Interactive HMM construction based on interesting sequences

Szymon Jaroszewicz

National Institute of Telecommunications
Warsaw, Poland

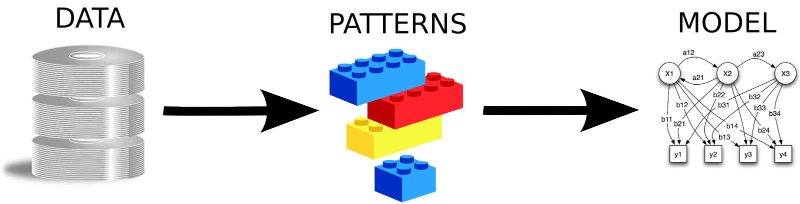
LeGo 2008

Overview

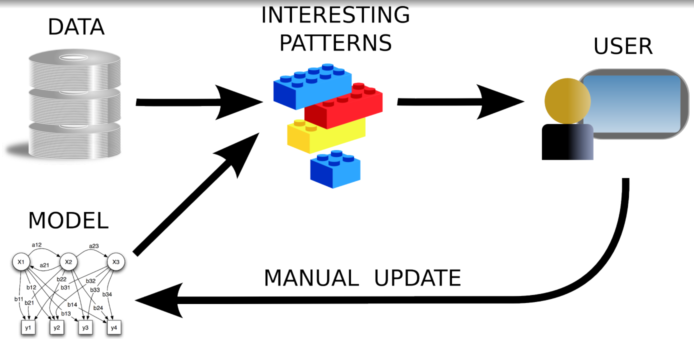
- Building models interactively based on interesting patterns
- Hidden Markov Models
- Interesting patterns w.r.t. Hidden Markov Models
- Experimental evaluation: web server log
- Conclusions and Future research

Typical approach: Automatic model construction

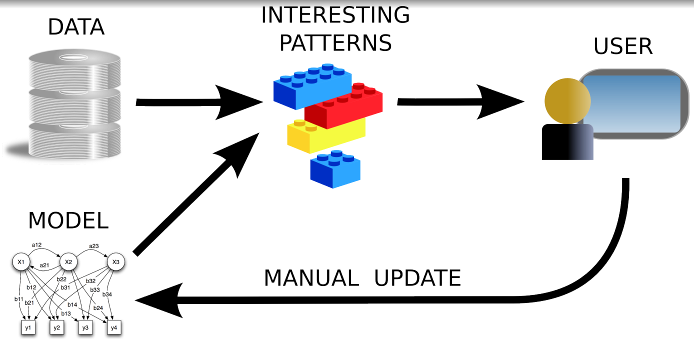
Or:



Here: Interactive model construction



Here: Interactive model construction



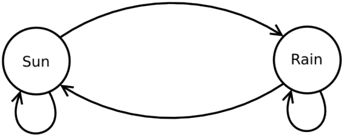
+ Understandable models
+ Learn while building models
– Have to do ‘manual’ work :(

Previous related work

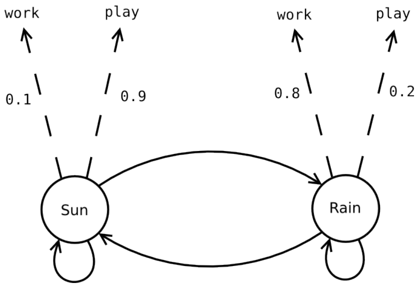
Scalable pattern mining with Bayesian networks as background knowledge
S. Jaroszewicz, T. Scheffer, D. Simovici
KDD'04, KDD'05, DMKD (to appear)

- Bayesian networks used as background model
- Exact and approximate algorithms given
- Models much closer to real relationships than automatically built models

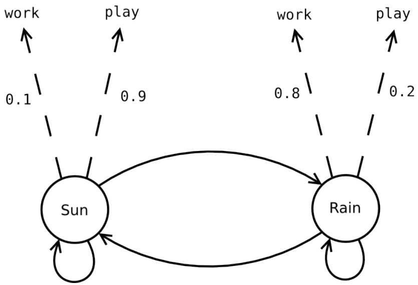
Hidden Markov Models (HMMs)



