How to Make Best Use of Cross-Company Data in Software Effort Estimation?

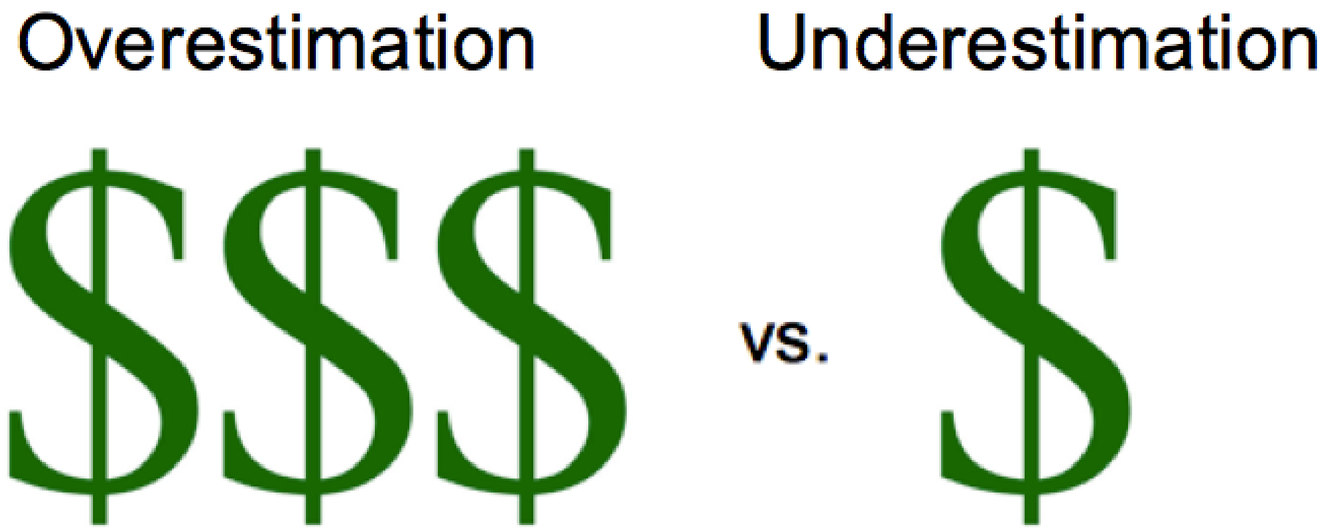
Leandro Minku, Xin Yao
{L.L.Minku, X.Yao}@cs.bham.ac.uk

CERCIA, School of Computer Science, The University of Birmingham

4th June 2014
ICSE 2014, Hyderabad, India

Software Effort Estimation (SEE):

- Estimation of the effort required to develop a software project.
- Effort is measured in person-hours, or person-months, etc.
- Based on features such as required reliability, programming language, development type, team expertise, etc.
- Main factor influencing project cost.



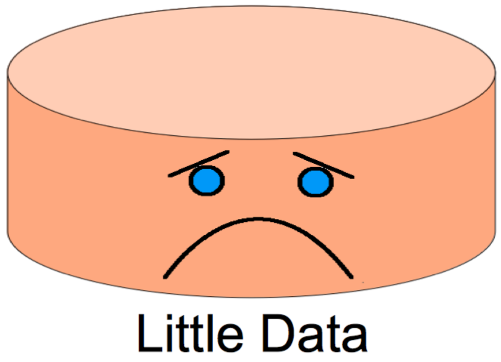
Introduction – SEE

NASA cancelled its incomplete Check-out Launch Control Software System project after the initial $200M estimate was exceeded by another $200M.

