DISCOVERING OPENINGS AND THEIR BALANCE IN COMPETITIVE RTS GAMING

An approach with sequential pattern mining

G. Bosc, M. Kaytoue, C. Raïssi, J.-F. Boulicaut

Machine Learning and Data Mining for Sports Analytics, MLSA@ECML/PKDD 2013 workshop, 27 September 2013, Praha, Czech Republic

A Digital Society

- **Anyone, anytime, anywhere, … *any what***
- **Flourishing video game industry**
  - WORLD OF WARCRAFT (2004) attracted millions of gamers, US 20$/month/user
  - GRAND THEFT AUTO V (2013) : US 200 M$ budget, US 800 M$ sells in 24h
  - CANDY CRUSH SAGA: 45 M users, 15 M users active every days
  - WATCH DOGS (2013), and many others, being adapted to cinema
- A context revealing the former niched **competitive gaming**

Electronic Sport

A new marketing landscape

Entering in popular culture

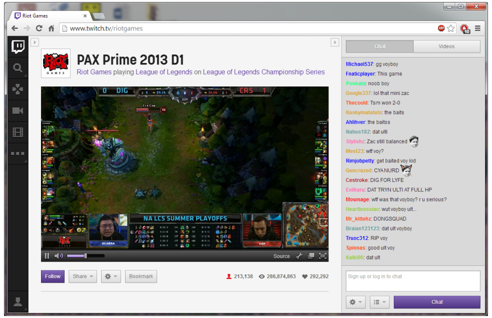
A sport?

E-sport

- **Professional & amateurs** (coaches, teams, fanclubs, streaming)
- **Tournaments** (commentators, sponsors)
  BLIZZARD WCS 2013: *$1.6 million USD*
  DOTA 2 The international Aug. 2013: *$2.2 millions USD*
- Attracting **industrials**, sponsors :
  - Sept. 2013: ***Samsung*** in the process to form a Pro Team (300~500K $)
  - Intel, Nvidia, … community managers
- Followed in great numbers, more and more are making a living of it

T. L. Taylor
Raising the Stakes :E-Sports and the Professionalization of Computer Gaming.
In *MIT Press*. 2012.

M. Kaytoue, A. Silva, L. Cerf, Wagner Meira Jr. et C. Raïssi
Watch me playing, i am a professional : a first study on video game live streaming.
In *WWW 2012 (Companion Volume), pages 1181–1188*. ACM, 2012.



PAX Prime 2013 D1
Riot Games playing League of Legends on League of Legends Championship Series

Michael537: gg voyboy
Fnaticplayer: This game
Ponkata: noob boy
Google337: lol that mini zac
Thecoold: Tsm won 2-0
Kuskymatatuto: the baits
Ahlihver: the baitss
Nahus182: dat ulti
Stylishz: Zac still balanced
Mest23: wtf voy?
Rimjobpetty: get baited voy kid
Guncrazed: CYA NURD
Cestroke: DIG FOR LYFE
Evilharu: DAT TRYN ULTI AT FULL HP
Mownage: wtf was that voyboy? r u serious?
Heartlessxiao: wut voyboy ult...
Mr_kittehz: DONGSQUAD
Braian123123: dat ult voyboy
Truoc312: RIP voy
Spinnas: good ult voy
Kalki90: dat ult

