Machine Learning applied to Go

Dmitry Kamenetsky

Supervisor: Nic Schraudolph

September 2006

Aim

- Create a clean and accessible database of Go games
- Use Machine Learning to learn from expert games
- Use the learned knowledge to build a powerful Go program

Aim

- Create a clean and accessible database of Go games
- Use Machine Learning to learn from expert games
- Use the learned knowledge to build a powerful Go program

Aim

- Create a clean and accessible database of Go games
- Use Machine Learning to learn from expert games
- Use the learned knowledge to build a powerful Go program

What is Go?

- Two-player zero-sum board game
- Originated in ancient China. Today very popular in China, Japan and Korea
- 19x19 grid
- Simple rules, but very complex strategy

Go example

Go example

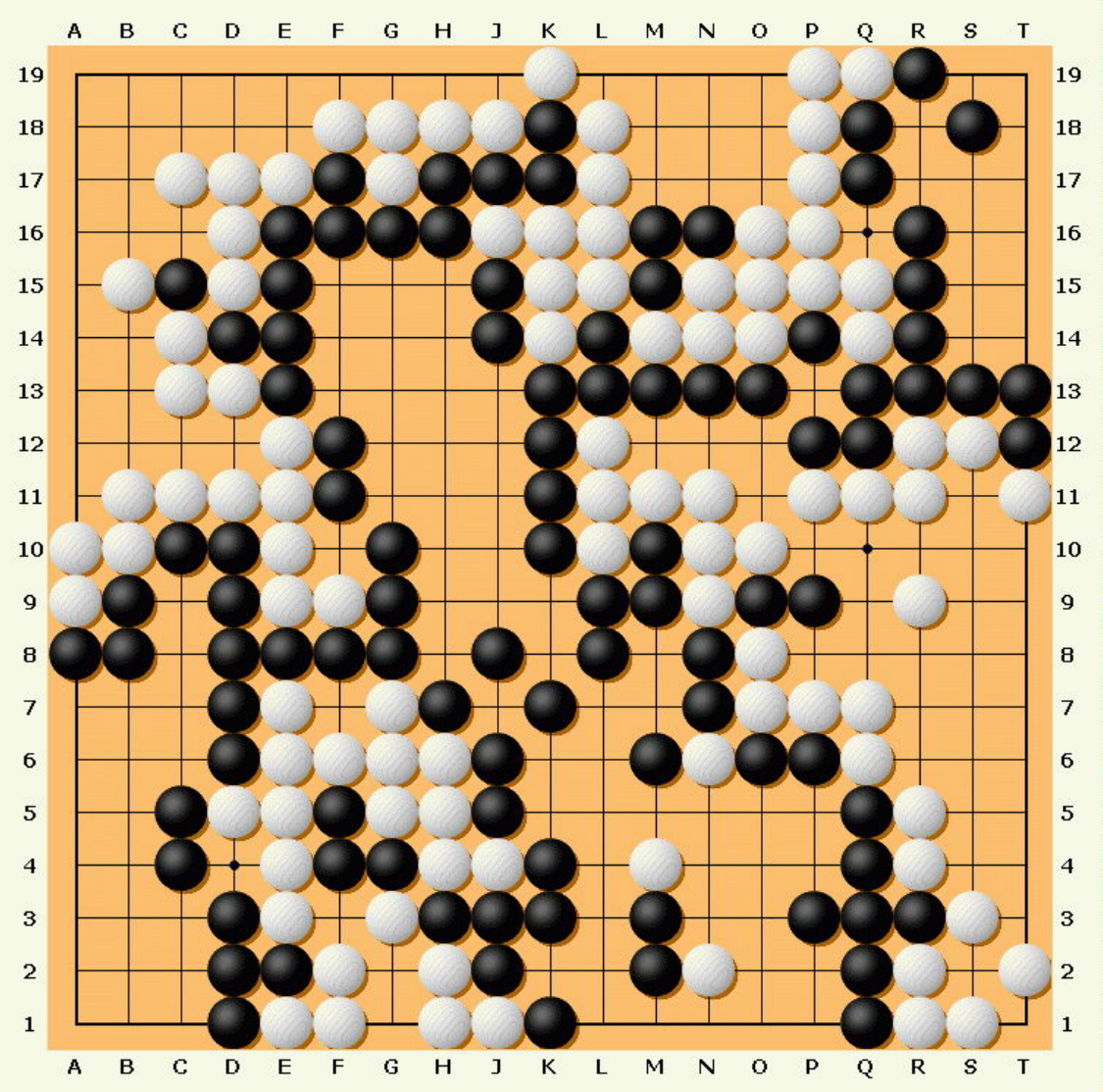
Go example

Go example

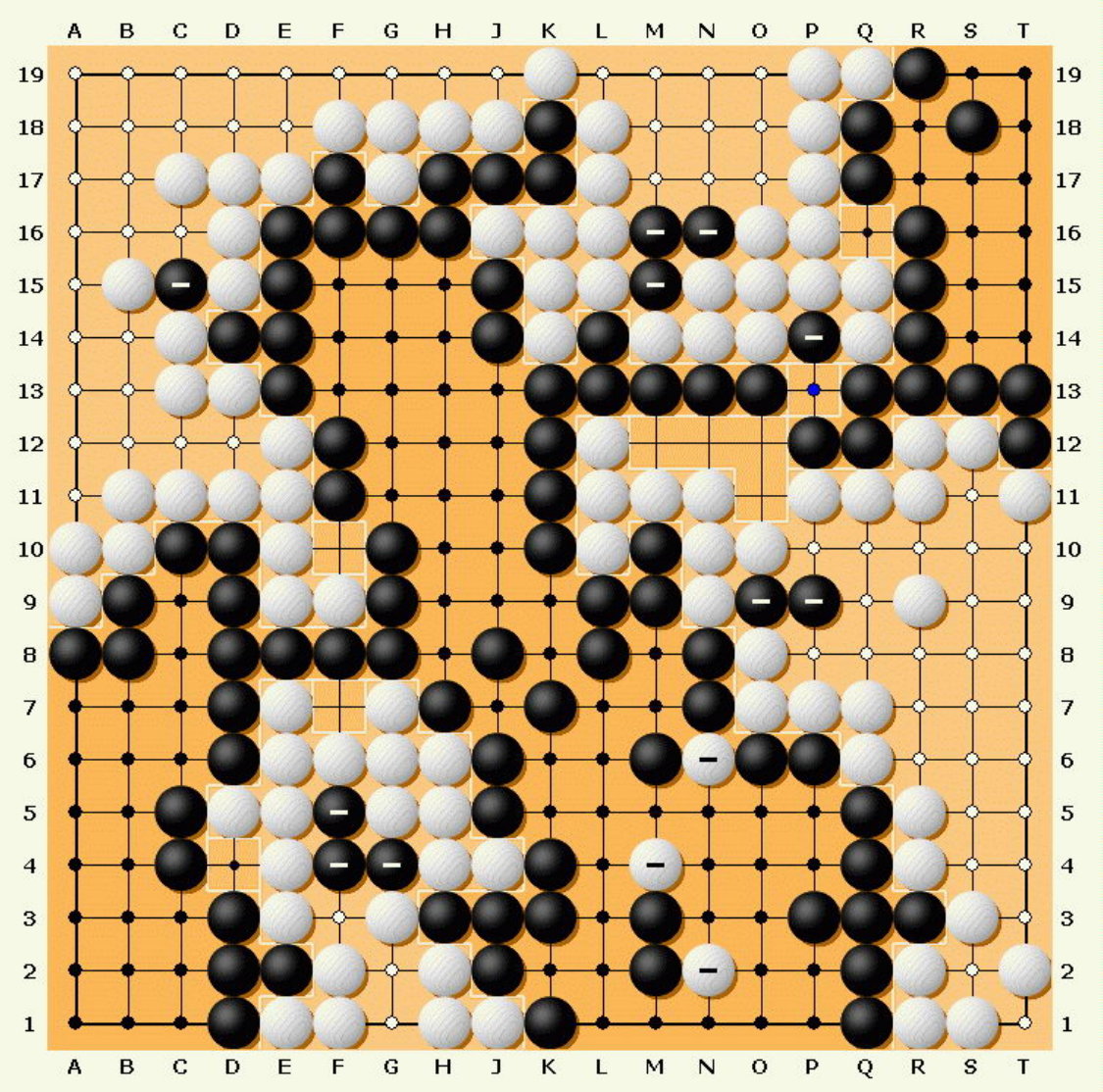
Go example

Go example

Go example



Go example



Why is Go interesting for Computer Research?