Introduction – ML for SEE

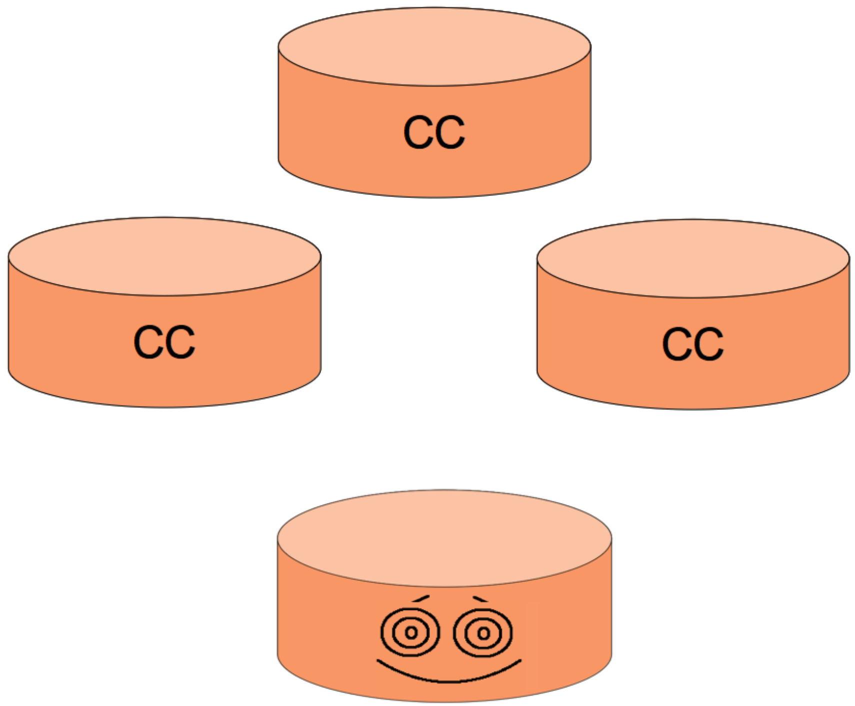
Machine learning for SEE:

- Use completed projects as training data to create SEE models.
- Problem: collecting training examples can be very costly.
- Result: low performance.



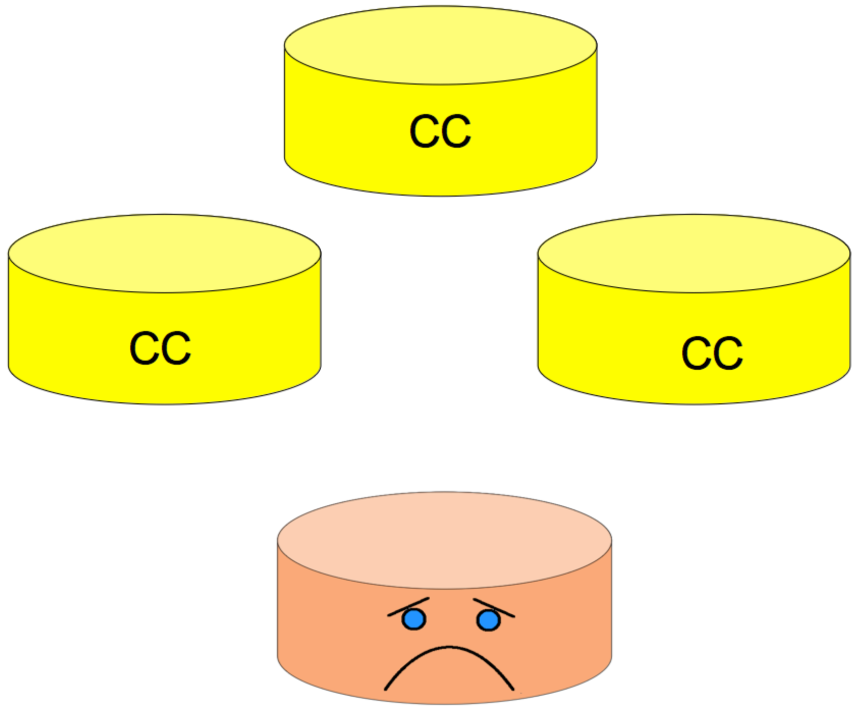
Cross-company (CC) Learning

- CC models may be used for making Within-Company (WC) predictions.
- Previous work was successful in identifying when CC models are useful.
- This can improve performance in comparison to WC models.



Cross-company (CC) Learning

- However, only when the CC context matches the WC context. Otherwise, a good amount of WC data is still necessary.
- Companies share the same context if a project described by a given set of input features requires the same effort in both companies.