StarCraft II

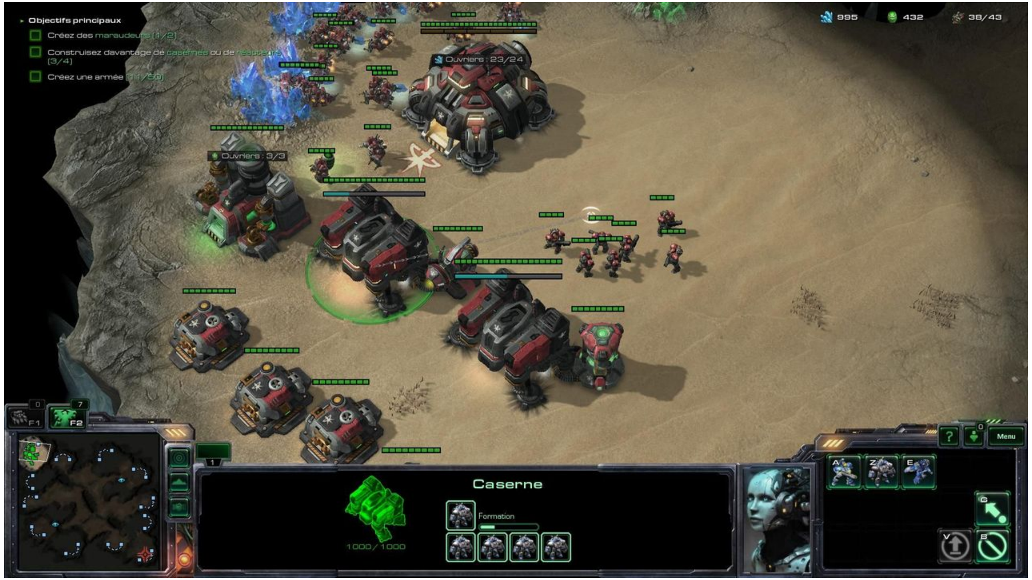
Main Principes of RTS games

Build orders openings

Strategy balancing

A Real-Time Strategy Video Game

- A **map**, aerial view (like gmap)
- **2 players** in command (1v1)
- Controls units and building
  - **Economy** Gather resources
  - **Production and technology** Construct new buildings
  - **Training** combat units from buildings
  - Order to **explore** the map (fog of war)
  - Order to **attack** an enemy
- **Winning condition** : opponent resigns or has no more forces

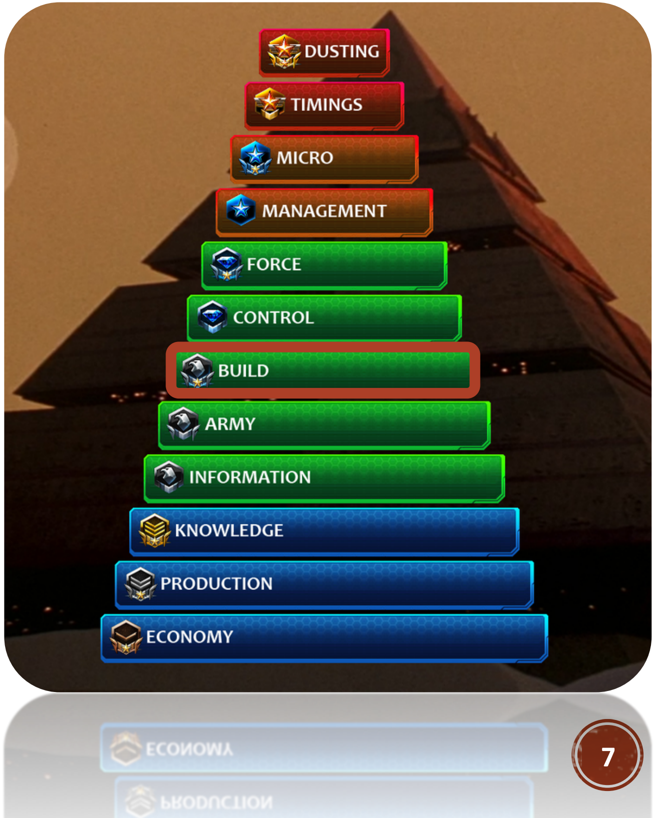

Objectifs principaux
□ Créez des maraudeurs (1/2)
□ Construisez davantage de casernes ou de réacteurs (3/4)
□ Créez une armée (11/30)
995
432
38/43
Ouvriers : 23/24
Ouvriers : 3/3
F1
F2
Menu
Caserne
Formation
1000 / 1000

What skills make professionals… payed to play

- Can operate **fast multi-decisions making process in uncertain environment**
  - **Multi-tasking** (macro, micro, several fronts, timings, scouting…)
  - **Velocity** (200 to 400 clicks/Key pressed per minute)
  - **Decisions** Rock/Paper/Scissors principle, army composition, attacks, …
  - **Uncertainty** under a fog of war
  - **Actions may fail** misclicks

It requires strategies, training, meta-gaming, etc.

