Putting Markov Random Fields on a Tensor Train

Alexander Novikov[1] Anton Rodomanov[1] Anton Osokin[1] Dmitry Vetrov[1,2]

[1]Moscow State University, Moscow, Russia [2]Higher School of Economics, Moscow, Russia

Summary

- Many inference tasks arising in Markov random fields (MRFs) are hard.
- The Tensor Train (TT) decomposition [1] provides a way to compactly represent tensors of very high dimensionality. Many operations on tensors in the TT-format are efficient.
- We consider the energy function and the probability function of an MRF as tensors.
- We tackle the tasks of computing the partition function (normalization constant), estimating marginal distributions and performing the MAP-inference using the machinery provided by the TT-framework.

Tensor Train

TT-format [1] for a tensor $\boldsymbol{A}(x_1, \dots, x_n)$: $(x_i \in \{1, \dots, d\})$

$$\boldsymbol{A}(x_1, \dots, x_n) = \underbrace{G_1^{\boldsymbol{A}}[x_1]}_{1\times r_1(\boldsymbol{A})} \underbrace{G_2^{\boldsymbol{A}}[x_2]}_{r_1(\boldsymbol{A})\times r_2(\boldsymbol{A})} \dots \underbrace{G_n^{\boldsymbol{A}}[x_n]}_{r_{n-1}(\boldsymbol{A})\times 1}.$$

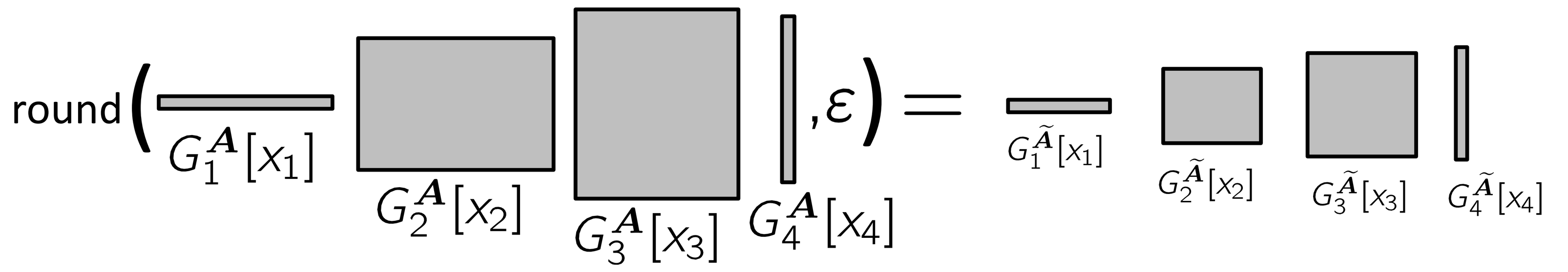
- $G_i^{\boldsymbol{A}}[x_i]$: **TT-cores**; $r_i(\boldsymbol{A})$: **TT-ranks** (crucial for efficiency);
- TT-format requires $O(nd\,r^2(\boldsymbol{A}))$ memory;
- TT-format can also be used for matrices:

$$M(x_1, \dots, x_n;\ y_1, \dots, y_n) = G_1^M[x_1, y_1] \dots G_n^M[x_n, y_n]$$

- **Efficient operations** in the TT-format:

Operation	Output ranks
$\boldsymbol{C} = \boldsymbol{A} \cdot \text{const}$	$r(\boldsymbol{A})$
$\boldsymbol{C} = \boldsymbol{A} + \text{const}$	$r(\boldsymbol{A})+1$
$\boldsymbol{C} = \boldsymbol{A} + \boldsymbol{B}$	$r(\boldsymbol{A})+r(\boldsymbol{B})$
$\boldsymbol{C} = \boldsymbol{A} \odot \boldsymbol{B}$	$r(\boldsymbol{A})\,r(\boldsymbol{B})$
$\boldsymbol{c} = M\boldsymbol{b}$	$r(M)\,r(\boldsymbol{b})$
$\text{sum}\ \boldsymbol{A}$	–
$\lVert\boldsymbol{A}\rVert_F$	–
$\boldsymbol{C} = \text{round}(\boldsymbol{A}, \varepsilon)$	$r(\boldsymbol{C}) \le r(\boldsymbol{A})$

- Most of the operations increase TT-ranks. The **TT-rounding** operation allows to decrease the TT-ranks at the cost of introducing small errors into the representation: $\lVert\mathbf{A} - \widetilde{\mathbf{A}}\rVert_F \le \varepsilon \lVert\mathbf{A}\rVert_F$



There are several general-purpose algorithms for converting tensors into the TT-format:

- **TT-SVD** [1] represents a tensor in the TT-format exactly but suitable only for rather small tensors.
- **AMEn-cross** [2] represents a tensor in the TT-format approximately using only a small fraction of its elements.