Research Question

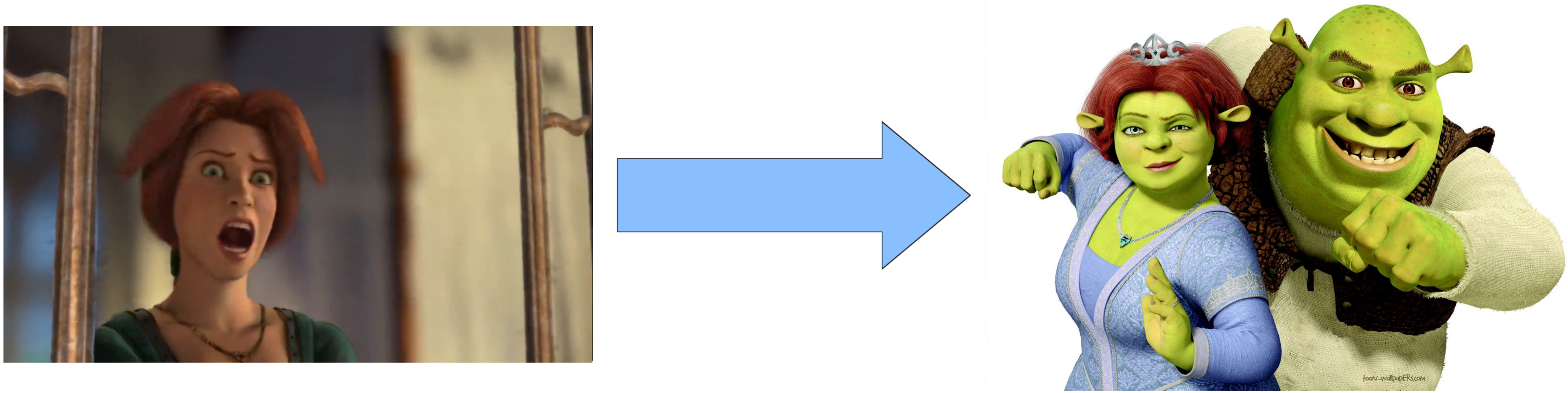
- No approach has tried to map CC data or models to the WC context in SEE.

Research Question

- No approach has tried to map CC data or models to the WC context in SEE.

Research Question

If we map CC models to the WC context, can we reduce the amount of WC training examples necessary for learning while maintaining or improving performance in comparison to a WC model?



Proposed CC SEE Learning Scenario

We consider that there is a relationship between the SEE context of a certain company and other companies:

$$f_A(\mathbf{x}) = g_{BA}(f_B(\mathbf{x}))$$

Example of simple relationship: $f_A(\mathbf{x}) = g_{BA}(f_B(\mathbf{x})) = 1.2 \cdot f_B(\mathbf{x})$

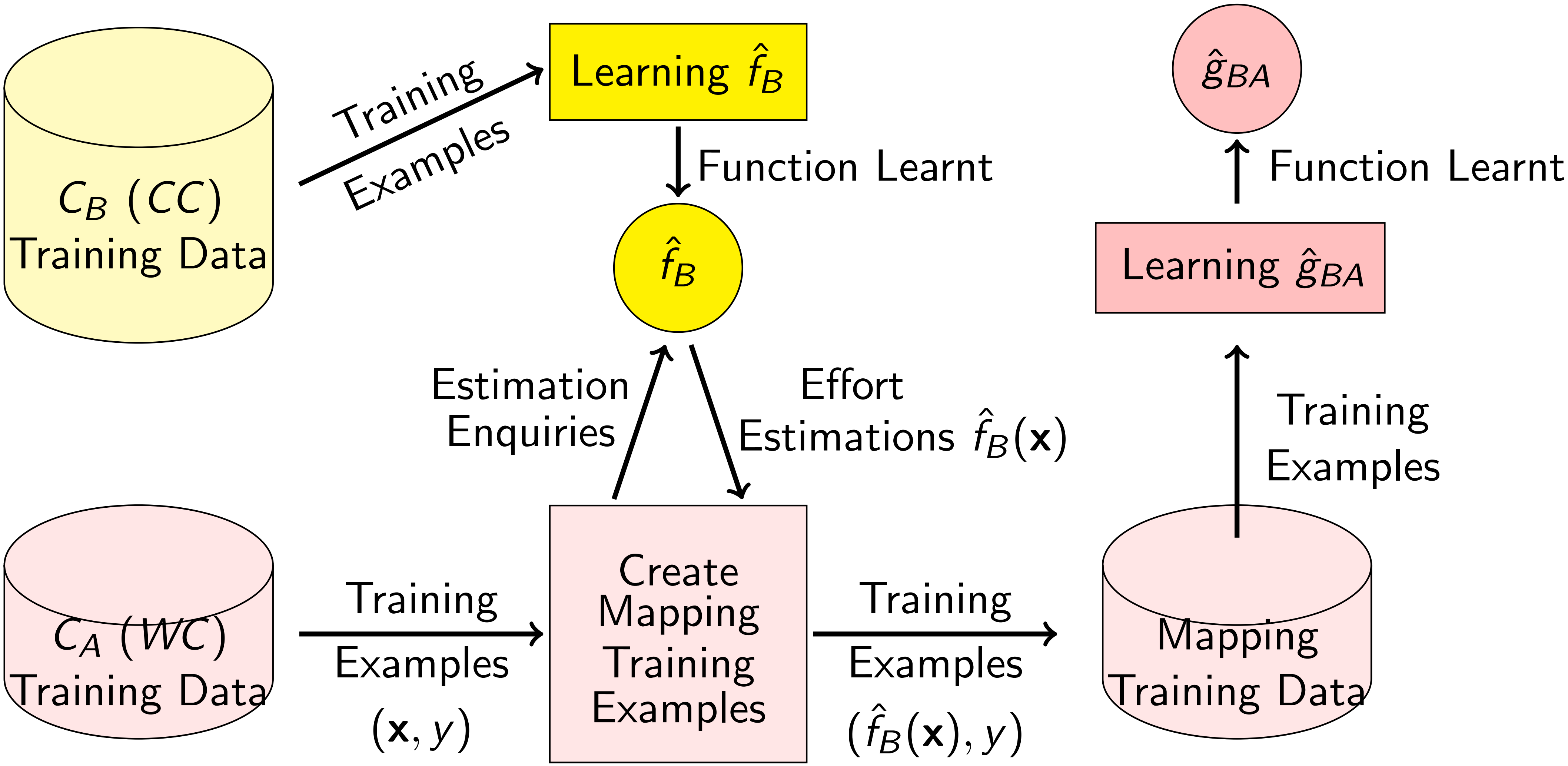
ID	Functional Size	Development	Lang.	C_B's Effort	C_A's Effort
0	100	Enhancement	3GL	500	600
1	300	Re-development	4GL	1300	1560
2	400	New Development	4GL	2000	2400
3	500	New Development	3GL	3000	3600