Hidden Markov Models (HMMs)



Hidden Markov Models (HMMs)



User gives the structure of the HMM:

- internal states
- which transitions are possible (not probabilities)
- which emission symbols are possible for each state (not probabilities)

Interestingness of sequences w.r.t. an HMM

$$Inter(seq) = \left|\mathrm{Prob}^{HMM}\{seq\} - \mathrm{Prob}^{Data}\{seq\}\right|$$

Algorithm for finding all ε-interesting sequences

1. Train *HMM* parameters based on *Data* (Baum-Welch)
2. Find all *seq* such that $\mathrm{Prob}^{\mathit{Data}}\{\mathit{seq}\} > \varepsilon$
3. Find all *seq* such that $\mathrm{Prob}^{\mathit{HMM}}\{\mathit{seq}\} > \varepsilon$
4. Compute $\mathrm{Prob}^{\mathit{Data}}$ for *seq* frequent in *HMM* but not in *Data*
5. Compute $\mathrm{Prob}^{\mathit{HMM}}$ for *seq* frequent in *Data* but not in *HMM*
6. Compute $\mathit{Inter}(\mathit{seq})$ for all sequences
7. Output ε-interesting sequences

Inference in Hidden Markov Models

- Probability that sequence *seq* (starting at $t = 0$) is emitted and *HMM* ends in state s_i

$$\alpha(seq, s_i)$$

- Efficient recursive updating:

$$\alpha(seq + o^{n+1}, s_i) = \sum_j \alpha(seq, s_j)\mathbf{P}_{ji}\mathbf{E}_{io^{n+1}}$$

- $\mathrm{Prob}^{HMM}\{seq\} = \sum_i \alpha(seq, s_i)$

Finding frequent sequences in Hidden Markov Models

- Monotonicity property holds

$$\mathrm{Prob}^{HMM}\{seq + o\} \leq \mathrm{Prob}^{HMM}\{seq\}$$

- Standard depth-first frequent pattern mining works
  alpha probabilities used instead of support counting

- Very efficient: probability updating is fast

Weblog of the National Institute of Telecommunications

Web log format:

```
195.205.118.10 [01/Jan/2007:00:04:33 +0100] "GET
/journal/paper_1.pdf" 200 8833 "http://www.google.pl/"
```

```
65.55.208.68 [01/Jan/2007:00:04:45] "GET /robots.txt" 200
51 "-" "msnbot/1.0"
```

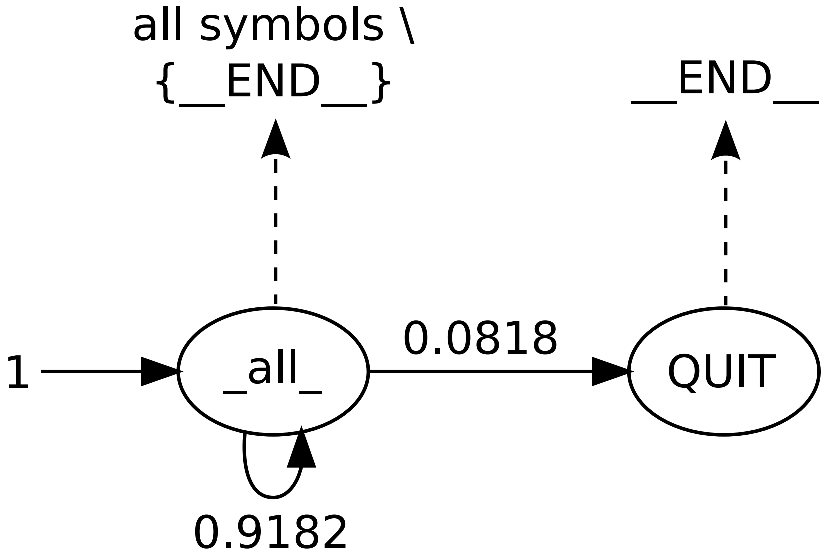
Preprocessing:

