Differentiable Dynamic Programming for Stack Trace Deduplication

Adem Aber Aoun

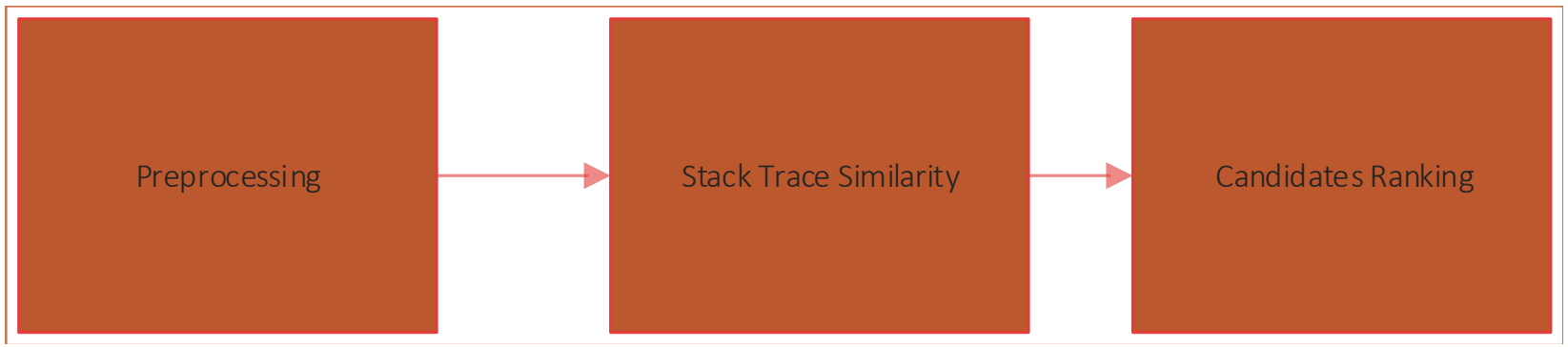
adem.aber-aouni@polymtl.ca

École Polytechnique de Montréal
Laboratoire DORSAL

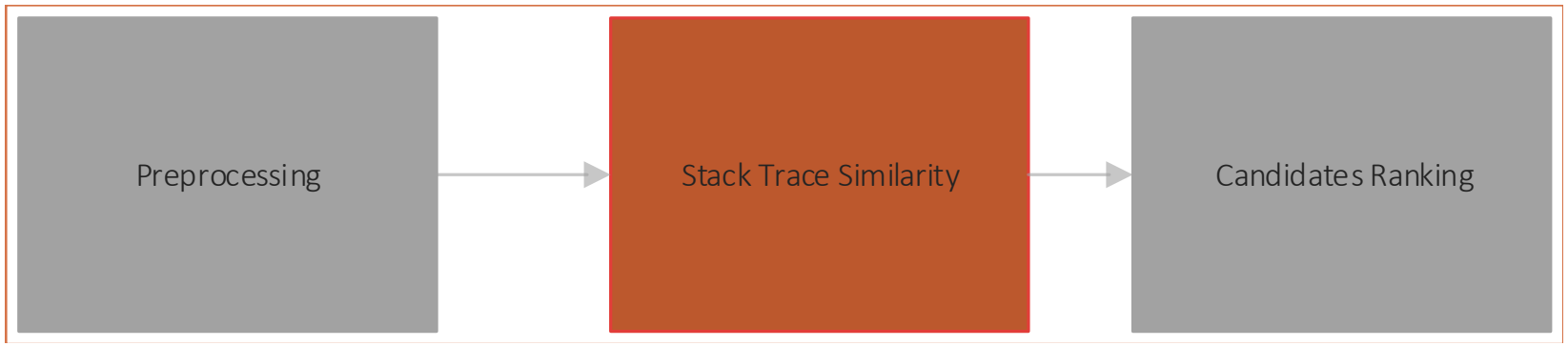
Motivation

- Continuation of the work done by Rodrigues et al. [6] on stack trace deduplication.
- Tune edit-distance penalties for a stack trace dataset quicker.

Bug Deduplication Methodology Pipeline



Bug Deduplication Methodology Pipeline



Dynamic Programming

- "Dynamic programming usually refers to simplifying a decision by breaking it down into a sequence of decision steps over time" [7]
- Used for several problems
  - Fibonacci Serie : $S_i = S_{i-2} + S_{i-1}$
  - Levenshtein distance [8]:

$$lev(a,b) = \begin{cases} |a| & if\ |b| = 0, \\ |b| & if\ |a| = 0, \\ lev(tail(a), tail(b)) & ifa[0] = b[0] \\ 1 + min \begin{cases} lev(tail(a), b) \\ lev(a, tail(b)) \\ lev(tail(a), tail(b)) \end{cases} & otherwise. \end{cases}$$

,where $tail(x)$ is a substring excluding the first character.

differentiation Problem

- Alignment score of two sequences [2] :

$$H_{i,j} = \max \begin{cases} H_{i-1,j-1} + S_{i,j} \\ H_{i-1,j} - d \\ H_{i,j-1} - d \end{cases}$$

, where d is a fixed gap penalty and S_{ij} is the match score.