Automatic discovery of build orders openings
A challenge for game editors and (pro)-gamers



Methodology

Data as traces of interaction

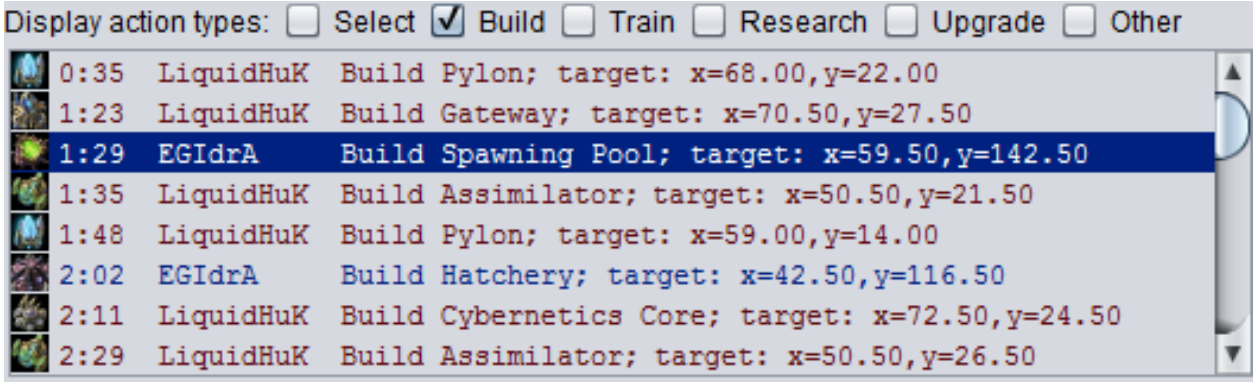
Sequential pattern mining

Build order opening discovery

Balance of an opening

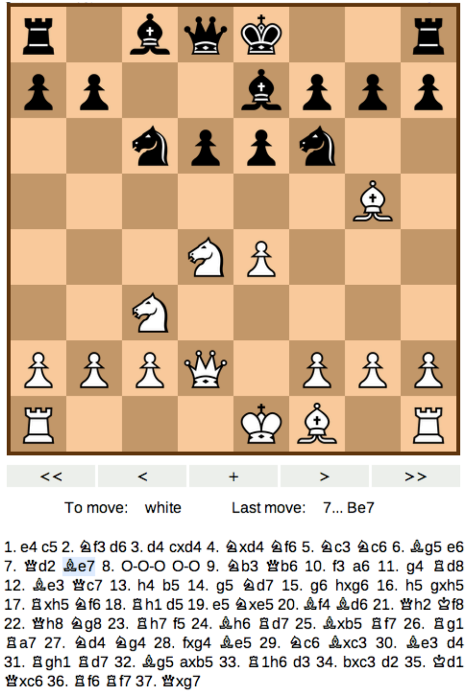
StarCraft 2 replay

- A series of **timestamped** actions of different types for both players



Display action types: □ Select ☑ Build □ Train □ Research □ Upgrade □ Other

```
0:35  LiquidHuK  Build Pylon; target: x=68.00,y=22.00
1:23  LiquidHuK  Build Gateway; target: x=70.50,y=27.50
1:29  EGIdrA     Build Spawning Pool; target: x=59.50,y=142.50
1:35  LiquidHuK  Build Assimilator; target: x=50.50,y=21.50
1:48  LiquidHuK  Build Pylon; target: x=59.00,y=14.00
2:02  EGIdrA     Build Hatchery; target: x=42.50,y=116.50
2:11  LiquidHuK  Build Cybernetics Core; target: x=72.50,y=24.50
2:29  LiquidHuK  Build Assimilator; target: x=50.50,y=26.50
```



To move: white   Last move: 7... Be7

1. e4 c5 2. ♘f3 d6 3. d4 cxd4 4. ♘xd4 ♘f6 5. ♘c3 ♘c6 6. ♗g5 e6 7. ♕d2 ♗e7 8. O-O-O O-O 9. ♘b3 ♕b6 10. f3 a6 11. g4 ♖d8 12. ♗e3 ♕c7 13. h4 b5 14. g5 ♘d7 15. g6 hxg6 16. h5 gxh5 17. ♖xh5 ♘f6 18. ♖h1 d5 19. e5 ♘xe5 20. ♗f4 ♗d6 21. ♕h2 ♔f8 22. ♕h8 ♘g8 23. ♖h7 f5 24. ♗h6 ♖d7 25. ♗xb5 ♖f7 26. ♖g1 ♖a7 27. ♘d4 ♘g4 28. fxg4 ♗e5 29. ♘c6 ♗xc3 30. ♗e3 d4 31. ♖gh1 ♖d7 32. ♗g5 axb5 33. ♖1h6 d3 34. bxc3 d2 35. ♔d1 ♕xc6 36. ♖f6 ♖f7 37. ♕xg7

