State-Aware TrueSkill For Tennis Prediction

Hin Hong TAM (Ryan)
Imperial College London

September 15, 2016

GOALS

- Comprehensive Overview Of TrueSkill

GOALS

- Comprehensive Overview Of TrueSkill
- Use TrueSkill To Model Tennis

GOALS

- Comprehensive Overview Of TrueSkill
- Use TrueSkill To Model Tennis
- Forumulate And Experiment *State-Aware* TrueSkill

GOALS

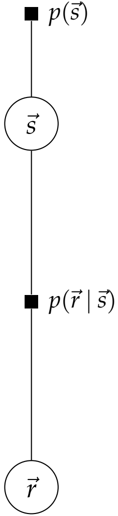
- Comprehensive Overview Of TrueSkill
- Use TrueSkill To Model Tennis
- Forumulate And Experiment *State-Aware* TrueSkill
- Use State-Aware TrueSkill To Model Tennis

FORMULATION OF TRUESKILL

- Uses Factor Graph

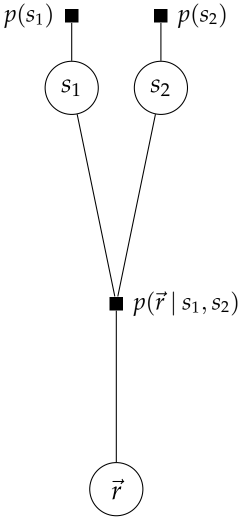
FORMULATION OF TRUESKILL

- Uses Factor Graph
- $$Pr(\vec{s}, \vec{r}) = Pr(\vec{r} \mid \vec{s}) Pr(\vec{s})$$



FORMULATION OF TRUESKILL

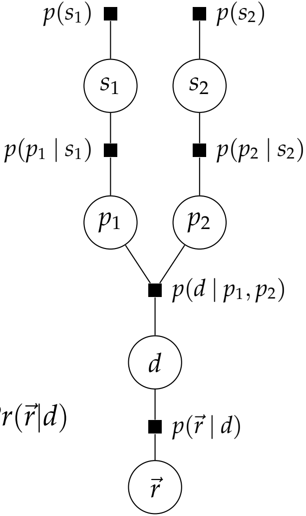
- Uses Factor Graph
- $$Pr(\vec{s}, \vec{r}) = Pr(\vec{r} \mid \vec{s}) Pr(\vec{s})$$
- Factorising Priors
$$Pr(\vec{s}) \triangleq \prod_{i=1}^{n} Pr(s_i)$$



FORMULATION OF TRUESKILL

- Uses Factor Graph
- $$Pr(\vec{s}, \vec{r}) = Pr(\vec{r} \mid \vec{s}) Pr(\vec{s})$$
- Factorising Priors
$$Pr(\vec{s}) \triangleq \prod_{i=1}^{n} Pr(s_i)$$
- Factorising Likelihood

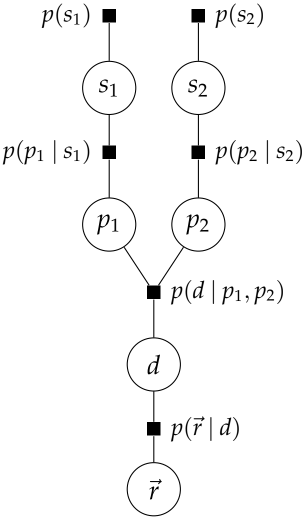
$$Pr(\vec{r}|s_1, s_2) \triangleq Pr(p_1|s_1)Pr(p_2|s_2)Pr(d|p_1, p_2)Pr(\vec{r}|d)$$