Learning Task

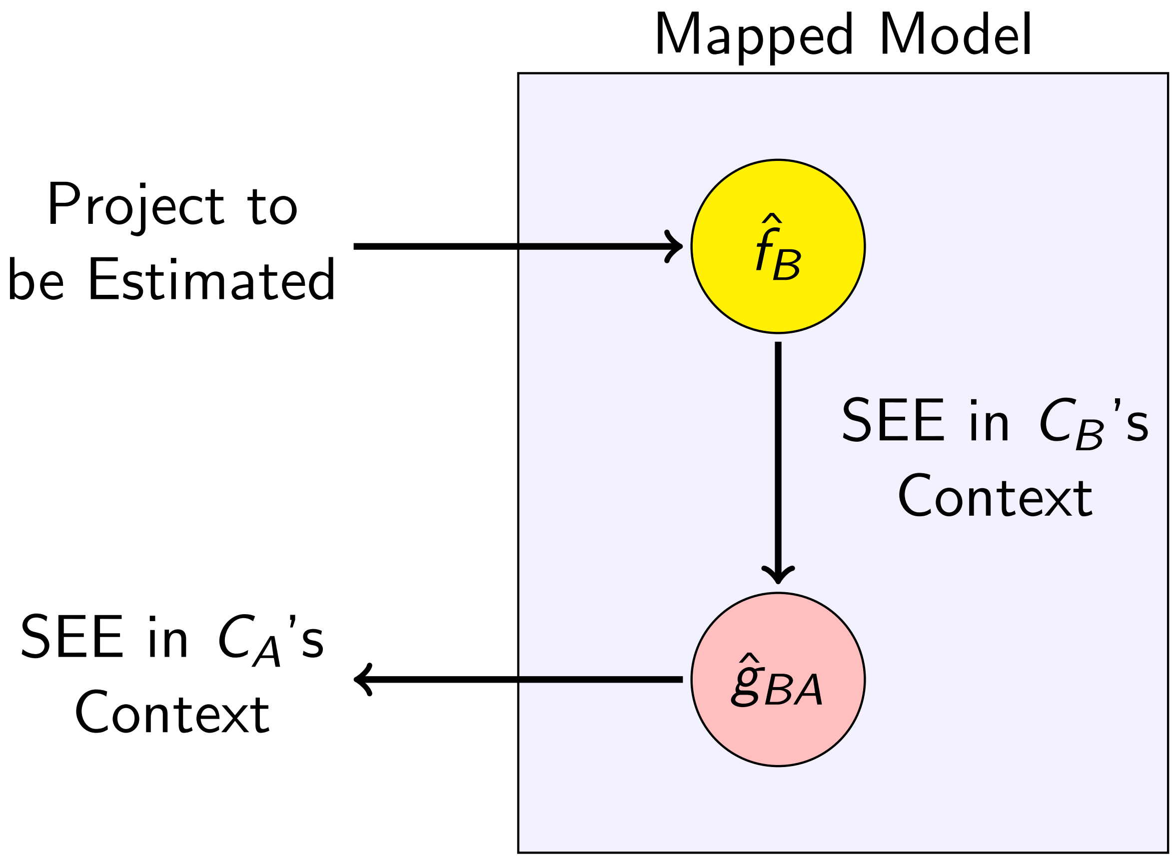
The SEE learning task involves learning the mapping functions between other companies and C_A.