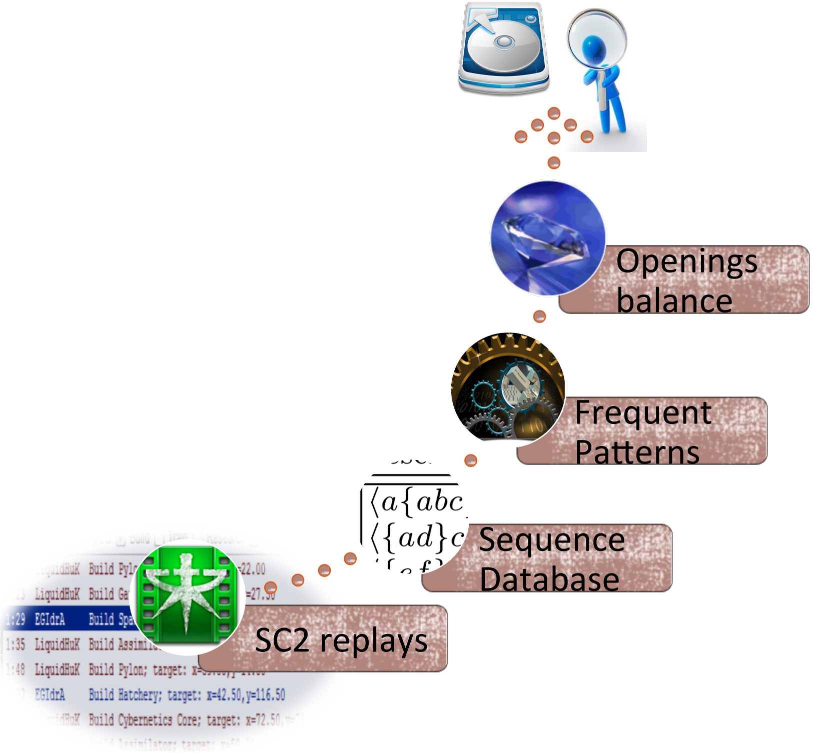
(Frequent) Sequential Pattern Mining

id	sequence
1	<a{abc}{ac}d{cf}>
2	<{ad}c{bc}{ae}>
3	<{ef}{ab}{df}cb>
4	<eg{af}cbc>

- **Sequence** a series of events from an alphabet
- **Sequence database** a set of sequences
- **Frequent sequential pattern** an arbitrary sequence more general than (supported by) a *given* proportion of the database
- E.g. Setting minimal support to 75%
  - <acc> is frequent
  - <a{bc}a> is not frequent

Openings as Sequential Patterns

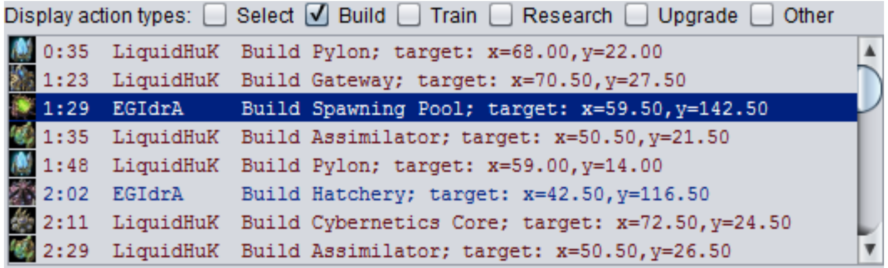
- **How to turn replays files to sequences**
  - Events?
  - Windows of time?
  - One or two players?
- **How to validate extracted patterns**
  - Measuring the balance of the strategy?
- **How to analyze the results**
  - Search & Sort & Filter
  - Patterns to be re-used for various task



```
   LiquidHuK  Build Pyl          =22.00
:23 LiquidHuK  Build Ga          =27.50
1:29 EGIdrA    Build Spa
1:35 LiquidHuK  Build Assimil
1:48 LiquidHuK  Build Pylon; target: x=
     EGIdrA    Build Hatchery; target: x=42.50,y=116.50
     uidHuK    Build Cybernetics Core; target: x=72.50,y=
```