- Large state space
  - Branching factor of about 200
  - 10^{172} legal board positions
  - 10^{761} game tree nodes
- Position evaluation is difficult
- Best programs are much weaker than humans

Why is Go interesting for Computer Research?

- Large state space
  - Branching factor of about 200
  - 10^{172} legal board positions
  - 10^{360} game tree nodes
- Position evaluation is difficult
- Best programs are much weaker than humans

Why is Go interesting for Computer Research?

- Large state space
  - Branching factor of about 200
  - 10^{172} legal board positions
  - 10^{360} game tree nodes
- Position evaluation is difficult
- Best programs are much weaker than humans

Why is Go interesting for Computer Research?

- Large state space
  - Branching factor of about 200
  - 10^{172} legal board positions
  - 10^{360} game tree nodes
- Position evaluation is difficult
  - Algorithm judgements in a huge space is hard
  - Requires visual pattern recognition
  - Human knowledge is vast but hard to encode
- Best programs are much weaker than humans

Why is Go interesting for Computer Research?

- Large state space
  - Branching factor of about 200
  - 10^{172} legal board positions
  - 10^{360} game tree nodes
- Position evaluation is difficult
  - Hard to judge strength of groups statically
  - Requires visual pattern recognition
  - Human knowledge is used for [illegible]
- Best programs are much weaker than humans

Why is Go interesting for Computer Research?

- Large state space
  - Branching factor of about 200
  - 10^{172} legal board positions
  - 10^{360} game tree nodes
- Position evaluation is difficult
  - Hard to judge strength of groups statically
  - Requires visual pattern recognition
  - Stones have local as well as long-distance effects
- Best programs are much weaker than humans

Why is Go interesting for Computer Research?

- Large state space
  - Branching factor of about 200
  - 10^{172} legal board positions
  - 10^{360} game tree nodes
- Position evaluation is difficult
  - Hard to judge strength of groups statically
  - Requires visual pattern recognition
  - Stones have local as well as long-distance effects
- Best programs are much weaker than humans

Why is Go interesting for Computer Research?

- Large state space
  - Branching factor of about 200
  - 10^{172} legal board positions
  - 10^{360} game tree nodes
- Position evaluation is difficult
  - Hard to judge strength of groups statically
  - Requires visual pattern recognition
  - Stones have local as well as long-distance effects
- Best programs are much weaker than humans

Why is Go interesting for Computer Research?

- Large state space
  - Branching factor of about 200
  - 10^{172} legal board positions
  - 10^{360} game tree nodes
- Position evaluation is difficult
  - Hard to judge strength of groups statically
  - Requires visual pattern recognition
  - Stones have local as well as long-distance effects
- Best programs are much weaker than humans

Why is Go interesting for Computer Research?

- Large state space
  - Branching factor of about 200
  - 10^{172} legal board positions
  - 10^{360} game tree nodes
- Position evaluation is difficult
  - Hard to judge strength of groups statically
  - Requires visual pattern recognition
  - Stones have local as well as long-distance effects
- Best programs are much weaker than humans

My Work

- Influence Function
- Database
- Game scorer

Influence Function

- Stones radiate influence to neighboring intersections
- Crude measure of group's strength - useful if can be computed quickly
- Resistor Grid idea
  - Compute Ohm's Law
  - Current flows from high to low potential, and the current is the difference
  - Use 4 neighbors
  - Use 8 neighbors

Influence Function

- Stones radiate influence to neighboring intersections
- Crude measure of group's strength - useful if can be computed quickly
- Resistor Grid idea
  - Clamp stones: Black = +1V, White = -1V
  - Current (influence) spreads to neighbors via resistors (grid lines)
  - Get voltages at intersections