Proposed CC SEE Model Mapping Framework

Mapping one CC SEE model $\hat{f}_A(\mathbf{x}) = \hat{g}_{BA}(\hat{f}_B(\mathbf{x}))$:



A Mapped Model



Online Learning

In this work, an online learning instance of the proposed framework will be presented.

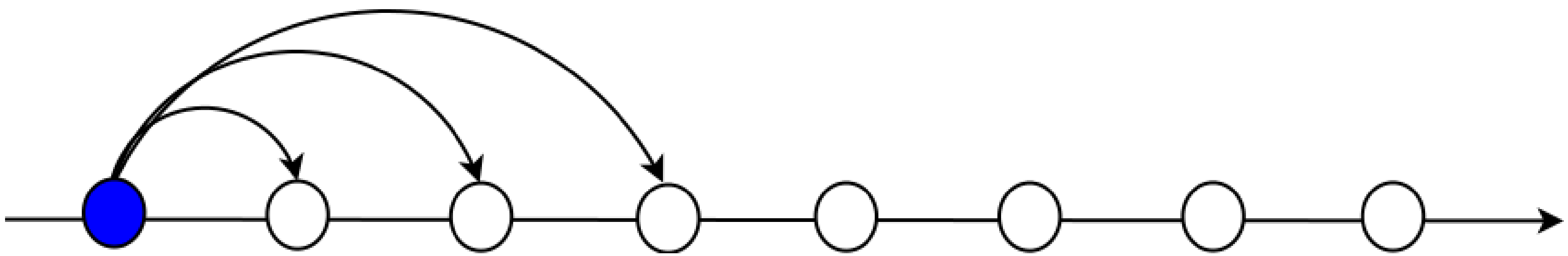
Online learning reflects more closely the real environment where SEE models operate. It allows to:

- use new completed projects to improve models,
- adapt models to changes, and
- restrict training to use only previous projects.

Online Learning Scenarios

Scenario 1:

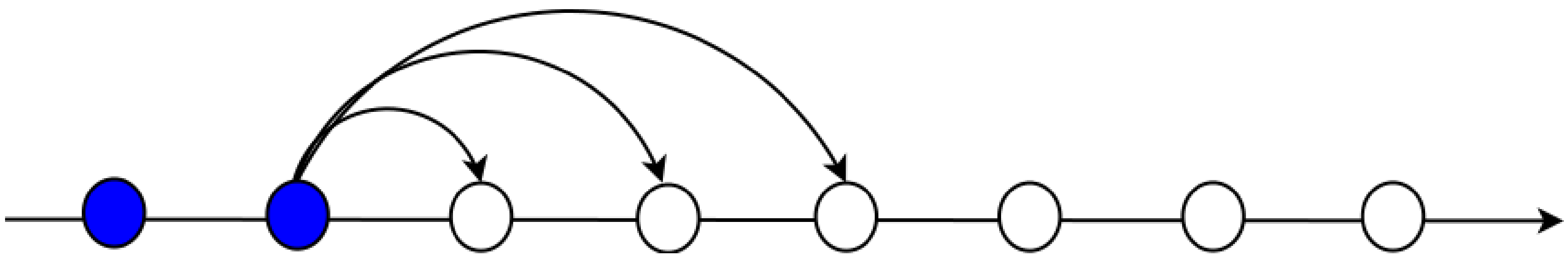
- A new WC project is completed at each time step and used for training.
- At each time step, predict the next c WC projects.