Specification Of Factors In TrueSkill

- Gaussian Skill Priors

$$p(s_i) = \mathcal{N}(s_i \mid \mu_i, \sigma_i^2 + \tau^2)$$



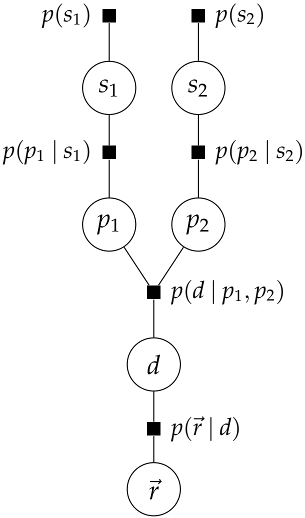
SPECIFICATION OF FACTORS IN TRUESKILL

- Gaussian Skill Priors

$$p(s_i) = \mathcal{N}(s_i \mid \mu_i, \sigma_i^2 + \tau^2)$$

- Skill-Performance Factors

$$p(p_i \mid s_i) = \mathcal{N}(p_i \mid s_i, \beta^2)$$



SPECIFICATION OF FACTORS IN TRUESKILL

- Gaussian Skill Priors

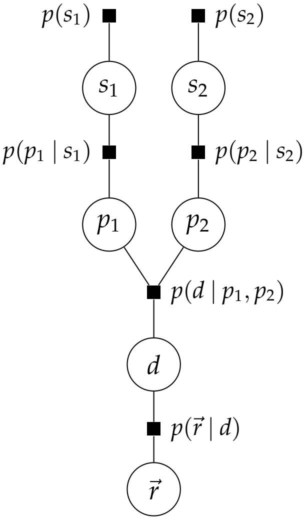
$$p(s_i) = \mathcal{N}(s_i \mid \mu_i, \sigma_i^2 + \tau^2)$$

- Skill-Performance Factors

$$p(p_i \mid s_i) = \mathcal{N}(p_i \mid s_i, \beta^2)$$

- Performance-Differencing Factor

$$p(d \mid p_1, p_2) = \mathbb{I}(d = p_1 - p_2)$$



SPECIFICATION OF FACTORS IN TRUESKILL

- Gaussian Skill Priors

$$p(s_i) = \mathcal{N}(s_i \mid \mu_i, \sigma_i^2 + \tau^2)$$

- Skill-Performance Factors

$$p(p_i \mid s_i) = \mathcal{N}(p_i \mid s_i, \beta^2)$$

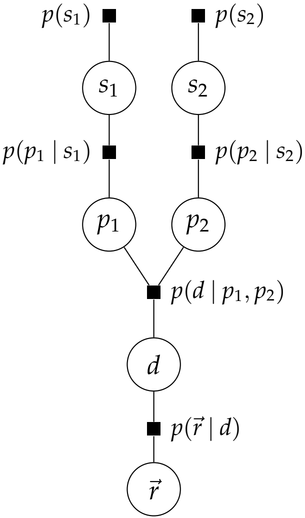
- Performance-Differencing Factor

$$p(d \mid p_1, p_2) = \mathbb{I}(d = p_1 - p_2)$$

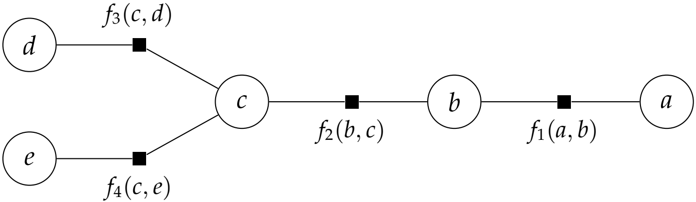
- Outcome-Truncation Factor

$p(r \mid d) = \mathbb{I}(d > 0)$ if player 1 won

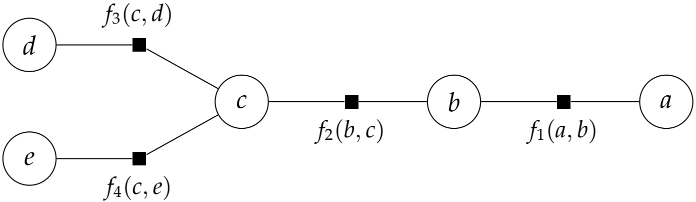
$p(r \mid d) = \mathbb{I}(d < 0)$ if player 2 won