Influence Function

- Stones radiate influence to neighboring intersections
- Crude measure of group's strength - useful if can be computed quickly
- Resistor Grid idea
  - Clamp stones: Black = +1V, White = -1V
  - Current (influence) spreads to neighbors via resistors (grid lines)
  - Fast and accurate

Influence Function

- Stones radiate influence to neighboring intersections
- Crude measure of group's strength - useful if can be computed quickly
- Resistor Grid idea
  - Clamp stones: Black = +1V, White = -1V
  - Current (influence) spreads to neighbors via resistors (grid lines)
  - Fast and accurate

Influence Function

- Stones radiate influence to neighboring intersections
- Crude measure of group's strength - useful if can be computed quickly
- Resistor Grid idea
  - Clamp stones: Black = +1V, White = -1V
  - Current (influence) spreads to neighbors via resistors (grid lines)
  - Fast and accurate

Influence Function

- Stones radiate influence to neighboring intersections
- Crude measure of group's strength - useful if can be computed quickly
- Resistor Grid idea
  - Clamp stones: Black = +1V, White = -1V
  - Current (influence) spreads to neighbors via resistors (grid lines)
  - Fast and accurate

Influence example

Database

- 9 million Go games from various sources
- MySQL
- Problems
  - ALL or NOTHING!
    - Illegal moves
    - Inconsistent encoding
    - Incomplete
    - Wrong Tournament

Database

- 9 million Go games from various sources
- MySQL
- Problems
  - 70GB of raw text
  - Illegal and duplicate games
  - Inconsistently formatted files
  - Extra information

Database

- 9 million Go games from various sources
- MySQL
- Problems
  - 70GB of raw text
  - Illegal and duplicate games
  - Inconsistent game properties
  - MySQL not installed

Database

- 9 million Go games from various sources
- MySQL
- Problems
  - 70GB of raw text
  - Illegal and duplicate games
  - Inconsistent game properties
  - MySQL not installed

Database

- 9 million Go games from various sources
- MySQL
- Problems
  - 70GB of raw text
  - Illegal and duplicate games
  - Inconsistent game properties
  - MySQL not installed

Database

- 9 million Go games from various sources
- MySQL
- Problems
  - 70GB of raw text
  - Illegal and duplicate games
  - Inconsistent game properties
  - MySQL not installed

Database

- 9 million Go games from various sources
- MySQL
- Problems
  - 70GB of raw text
  - Illegal and duplicate games
  - Inconsistent game properties
  - MySQL not installed

Cooperative Scorer

- Important to determine final territory
- Key ideas:
  - Sufficient to know the list of dead stones
  - Players cooperate. Make moves that do not affect score
  - Unassumptious strategy
- Result voted from 11 simulations. Compare to GnuGo's list and original Game Record

Cooperative Scorer

- Important to determine final territory
- Key ideas:
  - Sufficient to know the list of dead stones
  - Players cooperate. Make moves that do not affect score
  - Use simple heuristics - fast!
- Result voted from 11 simulations. Compare to GnuGo's list and original Game Record

Cooperative Scorer

- Important to determine final territory
- Key ideas:
  - Sufficient to know the list of dead stones
  - Players cooperate. Make moves that do not affect score
  - Use simple heuristics - fast!
- Result voted from 11 simulations. Compare to GnuGo's list and original Game Record

Cooperative Scorer

- Important to determine final territory
- Key ideas:
  - Sufficient to know the list of dead stones
  - Players cooperate. Make moves that do not affect score
  - Use simple heuristics - fast!
- Result voted from 11 simulations. Compare to GnuGo's list and original Game Record

Cooperative Scorer

- Important to determine final territory
- Key ideas:
  - Sufficient to know the list of dead stones
  - Players cooperate. Make moves that do not affect score
  - Use simple heuristics - fast!
- Result voted from 11 simulations. Compare to GnuGo's list and original Game Record

Cooperative Scorer

- Important to determine final territory
- Key ideas:
  - Sufficient to know the list of dead stones
  - Players cooperate. Make moves that do not affect score
  - Use simple heuristics - fast!
- Result voted from 11 simulations. Compare to GnuGo's list and original Game Record