- How to tune d and/or S_{ij} for a dataset?

differentiation Problem

- Alignment score of two sequences [2] :

$$H_{i,j} = \max \begin{cases} H_{i-1,j-1} + S_{i,j} \\ H_{i-1,j} - d \\ H_{i,j-1} - d \end{cases}$$

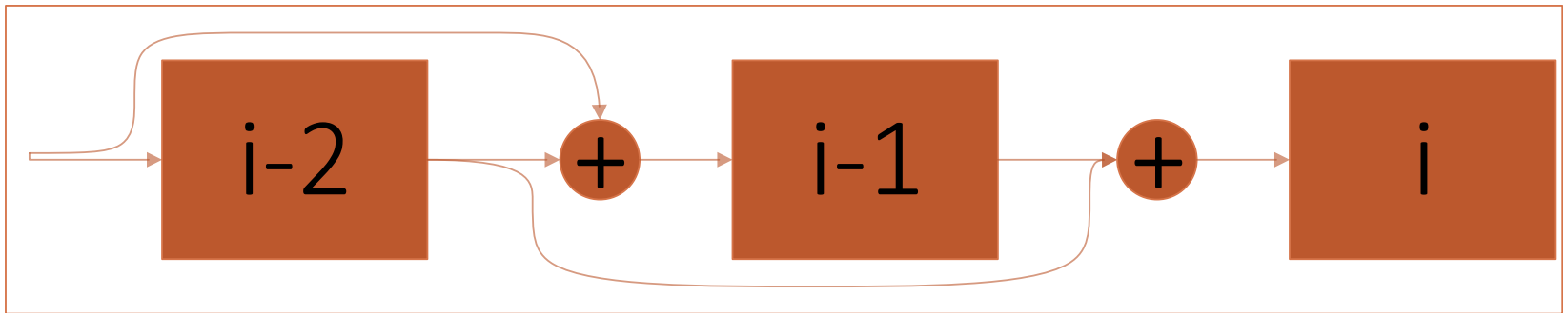
, where d is a fixed gap penalty and S_{ij} is the match score.

- How to tune d and/or S_{ij} for a dataset?
  - Solution 1: Consider them as hyperparameters and perform a grid search
  - Solution 2: Differentiable Dynamic Programming