Online Learning Scenarios

Scenario 1:

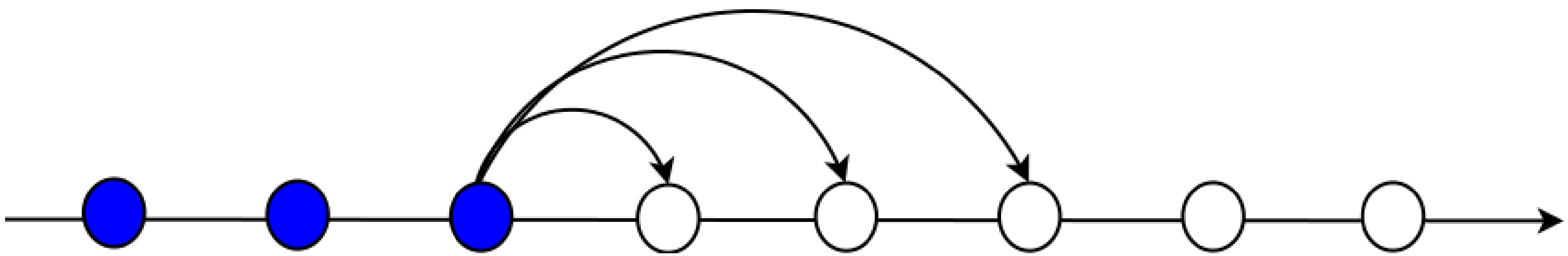
- A new WC project is completed at each time step and used for training.
- At each time step, predict the next c WC projects.



Online Learning Scenarios

Scenario 1:

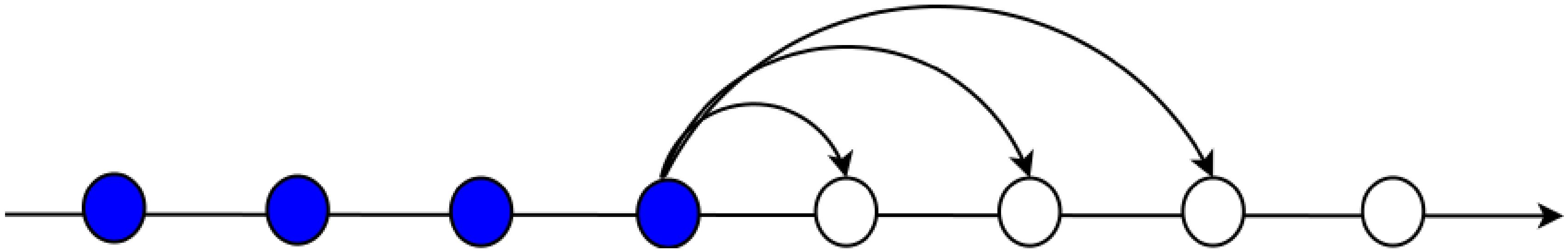
- A new WC project is completed at each time step and used for training.
- At each time step, predict the next c WC projects.



Online Learning Scenarios

Scenario 1:

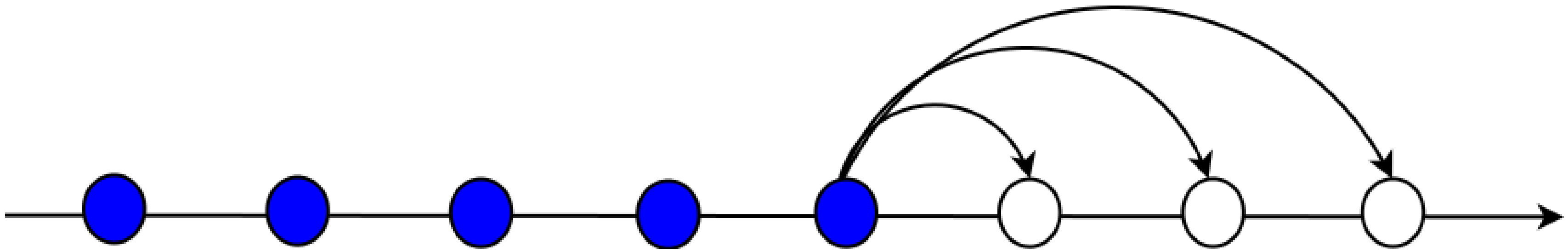
- A new WC project is completed at each time step and used for training.
- At each time step, predict the next c WC projects.