Encoding a SC2 replay as a sequence

- A **sequence** is a **series of windows** of **time composed of actions** done by both players
  To differentiate players, events are tagged as **winner (success) or looser (fail)** (1v1 situation)
- **Events/Actions** are elements of $\mathcal{I} = A \times T \times \{success, fail\}$



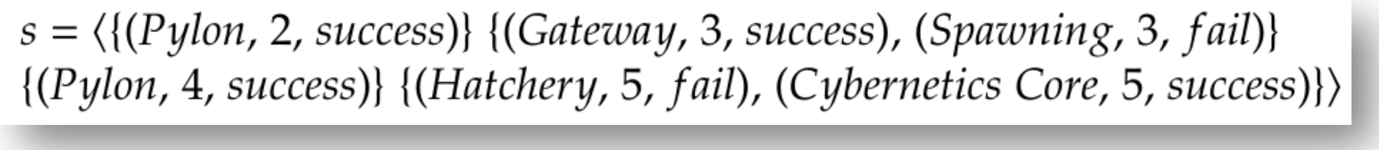
Display action types: □ Select ☑ Build □ Train □ Research □ Upgrade □ Other

```
0:35  LiquidHuK  Build Pylon; target: x=68.00,y=22.00
1:23  LiquidHuK  Build Gateway; target: x=70.50,y=27.50
1:29  EGIdrA     Build Spawning Pool; target: x=59.50,y=142.50
1:35  LiquidHuK  Build Assimilator; target: x=50.50,y=21.50
1:48  LiquidHuK  Build Pylon; target: x=59.00,y=14.00
2:02  EGIdrA     Build Hatchery; target: x=42.50,y=116.50
2:11  LiquidHuK  Build Cybernetics Core; target: x=72.50,y=24.50
2:29  LiquidHuK  Build Assimilator; target: x=50.50,y=26.50
```

$$s = \langle \{(Pylon, 2, success)\} \ \{(Gateway, 3, success), (Spawning, 3, fail)\} \ \{(Pylon, 4, success)\} \ \{(Hatchery, 5, fail), (Cybernetics\ Core, 5, success)\} \rangle$$

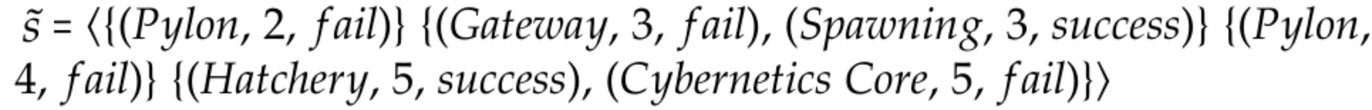
Measuring the balance of a strategy

- Given a frequent sequential pattern extract with a FPM algorithm



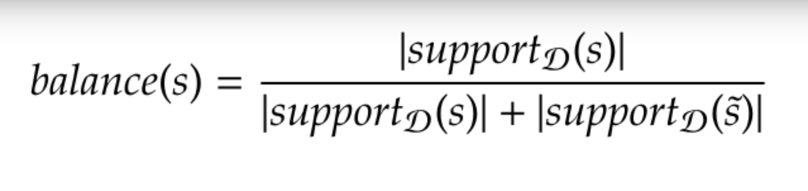
$$s = \langle\{(Pylon, 2, success)\}\ \{(Gateway, 3, success), (Spawning, 3, fail)\}\ \{(Pylon, 4, success)\}\ \{(Hatchery, 5, fail), (Cybernetics\ Core, 5, success)\}\rangle$$

- its dual represent the reversed scenario, where the winner would have lost



$$\tilde{s} = \langle\{(Pylon, 2, fail)\}\ \{(Gateway, 3, fail), (Spawning, 3, success)\}\ \{(Pylon, 4, fail)\}\ \{(Hatchery, 5, success), (Cybernetics\ Core, 5, fail)\}\rangle$$