FACTOR GRAPH EXAMPLE

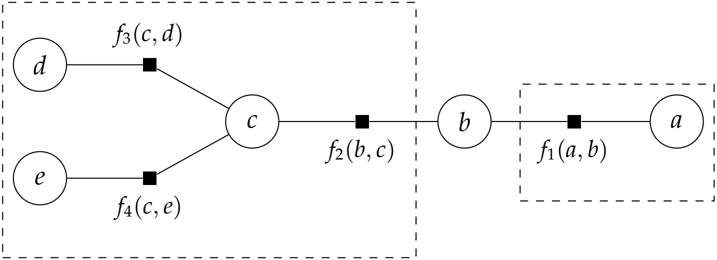


FACTOR GRAPH EXAMPLE



$$p(b) = \sum_a \sum_c \sum_d \sum_e f_1(a, b) f_2(b, c) f_3(c, d) f_4(c, e)$$

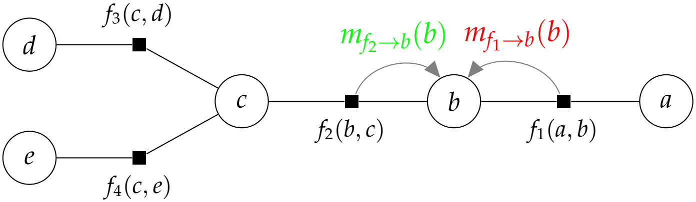
FACTOR GRAPH EXAMPLE



$$p(b) = \sum_a \sum_c \sum_d \sum_e f_1(a,b) f_2(b,c) f_3(c,d) f_4(c,e)$$

$$\implies p(b) = \sum_a f_1(a,b) \times [\sum_c \sum_d \sum_e f_2(b,c) f_3(c,d) f_4(c,e)]$$

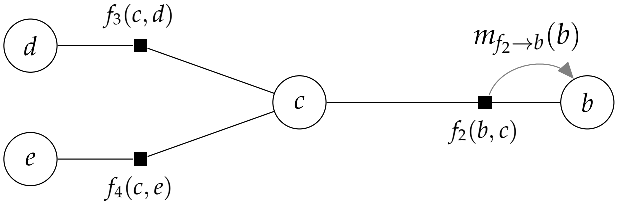
FACTOR GRAPH EXAMPLE



$$p(b) = \sum_a \sum_c \sum_d \sum_e f_1(a,b) f_2(b,c) f_3(c,d) f_4(c,e)$$

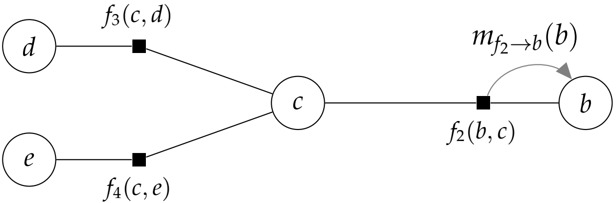
$$\implies p(b) = \sum_a f_1(a,b) \times \left[\sum_c \sum_d \sum_e f_2(b,c) f_3(c,d) f_4(c,e)\right]$$

FACTOR GRAPH



$$m_{f_2\to b}(b) = \sum_c \sum_d \sum_e f_2(b,c) f_3(c,d) f_4(c,e)$$

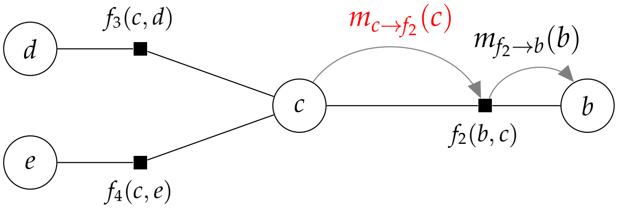
FACTOR GRAPH



$$m_{f_2 \to b}(b) = \sum_c \sum_d \sum_e f_2(b,c) f_3(c,d) f_4(c,e)$$

$$\implies m_{f_2 \to b}(b) = \sum_c [f_2(b,c)[\sum_d \sum_e f_3(c,d) f_4(c,e)]]$$