Online Learning Scenarios

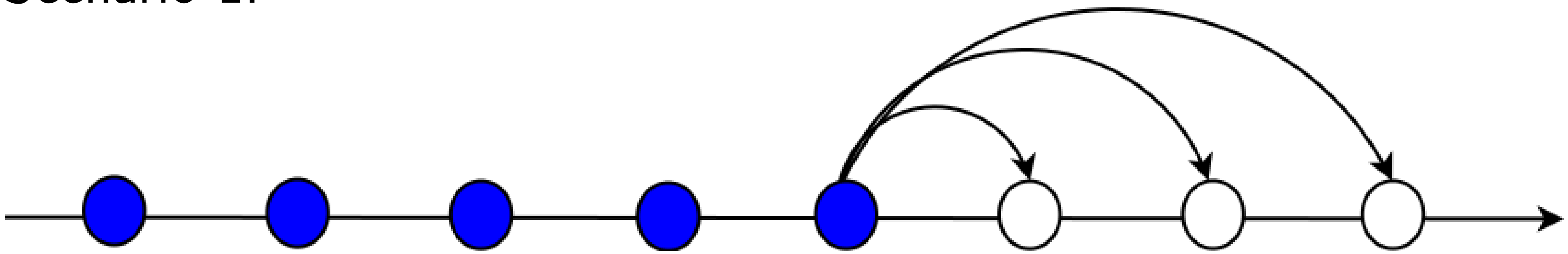
Scenario 1:

- A new WC project is completed at each time step and used for training.
- At each time step, predict the next c WC projects.



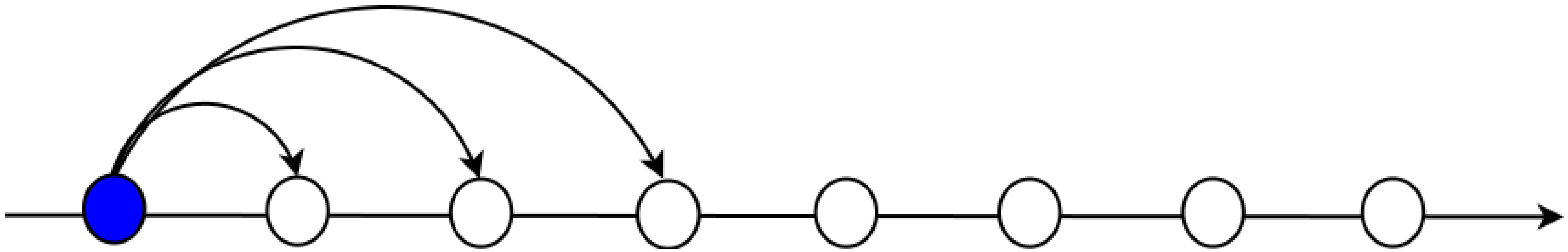
Online Learning Scenarios

Scenario 1:



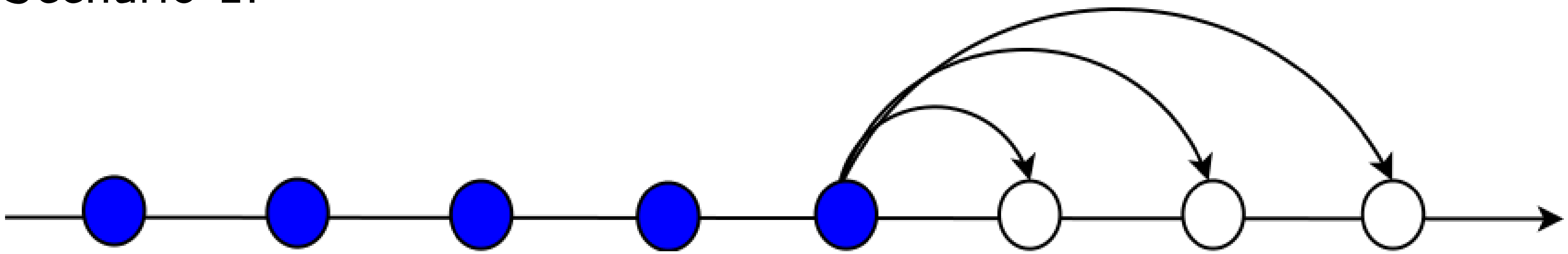
Scenario 2:

- Only WC projects completed at every p ($p > 1$) time steps are used for training.
- At each time step, predict the next c WC projects.