- keep only top level directory
- sessionizing

Result: sessions:

```
journal/, journal/, __END__
robots.txt, index.html, journal/, ..., __END__
exchweb/, exchange/, exchange/, ..., __END__
...
```

Initial HMM



The Sophos antivirus

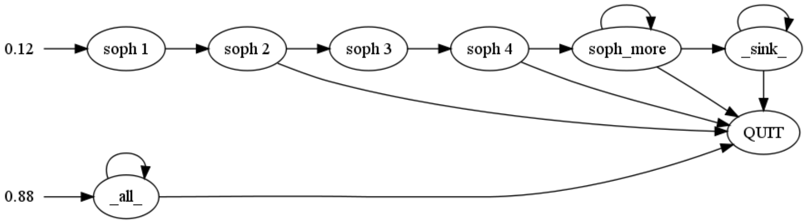
Top sequences:

- `sophos/,sophos/`
  $\mathrm{Prob}^{HMM} = 1.17\%$
  $\mathrm{Prob}^{Data} = 11.48\%$
- `sophos/,sophos/,sophos/,sophos/`
  $\mathrm{Prob}^{HMM} = 0.013\%$
  $\mathrm{Prob}^{Data} = 9.29\%$

- Update of the Sophos antivirus
- **Always** accessed 2, 4 or more times

The Sophos antivirus: update to the model

- The new model is:



- Each soph state only emits the sophos/ symbol
- sophos/ symbol removed from _all_ state

Journal PDF files + icon

- Sequence: `journals/`, `journals/`, `favicon.ico`

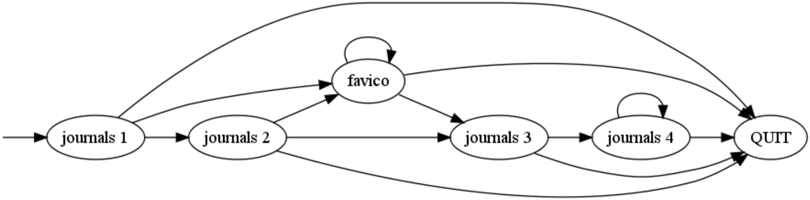
  $$\mathrm{Prob}^{HMM} \approx 0$$
  $$\mathrm{Prob}^{Data} \approx 2\%$$

- `favicon.ico` small icon next to web address 

- Default location: main directory
  At the Institute: `img/` directory
- HTML header contains the other location; PDF can't
- Browser tries the default location and fails
- Fixed: icon appears now

Journal PDF files + icon

- Added the following segment to the model:



- The same PDF file often accessed twice; unable to explain:
  - accelerators?
  - browser errors?
  - server errors?

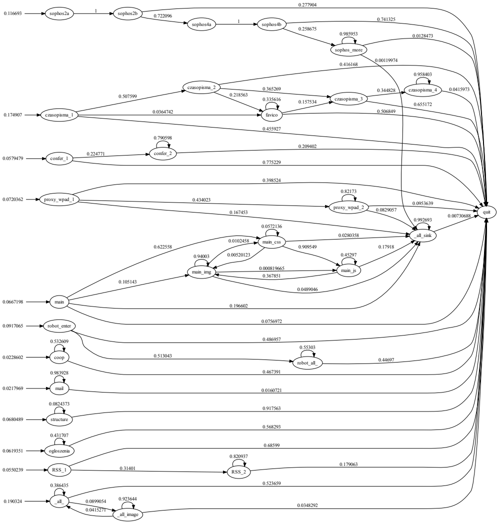
Other patterns

- Exchange mail web reader
- robots: Google / MSN / Yahoo
- RSS readers
- ...