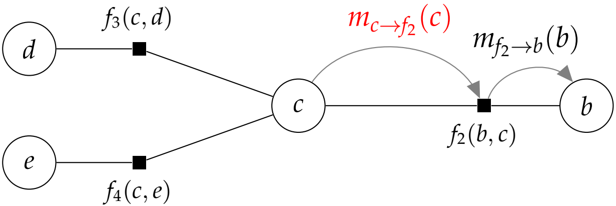
Factor Graph



$$m_{f_2\to b}(b) = \sum_c \sum_d \sum_e f_2(b,c) f_3(c,d) f_4(c,e)$$

$$\implies m_{f_2\to b}(b) = \sum_c [f_2(b,c)[\sum_d \sum_e f_3(c,d) f_4(c,e)]]$$

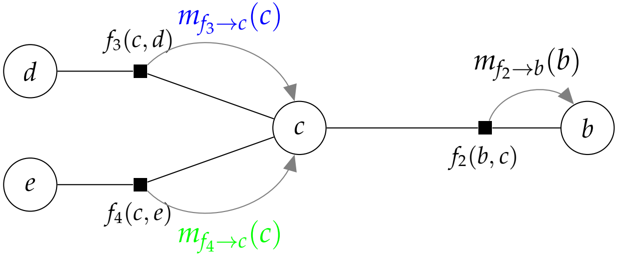
FACTOR GRAPH



$$m_{f_2\to b}(b) = \sum_c \sum_d \sum_e f_2(b,c) f_3(c,d) f_4(c,e)$$

$$\implies m_{f_2\to b}(b) = \sum_c [f_2(b,c)[\sum_d f_3(c,d)][\sum_e f_4(c,e)]]$$

FACTOR GRAPH



$$m_{f_2 \to b}(b) = \sum_c \sum_d \sum_e f_2(b,c) f_3(c,d) f_4(c,e)$$

$$\implies m_{f_2 \to b}(b) = \sum_c [f_2(b,c)[\sum_d f_3(c,d)][\sum_e f_4(c,e)]]$$

Sum-Product Algorithm

- Variable Node To Factor Node

$$m_{x_m \to f_s}(x_m) = \prod_{l \in ne(x_m) \setminus f_s} \left(m_{f_l \to x_m}(x_m) \right)$$

SUM-PRODUCT ALGORITHM

- Variable Node To Factor Node

$$m_{x_m \to f_s}(x_m) = \prod_{l \in ne(x_m) \backslash f_s} \left(m_{f_l \to x_m}(x_m) \right)$$

- Factor Node To Variable Node

$$m_{f_s \to x}(x) = \sum_{x_1} \cdots \sum_{x_M} \left(f_s(x, x_1, \ldots, x_M) \prod_{i \in ne(f_s) \backslash x} \left(m_{x_i \to f_s}(x_i) \right) \right)$$

SUM-PRODUCT ALGORITHM

- Variable Node To Factor Node

$$m_{x_m \to f_s}(x_m) = \prod_{l \in ne(x_m) \setminus f_s} \left(m_{f_l \to x_m}(x_m) \right)$$

- Factor Node To Variable Node

$$m_{f_s \to x}(x) = \sum_{x_1} \cdots \sum_{x_M} \left(f_s(x, x_1, \ldots, x_M) \prod_{i \in ne(f_s) \setminus x} \left(m_{x_i \to f_s}(x_i) \right) \right)$$

- Marginal

$$p(x) = \prod_{f_i \in ne(x)} m_{f_i \to x}(x)$$

SUM-PRODUCT ALGORITHM

- Variable Node To Factor Node

$$m_{x_m \to f_s}(x_m) = \prod_{l \in ne(x_m) \setminus f_s} \left(m_{f_l \to x_m}(x_m) \right)$$

- Factor Node To Variable Node

$$m_{f_s \to x}(x) = \sum_{x_1} \cdots \sum_{x_M} \left(f_s(x, x_1, \ldots, x_M) \prod_{i \in ne(f_s) \setminus x} \left(m_{x_i \to f_s}(x_i) \right) \right)$$