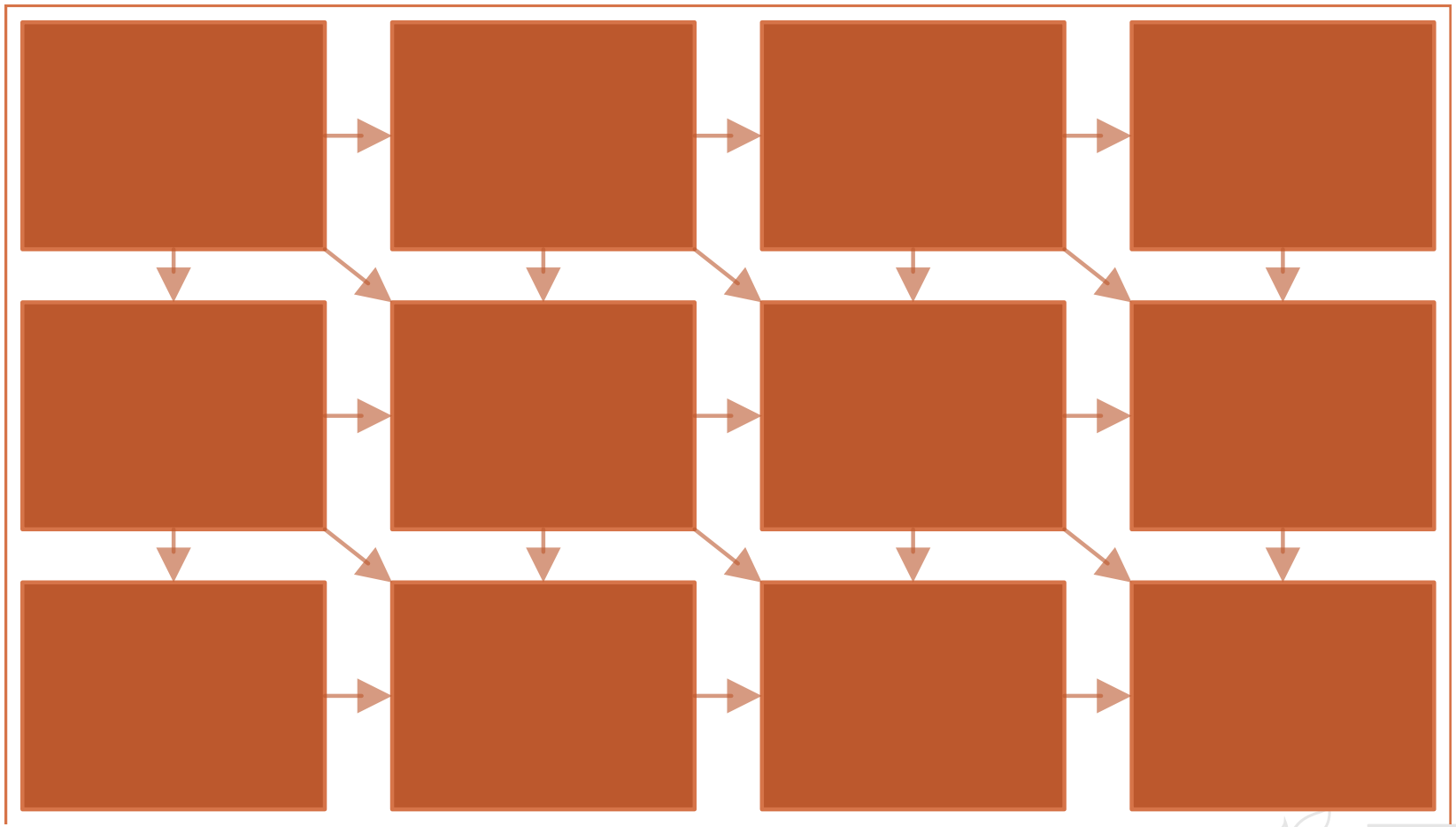
From Dynamic Programming to DDP

- Consider dynamic problem as an RNN problem.
- Each cell can be written as an RNN step.