Scorer Results

- Verified on Martin Mueller's 19x19 collection of 31 games
- Tested on Erik van der Werf's 9x9 collection of 18K games
  - 96.23% agreement. Comparable to the best classifiers
- Our collection of games
  - 3,500,000 games with territory. Varying board sizes and level of completeness

	All	GnuGo	Game Record	Coop.	None
Error	3.15%	4.12%	5.66%	10.84%	76.24%

Scorer Results

- Verified on Martin Mueller's 19x19 collection of 31 games
- Tested on Erik van der Werf's 9x9 collection of 18K games
  - 96.23% agreement. Comparable to the best classifiers
- Our collection of games
  - 3,500,000 games with territory. Varying board sizes and level of completeness

	All	GnuGo	Game Record	Coop.	None
Error	3.15%	4.12%	5.66%	10.84%	76.24%

  - Cooperative: median time = 0.30 s, worst time = 5.12 s
  - GnuGo: median time = 1.70 s, worst time = 446.69 s

Scorer Results

- Verified on Martin Mueller's 19x19 collection of 31 games
- Tested on Erik van der Werf's 9x9 collection of 18K games
  - 96.23% agreement. Comparable to the best classifiers
- Our collection of games
  - 3,500,000 games with territory. Varying board sizes and level of completeness

	All	GnuGo	Game Record	Coop.	None
Error	3.15%	4.12%	5.66%	10.84%	76.24%

  - Cooperative: median time = 0.30 s, worst time = 5.12 s
  - GnuGo: median time = 1.70 s, worst time = 446.69 s

Scorer Results

- Verified on Martin Mueller's 19x19 collection of 31 games
- Tested on Erik van der Werf's 9x9 collection of 18K games
  - 96.23% agreement. Comparable to the best classifiers
- Our collection of games
  - 3,500,000 games with territory. Varying board sizes and level of completeness

	All	GnuGo	Game Record	Coop.	None
Error	3.15%	4.12%	5.66%	10.84%	76.24%

  - Cooperative: median time = 0.30 s, worst time = 5.12 s
  - GnuGo: median time = 1.70 s, worst time = 446.69 s

Future Work

- Conditional Random Fields (CRF)
  - Grid based model. General graph model
  - Predict: final territory, next move
- Cooperative Scorer
- Complete MySQL Database

Future Work

- Conditional Random Fields (CRF)
  - Grid based model. General graph model
  - Predict: final territory, next move
- Cooperative Scorer
  - Improve score
  - Compare with methods
  - Incorporate a Monte Carlo approach
- Complete MySQL Database

Future Work

- Conditional Random Fields (CRF)
  - Grid based model. General graph model
  - Predict: final territory, next move
- Cooperative Scorer
  - Improve accuracy
  - Compare to other methods
  - Incorporate into Monte Carlo Graph Search
- Complete MySQL Database

Future Work

- Conditional Random Fields (CRF)
  - Grid based model. General graph model
  - Predict: final territory, next move
- Cooperative Scorer
  - Improve accuracy
  - Compare to other methods
  - Incorporate into a Monte Carlo program
- Complete MySQL Database

Future Work

- Conditional Random Fields (CRF)
  - Grid based model. General graph model
  - Predict: final territory, next move
- Cooperative Scorer
  - Improve accuracy
  - Compare to other methods
  - Incorporate into a Monte Carlo program
- Complete MySQL Database

Future Work

- Conditional Random Fields (CRF)
  - Grid based model. General graph model
  - Predict: final territory, next move
- Cooperative Scorer
  - Improve accuracy
  - Compare to other methods
  - Incorporate into a Monte Carlo program
- Complete MySQL Database

Future Work

- Conditional Random Fields (CRF)
  - Grid based model. General graph model
  - Predict: final territory, next move
- Cooperative Scorer
  - Improve accuracy
  - Compare to other methods
  - Incorporate into a Monte Carlo program
- Complete MySQL Database

Future Work

- Conditional Random Fields (CRF)
  - Grid based model. General graph model
  - Predict: final territory, next move
- Cooperative Scorer
  - Improve accuracy
  - Compare to other methods
  - Incorporate into a Monte Carlo program
- Complete MySQL Database

Questions?

- Its time for me to GO!