- Marginal

$$p(x) = \prod_{f_i \in ne(x)} m_{f_i \to x}(x)$$

$$\Longrightarrow \; p(x) = m_{f \to x}(x) \prod_{f_i \in ne(x) \setminus f} m_{f_i \to x}(x) \quad \forall f \in ne(x)$$

SUM-PRODUCT ALGORITHM

- Variable Node To Factor Node

$$m_{x_m \to f_s}(x_m) = \prod_{l \in ne(x_m) \backslash f_s} \left(m_{f_l \to x_m}(x_m) \right)$$

- Factor Node To Variable Node

$$m_{f_s \to x}(x) = \sum_{x_1} \cdots \sum_{x_M} \left(f_s(x, x_1, \ldots, x_M) \prod_{i \in ne(f_s) \backslash x} \left(m_{x_i \to f_s}(x_i) \right) \right)$$

- Marginal

$$p(x) = \prod_{f_i \in ne(x)} m_{f_i \to x}(x)$$

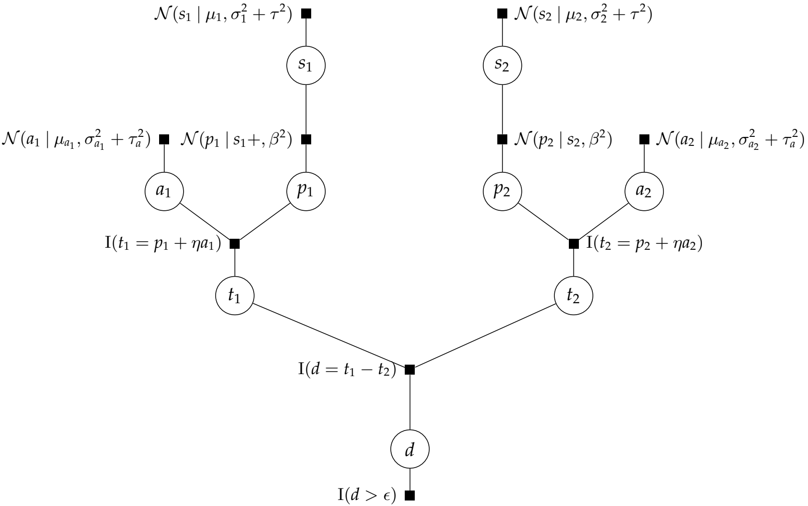
$$\implies p(x) = m_{f \to x}(x) \prod_{f_i \in ne(x) \backslash f} m_{f_i \to x}(x) \quad \forall f \in ne(x)$$

Results On Seperate Test Set

Data Granularity	Selection Based On	Brier Score	Error Rate
Match	Brier	0.199784	0.312693
Match	Error	0.202965	0.319917
Point	Brier	0.249268	0.477656
Point	Error	0.249284	0.477803

Table: Performance On A Separate Test Set Of Naïve Models

FACTOR GRAPH REPRESENTATION



Point Level Results On ATP Dataset

- Selection Of $\beta : 21 \rightarrow 10$

Point Level Results On ATP Dataset

- Selection Of β : $21 \rightarrow 10$
- Brier Score : $0.249268 \rightarrow 0.225575$

Point Level Results On ATP Dataset

- Selection Of β : $21 \rightarrow 10$
- Brier Score : $0.249268 \rightarrow 0.225575$
- Error Rate: $0.477656 \rightarrow 0.349461$

FUTURE WORK

- Extending The Model To Cover Multiplayer Games

FUTURE WORK

- Extending The Model To Cover Multiplayer Games
- Elegantly Deal With Continuous Features

FUTURE WORK

- Extending The Model To Cover Multiplayer Games
- Elegantly Deal With Continuous Features
- Include Other Features Of Tennis

FUTURE WORK

- Extending The Model To Cover Multiplayer Games
- Elegantly Deal With Continuous Features
- Include Other Features Of Tennis
- Dissociation Of Features