- And then



$$balance(s) = \frac{|support_{\mathcal{D}}(s)|}{|support_{\mathcal{D}}(s)| + |support_{\mathcal{D}}(\tilde{s})|}$$

Balance Properties

- **Takes values in**

$$balance(s) \in [0;1]$$

- **Equilibrium**

$$balance(s) = 0.5$$

- **Un-balanced strategy**

$$balance(s) = 1 \text{ or } 0$$

- **Mirror strategy**

$$balance(s) = 0.5$$

$$balance(s) = \frac{|support_{\mathcal{D}}(s)|}{|support_{\mathcal{D}}(s)| + |support_{\mathcal{D}}(\tilde{s})|}$$

$$balance(s) + balance(\tilde{s}) = 1$$

Computing frequent patterns and their balance

- **A naïve approach**
  - **Mining step**: Extract all frequent sequential patterns with a known algorithm (e.g. PrefixSpan)
  - **Post-processing step:** count how many times the dual of each pattern appears in the base

✗

Not scalable

- **A more elaborated approach**
  - **Mining step**: Extract all frequent sequential patterns with a know algorithm (e.g. PrefixSpan)
  - **Post-processing step:** Take advantage of the tree data-structure of the extracted patterns

✓

Faster computation of the balance
Avoid redundant patterns

More technical and algorithmic details in the proceedings

Experiments

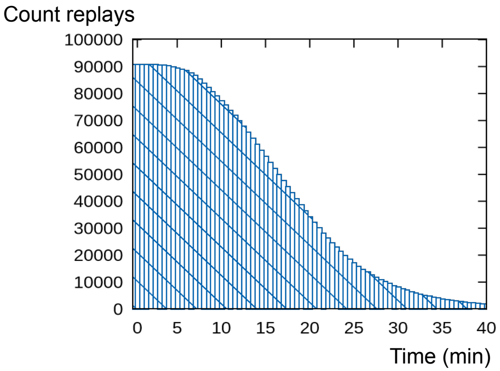
Dataset

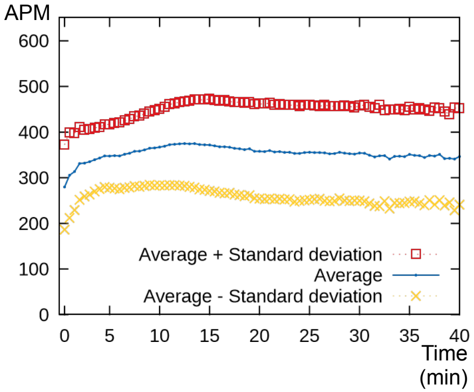
Quantitative results

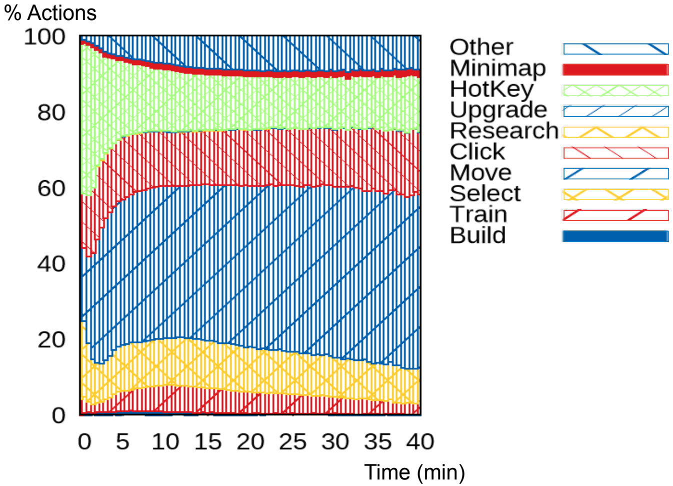
Qualitative results

Dataset

- Restricting ***371 267* SC2 replays** harvested on several dedicated websites
- **90 768 games** (1v1 only, APM>200, Master/Grandmaster)



