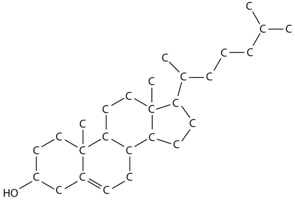
Generative adversarial training on structured domains

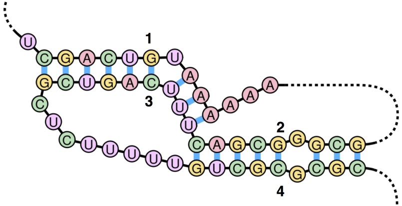
Stefan Mautner Fabrizio Costa
mautner@informatik.uni-freiburg.de
f.costa@exeter.ac.uk

Freiburg University
2017-02-16

Constructive Machine Learning

- **What:** answer design questions using examples
- We are interested in:
  constructive approaches for structured domains
- In chemo- and bio-informatics:
  synthesize molecules with a desired bio-activity





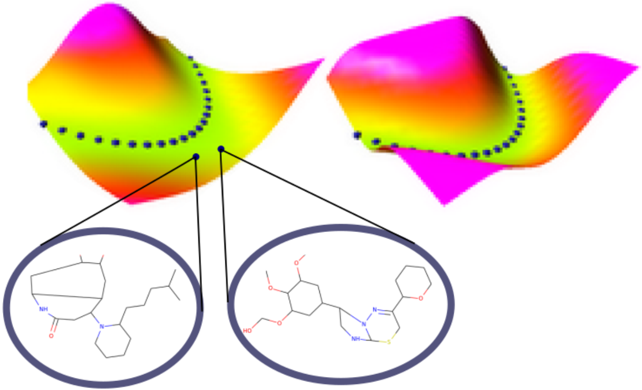
Assumed work

- EDeN (Explicit Decomposition with Neighborhoods)[1]
  - Vectorizes Graphs
  - Used when training model from graphs
  - (not discussed here)
- GraphLearn [2]
  - generates instances given examples
  - (overview given here)
- This is a sampling extension for GraphLearn

[1] github.com/fabriziocosta/EDeN
[2] github.com/fabriziocosta/GraphLearn

The Problem

- Density estimation based on observed graphs (preferably few, "positive" class only)
- Implies loose constraints on feasible manyfold
- Question: How tighten constraints?

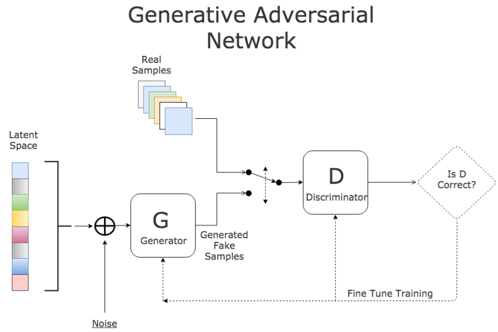


[3]

[3] Yann Le Cun, NIPS 2016 Keynote (modified)

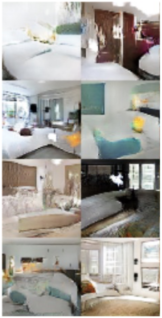
Generative ANN architectures

- Fully visible belief networks (FVBNs)
- Variational autoencoders
- Generative adversarial networks GANs



4

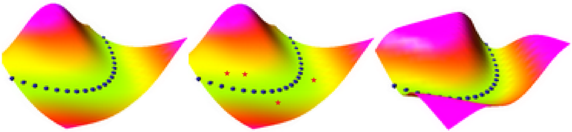
5



[4] Al Gharakhanian, Blogpost, Dec 2016
[5] Ian Goodfellow, NIPS 2016

GAT on structured domains

- The graph generation guiding model might be not tight enough
- ANN researchers inspired us by addressing a very similar problem using GANs
- **Proposal:** Generate instances and assume they are negative examples to improve the generation guiding model



The constructive learning problem for finite samples[6]

- Given a set of graphs G
- use a parametrized generator M_θ to construct set G_θ
- find optimal θ to jointly satisfy:
  1. probability density is the same if estimated over G or G_θ
  2. G_θ differs from G
- Optimize:

$$\arg\min_\theta L(P(G), P(G_\theta)) + \lambda \, Sim(G, G_\theta)$$

- where:
  - L is a loss over probability distributions (e.g. symmetric Kullback Leibler divergence)
  - Sim is a *set* graph kernel
  - λ is desired trade off

[6] Costa Artif. Intell. 2016

Parametrized Generator for Graphs

- Instead of generating $\mapsto$ sample from a corresponding probability distribution over graphs
- We use Metropolis Hastings (MH) Markov Chain Monte Carlo (MCMC)
  1. start from *seed* graph x
  2. propose according to $g(x \mapsto x')$
  3. accept according to:

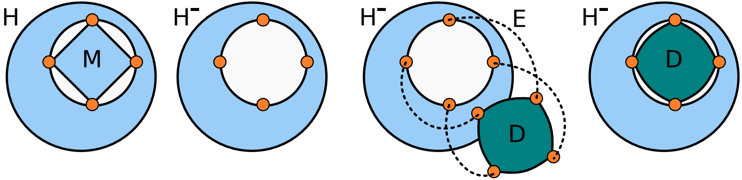
$$A(x \mapsto x') = \min(1, \frac{P(x')}{P(x)} \frac{g(x \mapsto x')}{g(x' \mapsto x)})$$

- **Q:** how not to reject proposed graphs too often?
- **A:** use graph grammar induced from data for $g(x \mapsto x')$

Graph Grammar

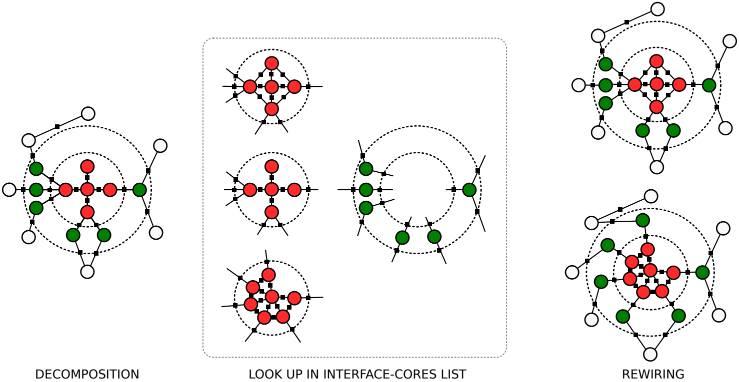
A graph grammar is a finite set of productions P=(M,D,E)

- M=mother graph
- D=daughter graph
- E=embedding mechanism



Substitutable Graph Grammar

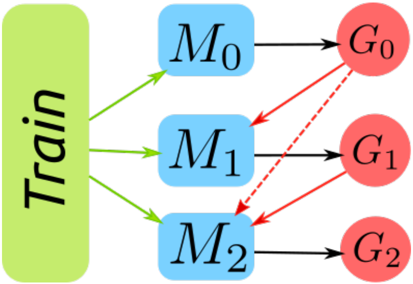
- cores (neighborhood graphs) can be substituted ..
- .. if they have the same interface graph



Generative adversarial training

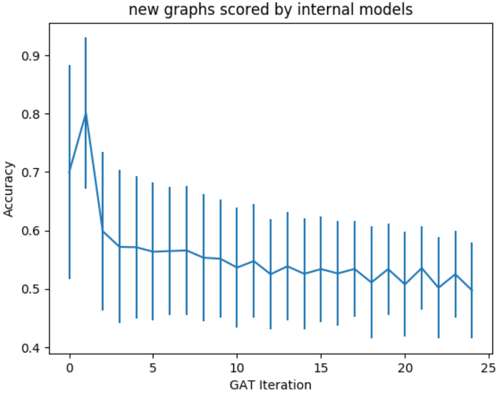
Input: *train*; a set of observed instances

1. train one class model on *train*
2. use *train* as seeds for generation
3. train two class model on *train* (classlabel 1) and all generated instances (classlabel 0)
4. use *train* as seeds for generation
5. goto 3



Training accuracy on internal models

- Are generated instances similar to the observed instances?



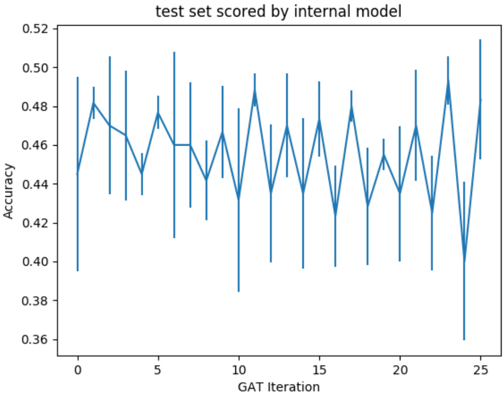
7

lower accuracy indicates that the sets are becoming harder to separate

[7] 500 graphs in train set, 3 repeats, pubchem aid 651610

Test accuracy on internal model

- Is the observed class actually learned? [8]



Lower training accuracy coincides with higher test accuracy

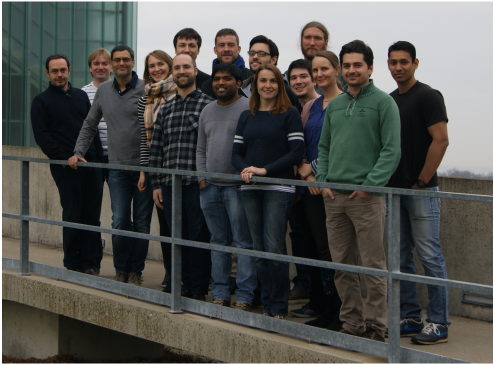
[8] note that the training process has never seen any real negative instances

Conclusion

- Generative adversarial training effective in tightening generation contrains
- Test on larger data set required

The End

- Thank you



- Greetings from Freiburg :D