MRFs as tensors

Energy: $\boldsymbol{E}(x_1, \dots, x_n) = \sum_{\ell=1}^m \boldsymbol{\Theta}_\ell(\boldsymbol{x}^\ell)$

Probability: $\widehat{\boldsymbol{P}}(x_1, \dots, x_n) = \prod_{\ell=1}^m \boldsymbol{\Psi}_\ell(\boldsymbol{x}^\ell) = \exp(-\boldsymbol{E}(\boldsymbol{x}))$

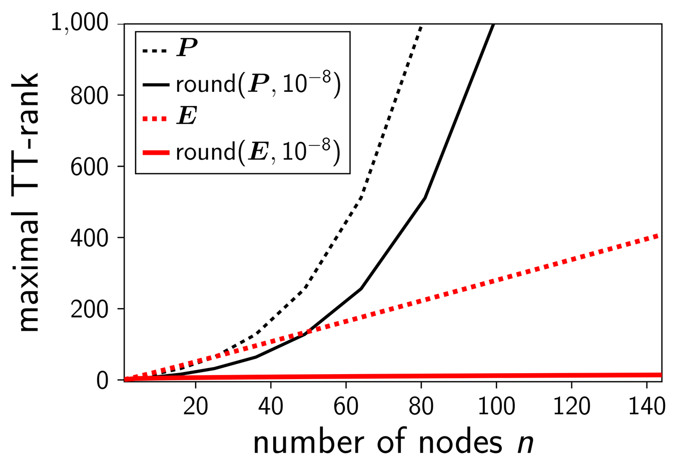
Problems of interest:

1. MAP-inference: $\min_{\boldsymbol{x}} \boldsymbol{E}(\boldsymbol{x})$;
2. Partition function estimation: $Z = \sum_{\boldsymbol{x}} \widehat{\boldsymbol{P}}(\boldsymbol{x})$;
3. Marginal distributions estimation: $\boldsymbol{P}(x_i) = \frac{1}{Z}\sum_{\boldsymbol{x}\setminus x_i} \widehat{\boldsymbol{P}}(\boldsymbol{x})$.

TT-decomposition of MRF tensors

Algorithm for converting MRF tensors into the TT-format:

1. Convert the potentials $\boldsymbol{\Theta}_\ell(\boldsymbol{x})$ (factors $\boldsymbol{\Psi}_\ell(\boldsymbol{x})$) into the TT-format.
2. Use the TT-operations: $\boldsymbol{E}(\boldsymbol{x}) = \sum_{\ell=1}^m \boldsymbol{\Theta}_\ell(\boldsymbol{x})$ $(\widehat{\boldsymbol{P}} = \bigodot_{\ell=1}^m \boldsymbol{\Psi}_\ell)$.



Theorem. The maximal TT-rank of $\boldsymbol{E}$ constructed by the algorithm is polynomially bounded: $r(\boldsymbol{E}) \le d^{\frac{p}{2}} m$, where p is the order of MRF.

Partition function and marginal distributions

TT-ranks of the probability tensor are very large, so it can't be represented in the TT-format. We compute the **partition function** without explicitly building TT-representation for $\widehat{\boldsymbol{P}}$:

1. $\widehat{\boldsymbol{P}}(\boldsymbol{x}) = \prod_{\ell=1}^m \underbrace{\boldsymbol{\Psi}_\ell(\boldsymbol{x})}_{\in\mathbb{R}} = \bigotimes_{\ell=1}^m \boldsymbol{\Psi}_\ell(\boldsymbol{x}) = \bigotimes_{\ell=1}^m \left(G_1^\ell[x_1] \cdots G_n^\ell[x_n]\right)$.
2. $\widehat{\boldsymbol{P}}(\boldsymbol{x}) = \left(G_1^1[x_1] \otimes \dots \otimes G_1^m[x_1]\right) \cdots \left(G_n^1[x_n] \otimes \dots \otimes G_n^m[x_n]\right)$.
   Mixed product property: $AC \otimes BD = (A \otimes B)(C \otimes D)$.
3. $Z = \sum_{\boldsymbol{x}} \widehat{\boldsymbol{P}}(\boldsymbol{x}) = \left(\sum_{x_1} A_1[x_1]\right) \dots \left(\sum_{x_n} A_n[x_n]\right)$,
   where $A_i[x_i] = G_i^1[x_i] \otimes \dots \otimes G_i^m[x_i]$, $A_i[x_i] \in \mathbb{R}^{mp/2 \times mp/2}$.