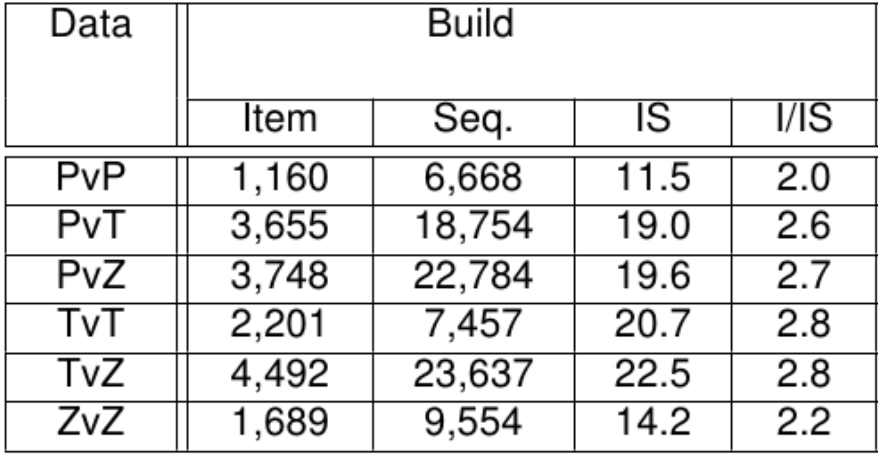




G. Bosc et al. - Machine Learning and Data Mining for Sports Analytics, ECML/PKDD 2013 workshop, 27 September 2013, Prague, Czech Republic

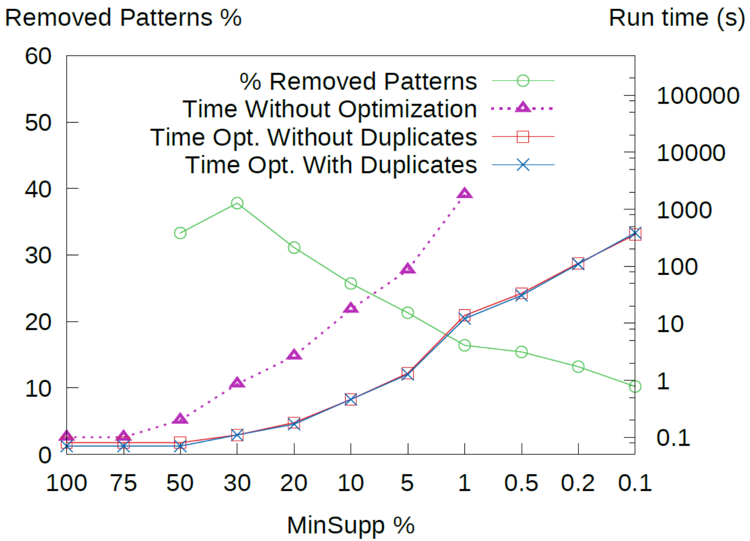
Experimental settings

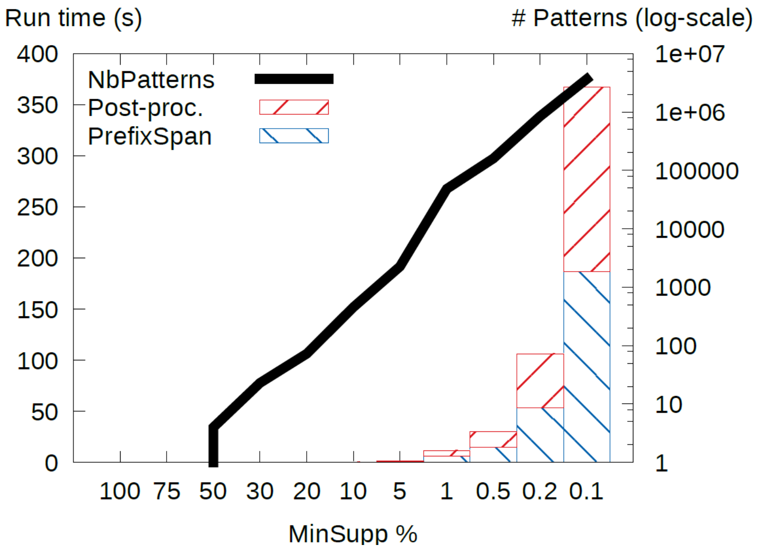
- One sequence database for each **match-up** (PvsP, PvsT, PvsZ, TvsZ, TvsT, ZvsZ)
- ‘**build-orders**’ actions
- **Windows** of time set to 30 seconds



