich, R., and Titov, I. (2018).
Context-Aware Neural Machine Translation Learns Anaphora Resolution.
In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1264–1274, Melbourne, Australia. Association for Computational Linguistics.
00047.

# References XIII


Wang, L., Tu, Z., Way, A., and Liu, Q. (2017).
Exploiting Cross-Sentence Context for Neural Machine Translation.
In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2826–2831, Copenhagen, Denmark. Association for Computational Linguistics.
00045.

Zhang, J., Luan, H., Sun, M., Zhai, F., Xu, J., Zhang, M., and Liu, Y. (2018).
Improving the Transformer Translation Model with Document-Level Context.
In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 533–542, Brussels, Belgium. Association for Computational Linguistics.
00028.

# References XIV


Zheng, Z., Yue, X., Huang, S., Chen, J., and Birch, A. (2020).
Toward Making the Most of Context in Neural Machine Translation.
*arXiv:2002.07982 [cs]*.
00000 arXiv: 2002.07982.

Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., and Fidler, S. (2015).
Aligning Books and Movies: Towards Story-like Visual Explanations by Watching Movies and Reading Books.
*arXiv:1506.06724 [cs]*.
00450 arXiv: 1506.06724.

# Markov Decision Processes

## Reinforcement Learning

General class of algorithms that allow an agent to learn how to behave in a stochastic and possibly unknown environment by trial-and-error.

## Markov Decision Process (MDP)

stochastic dynamical system specified by $< \mathbb{S}, \mathbb{A}, \mathcal{P}, \mathcal{R}, \gamma >$

1. $(\mathbb{S}, \mathcal{S})$ is a measurable state space
2. $(\mathbb{A}, \mathcal{A})$ is a measurable action space
3. $\mathcal{P} : \mathbb{S} \times \mathbb{A} \times \mathcal{S} \to \mathbb{R}$ is a Markov transition kernel
4. $\mathcal{R} : \mathbb{S} \times \mathbb{A} \to \mathbb{R}$ is a reward function
5. $0 < \gamma < 1$ is the discount factor.

# Monte-Carlo Policy Gradient: Pseudocode

**Input:** Stochastic policy $\pi_\theta$, Initial parameters $\theta_0$, learning rate $\{\alpha_k\}$
**Output:** Approximation of the optimal policy $\pi_{\theta*} \approx \pi_*$

**repeat**
  Sample $M$ trajectories $h^{(m)} = \{(s_t^{(m)}, a_t^{(m)}, r_{t+1}^{(m)})\}_{t=0}^{T^{(m)}}$ under policy $\pi_{\theta_k}$
  Approximate policy gradient

$$\nabla_\theta J(\theta_k) \approx \frac{1}{M} \sum_{m=0}^{M} \sum_{u=0}^{T^{(m)}-1} \nabla_\theta \log \pi_{\theta_k} \left( s_u^{(m)}, a_u^{(m)} \right) \sum_{v \geqslant u}^{T^{(m)}-1} \gamma^{v-u} r_{v+1}^{(m)}$$

  Update parameters using gradient ascent $\theta_{k+1} = \theta_k + \alpha_k \nabla_\theta J(\theta_k)$
  $k \leftarrow k + 1$
**until** converged

# Episodic PGPE Algorithm: Pseudocode

**Input:** Controller $F_\theta$, hyper-distribution $p_\xi$, initial guess $\xi_0$, learning rate $\{\alpha_k\}$
**Output:** Approximation of the optimal policy $F_{\xi*} \approx \pi_*$

1: **repeat**
2: **for** $m = 1, \ldots, M$ **do**
3: Sample controller parameters $\theta^{(m)} \sim p_{\xi_k}$
4: Sample trajectory $h^{(m)} = \{(s_t^{(m)}, a_t^{(m)}, r_{t+1}^{(m)})\}_{t=0}^{T^{(m)}}$ under policy $F_{\theta^{(m)}}$
5: **end for**
6: Approximate policy gradient

$$\nabla_\xi J(\xi_k) \approx \frac{1}{M} \sum_{m=1}^{M} \nabla_\xi \log p_\xi \left(\theta^{(m)}\right) \left[G\left(h^{(m)}\right) - b\right]$$

7: Update hyperparameters using gradient ascent $\xi_{k+1} = \xi_k + \alpha_k \nabla_\xi J(\xi_k)$
8: $k \leftarrow k + 1$
9: **until** converged

# Truncated Multiple Importance Sampling Estimator

## Importance Sampling

Given a bounded function $f : \mathcal{Z} \to \mathbb{R}$, and a set of i.i.d. outcomes $z_1, \ldots, z_N$ sampled from $Q$, the importance sampling estimator of $\mu := \underset{z \sim P}{\mathbb{E}}[f(z)]$ is:

$$\widehat{\mu}_{\text{IS}} = \frac{1}{N} \sum_{i=1}^{N} f(z_i) w_{P/Q}(z_i), \tag{1}$$

which is an unbiased estimator, i.e., $\underset{z_i \overset{\text{iid}}{\sim} Q}{\mathbb{E}} [\widehat{\mu}_{IS}] = \mu$.

## Truncated Estimator With Balance Heuristic

$$\check{\mu}_{\text{BH}} = \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{N_k} \min \left\{ M, \frac{p(z_{ik})}{\sum_{j=1}^{K} \frac{N_j}{N} q_j(z_{ik})} \right\} f(z_{ik}). \tag{2}$$

# OPTIMIST2

## Theorem

*regretdiscretized Let $\mathcal{X}$ be a d-dimensional compact arm set with $\mathcal{X} \subseteq [-D, D]^d$. For any $\kappa \geqslant 2$, under Assumptions 1 and 2, OPTIMIST2 with confidence schedule $\delta_t = \dfrac{6\delta}{\pi^2 t^2 \left(1 + \lceil t^{1/\kappa} \rceil^d\right)}$ and discretization schedule $\tau_t = \lceil t^{\frac{1}{\kappa}} \rceil$ guarantees, with probability at least $1 - \delta$:*

$$Regret(T) \leqslant \Delta_0 + C_1 T^{\left(1 - \frac{1}{\kappa}\right)d} + C_2 T^{\frac{1}{1+\epsilon}} \cdot \left[ v_\epsilon \left( (2 + {}^{d}/_{\kappa}) \log T + d \log 2 + \log \frac{\pi^2}{3\delta} \right) \right]^{\frac{\epsilon}{1+\epsilon}},$$

*where $C_1 = \dfrac{\kappa}{\kappa - 1} LD$, $C_2 = (1 + \epsilon) \left( 2\sqrt{2} + \dfrac{5}{3} \right) \|f\|_\infty$, and $\Delta_0$ is the instantaneous regret of the initial arm $\boldsymbol{x}_0$.*