TT-decomposition of $A_i[x_i]$ can be constructed analytically. Both summation and multiplication are tractable due to the properties of the TT-format.

We proved theoretical upper bounds on the error of the estimation of Z.

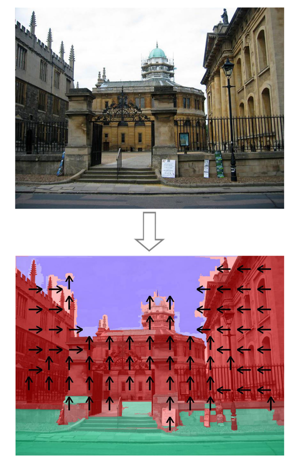
Marginal distributions can be computed similarly:

$$\widehat{\boldsymbol{P}}(x_i) = \left(\sum_{x_1} A_1[x_1]\right) \dots \left(\sum_{x_{i-1}} A_{i-1}[x_{i-1}]\right) A_i[x_i] \left(\sum_{x_{i+1}} A_{i+1}[x_{i+1}]\right) \dots$$

Experiment: MAP-inference

We convert MRF energy into the TT-format and apply the DMRG minimization algorithm [3] to solve the MAP-inference task. Comparison against TRW-S on the OpenGM benchmark [4]:

Problem	Variables	Labels	TRW-S	TT	Time (sec)
gm6	320	3	45.03	43.11	637
gm29	212	3	56.81	56.21	224
gm66	198	3	75.19	74.92	172
gm105	237	3	67.81	67.71	230
gm32	100	7	150.50	289.29	257
gm70	122	7	121.78	163.60	399
gm85	143	7	168.30	228.40	1 912
gm192	99	7	114.51	174.78	180