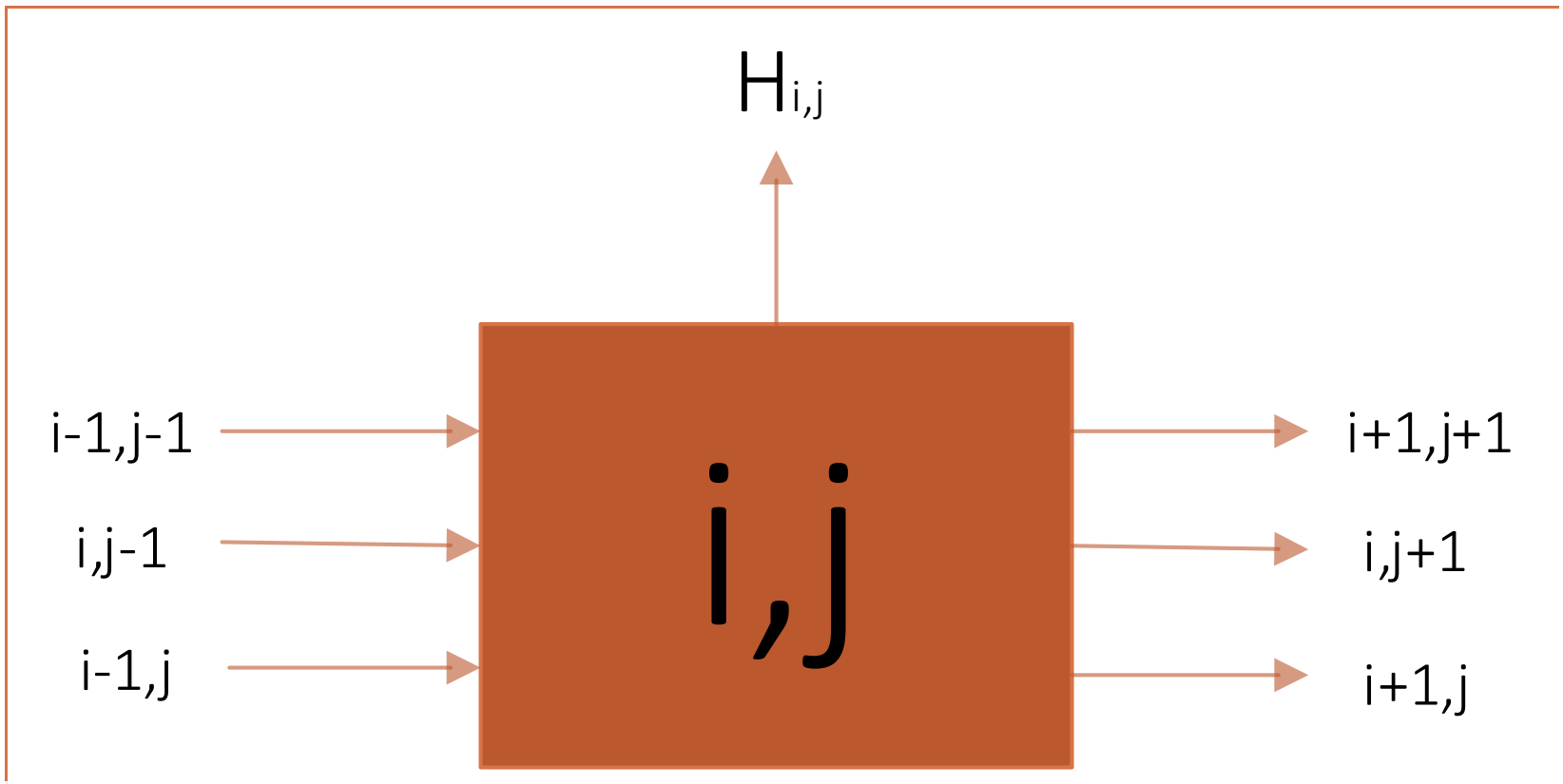
From Dynamic Programming to DDP



From Dynamic Programming to DDP



From Dynamic Programming to DDP



From Dynamic Programming to DDP

- The alignment function will have to be differentiable.
  - Replace *max* function by *RealSoftMax*(LogSumExp).
  - Add a sigmoid node at the end to predict similarity.

Similar Propositions

- Similar approach has been applied on Biological Sequences by Koide et al. [4].
- Promising results with their edit-invariant neural network.

Training

- DDP is a type of RNN : SGD can be used.
- Parameters can be tuned quickly compared to Hyperparmeter Grid-Search.

Potential Issues

- Same problem as RNN (exploding/vanishing gradient).

Our Research

- Alignment score is used in several previous work :
  - Brodie et al. [2]
  - Modani et al. [5]
  - Bartz et al. [1]
  - Dang et al. [3]
  - Rodrigues et al. [6]

Our Research

- DDP can be applied to any of the previous work and parameters and local minimum should be found faster.
- More complex alignment score functions with more parameters can be used.

Thank you!

[1] K Bartz, Jw Stokes, Jc Platt, Ryan Kivett, Ryan Kivett Jack W. Stokes Kevin Bartz, Gretchen Loihle Silviu Calinoiu David Grant John C. Platt, K Bartz, Jw Stokes, Jc Platt, and Ryan Kivett. 2008. Finding Similar Failures Using Callstack Similarity. *SysML*.

[2] Mark Brodie, Sheng Ma, Guy Lohman, Tanveer Syeda-Mahmood, Laurent Mignet, Natwar Modani, Mark Wilding, Jon Champlin, and Peter Sohn. 2005. Quickly finding known software problems via automated symptom matching. *Proceedings - Second International Conference on Autonomic Computing, ICAC 2005*, 2005:101—110.

[3] Yingnong Dang, Rongxin Wu, Hongyu Zhang, Dongmei Zhang, and Peter Nobel. 2012. ReBucket: A Method for Clustering Duplicate Crash Reports Based on Call Stack Similarity. *2012 34th International Conference on Software Engineering (ICSE)*, 1:1084–1093.

[4] Satoshi Koide, Keisuke Kawano, and Takuro Kutsuna. 2018.

Neural edit operations for biological sequences. *Advances in Neural Information Processing Systems*, 31:4960–4970.

[5] Natwar Modani, Rajeev Gupta, Guy Lohman, Tanveer Syeda-Mahmood, and Laurent Mignet. 2007. Automatically Identifying Known Software Problems. *2007 IEEE 23rd International Conference on Data Engineering Workshop*, pages 433–441.

[6] Irving Muller Rodrigues, Aleksandr Khvorov, Daniel Aloise, Roman Vasiliev, Dmitrij Koznov, Eraldo Rezende Fernandes, George Chernishev, Dmitry Luciv, and Nikita Povarov. 2021. Tracesim: An alignment method for computing stack trace similarity. *Empirical Software Engineering manuscript*.

[7] Wikipedia contributors. 2021. Dynamic programming — Wikipedia, the free encyclopedia. [Online; accessed 10-June-2021].

[8] Wikipedia contributors. 2021. Levenshtein distance —

Wikipedia, the free encyclopedia. [Online; accessed 10-June-2021].