Final model

- Quickly built a model of high level user behavior
- **Accuracy:** probability of all sequences modeled with error < 0.01
  Every sequence is either:
  - uninteresting (modeled well)
  - infrequent
- **Understandability:** the model is easily understandable
- **Learnt** a lot about the data while modeling

Final model

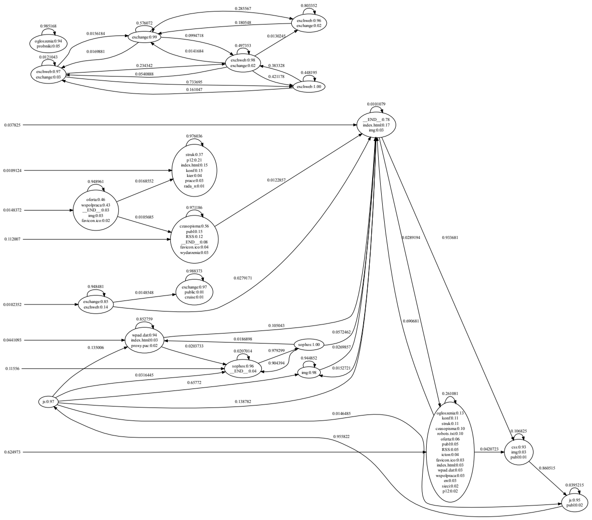


0.116693
sophos2a
1
sophos2b
0.277904
0.722096
sophos4a
1
sophos4b
0.741325
0.258675
0.985953
sophos_more
0.0128473
0.00119974
0.416168
czasopisma_2
0.958403
czasopisma_4
0.0415973
0.507599
0.365269
0.218563
0.335616
0.157534
czasopisma_3
0.344828
0.655172
0.174907
czasopisma_1
0.0364742
favico
0.506849
0.455927
0.790598
confer_2
0.209402
0.0579479
confer_1
0.224771
0.775229
0.398524
0.82173
proxy_wpad_2
0.0953639
0.0720362
proxy_wpad_1
0.434023
0.0829057
quit
0.167453
0.992693
0.00730688
0.0572136
all_sink
main_css
0.0280358
0.0102458
0.622558
0.00520123
0.909549
0.17918
0.94003
0.45297
main_img
0.000819665
main_js
0.367851
0.105143
0.0489046
0.0667198
main
0.196602
0.0756972
0.0917065
robot_enter
0.486957
0.532609
0.55303
0.0228602
coop
0.513043
robot_all_
0.44697
0.467391
0.983928
0.0217969
mail
0.0160721
0.0824373
0.0680489
structure
0.917563
0.431707
0.568293
0.0619351
ogloszenia
0.68599
0.820937
0.0550239
RSS_1
0.31401
RSS_2
0.179063
0.386435
0.523659
0.190324
all
0.0899054
0.923644
0.0415271
_all_image
0.0348292

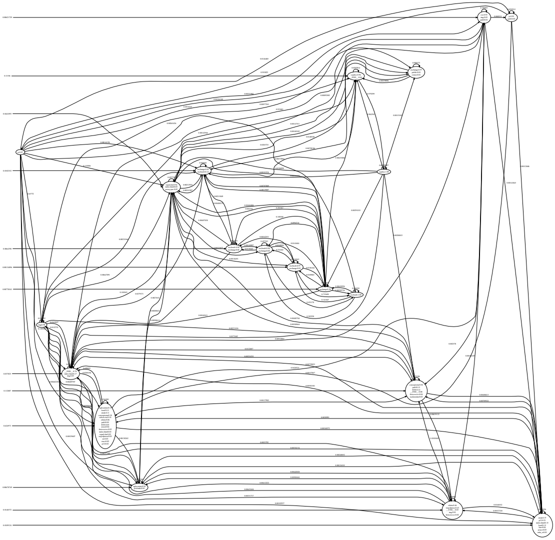
Comparison with automatically learned models

- 20 hidden states + Baum Welch algorithm
- only transitions with prob. > 0.01
- all transitions with prob. > 0.001

Only transitions with prob. > 0.01



All transitions with prob. > 0.001



Conclusions and Future work

Conclusions:

- Interactive model construction based on interesting patterns = **Understandability** + **Accuracy** + **Learning** about the data

Future work:

- Patterns starting at arbitrary time
- More general models: Dynamic Bayesian Networks, models of biological systems
- Automatic model updating (?)