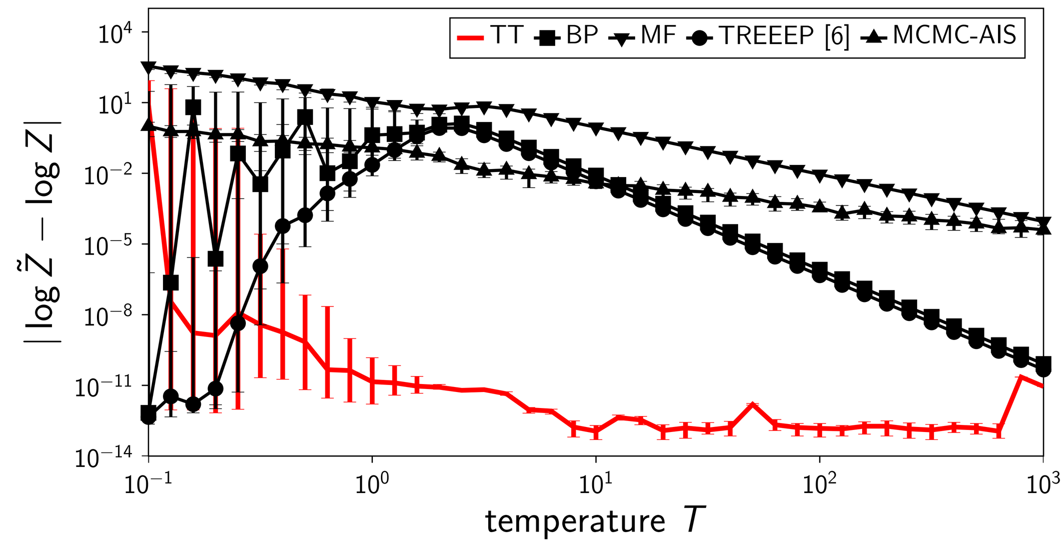


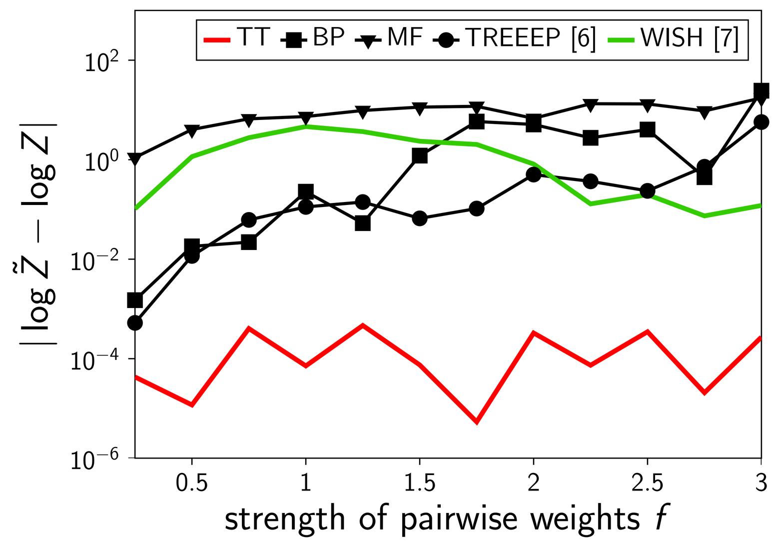
Experiment: partition function and marginal distributions

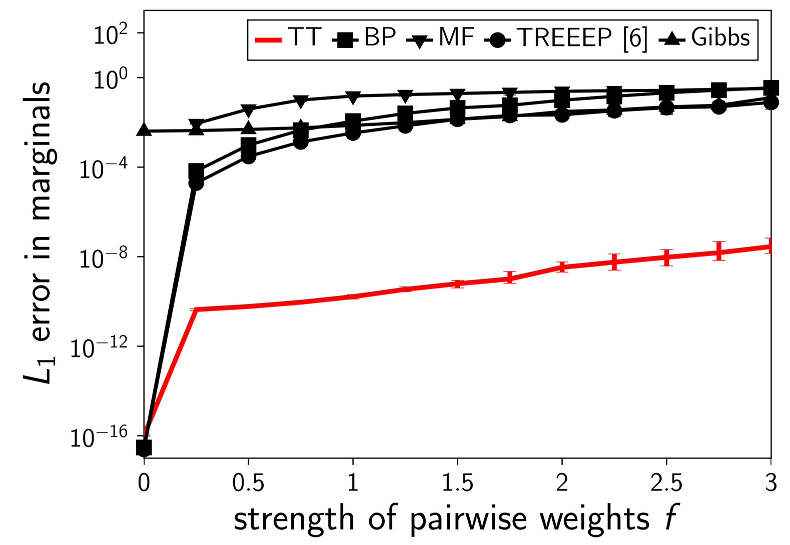
Spin glass model: $\boldsymbol{E}(\boldsymbol{x}) = -\frac{1}{T}\left(\sum_{i=1}^n h_i x_i + f \sum_{(i,j)} c_{ij} x_i x_j\right)$,

where $x_i \in \{-1, 1\}$, $c_{ij} \sim U[-1, 1]$, $h_i \sim U[-1, 1]$.

We compare TT against methods from the LibDAI system [5].







References

[1] I. V. Oseledets. "Tensor-Train Decomposition". In: *SIAM* 33.5 (2011), pp. 2295–2317

[2] I. V. Oseledets and E. E. Tyrtyshnikov. "TT-cross approximation for multidimensional arrays". In: *Linear Algebra Appl.* 432.1 (2010), pp. 70–88

[3] B. N. Khoromskij and I. V. Oseledets. *DMRG+QTT approach to computation of the ground state for the molecular Schrödinger operator.* Preprint 69. Leipzig: MPI MIS, 2010

[4] J. Kappes et al. "A Comparative Study of Modern Inference Techniques for Discrete Energy Minimization Problems". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2013, pp. 1328–1335

[5] J. M. Mooij. "libDAI: A Free and Open Source C++ Library for Discrete Approximate Inference in Graphical Models". In: *JMLR* 11 (Aug. 2010), pp. 2169–2173

[6] T. Minka and Y. Qi. "Tree-structured Approximations by Expectation Propagation". In: *NIPS.* 2004, pp. 193–200

[7] S. Ermon et al. "Taming the Curse of Dimensionality: Discrete Integration by Hashing and Optimization". In: *ICML.* 2013