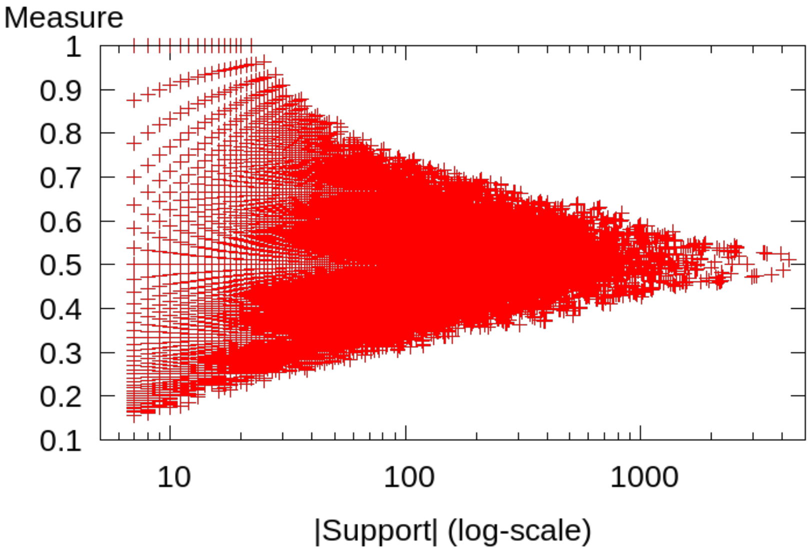
Data	Build			
	Item	Seq.	IS	I/IS
PvP	1,160	6,668	11.5	2.0
PvT	3,655	18,754	19.0	2.6
PvZ	3,748	22,784	19.6	2.7
TvT	2,201	7,457	20.7	2.8
TvZ	4,492	23,637	22.5	2.8
ZvZ	1,689	9,554	14.2	2.2

Quantitative experiments





Quantitative experimentations



Top frequent patterns are more balanced

They should represent balanced and well-known openings

Qualitative experiments

Well known openings, broken strategies, un-diversity in mirror matchups, etc.

Forge Expand

- 9: Pylon
- 14: Forge
- 17: Nexus
- 17: Pylon
- 18: Gateway
- 18: Photon Cannon
- 18: Assimilator x2

minSupp: 5%
591 patterns

*balance(s)=***0.52**

1-1-1 Build

- 12: Barracks
- 13: Refinery
- 17: Refinery
- 18: Factory
- 20: TechbLab
- 22: Starport

minSupp: 5%
400 patterns

*balance(s)=***0.46**

4 gates

- 9: Pylon
- 12: Gateway
- 14: Assimilator
- 15: Pylon
- 17: Cybernetics pCore
- 22: Warpgate
- 24: Gateway px3
- 26: Pylon
- 26: Assimilator

minSupp: 5%
3148 patterns

*balance(s)=***0.59**

Bunker rush IMBA!

Leveraged by Blizzard
(Queen range+, Supply before barracks)

balance(s) < **0.6**
balance(s) > **0.6**

Conclusion

Conclusion

Data-mining

- ***Emerging pattern mining:*** items are labelled, not the sequence itself
- ***Scalability****:* Computing the balance measure double the execution time in worst cases
- ***Other applications****:* two agents compete for a resource, one succeed while the other fails (sport analytics, AI, planning, …)

Video game analytics and E-Sport

- ***Balance*** is one of the hottest issue in the E-Sport scene
- ***Service*** that can help pro-gamers and their team to prepare match (e.g. after a SQL query on replays)
- Non professional **entertainment**

Perspectives

- Patterns to be re-used as 'knowledge unit' for other tasks, e.g. real-time winning prediction
- Computing balance during the mining step and improving specific pruning strategies
- New dimensions to characterize emerging patterns:
  This strategy is unbalanced in map x during season y for player z

Source code, dataset and questions
http://liris.cnrs.fr/mehdi.kaytoue

DISCOVERING OPENINGS AND THEIR BALANCE IN COMPETITIVE RTS GAMING

An approach with sequential pattern mining

G. Bosc, M. Kaytoue, C. Raïssi, J.-F. Boulicaut

Machine Learning and Data Mining for Sports Analytics, MLSA@ECML/PKDD 2013 workshop, 27 September 2013, Praha, Czech Republic