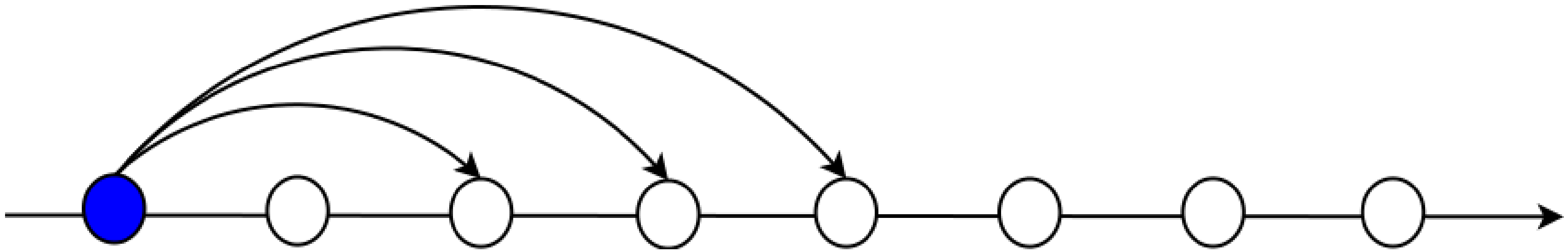
Online Learning Scenarios

Scenario 1:



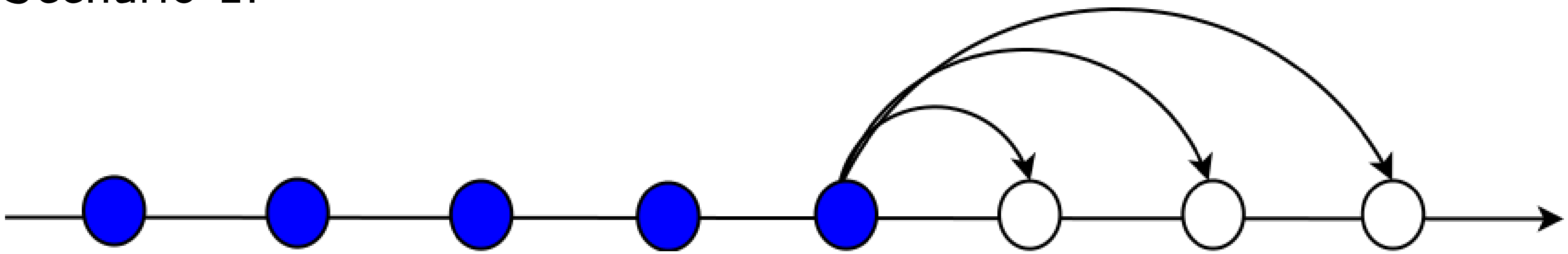
Scenario 2:

- Only WC projects completed at every p ($p > 1$) time steps are used for training.
- At each time step, predict the next c WC projects.



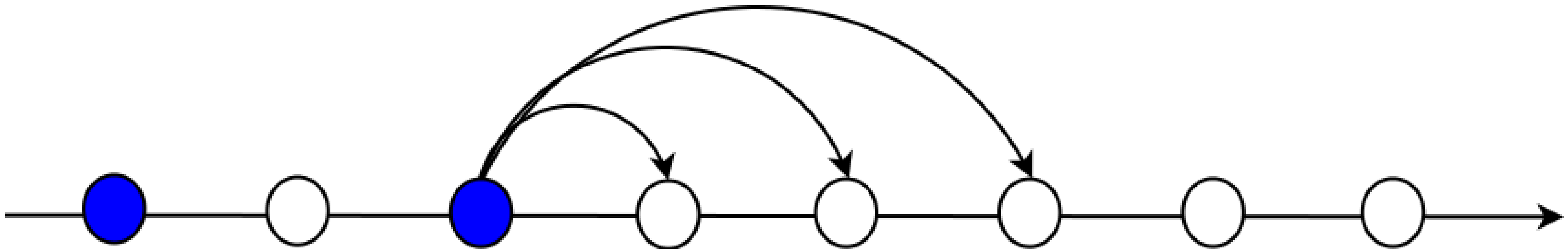
Online Learning Scenarios

Scenario 1:



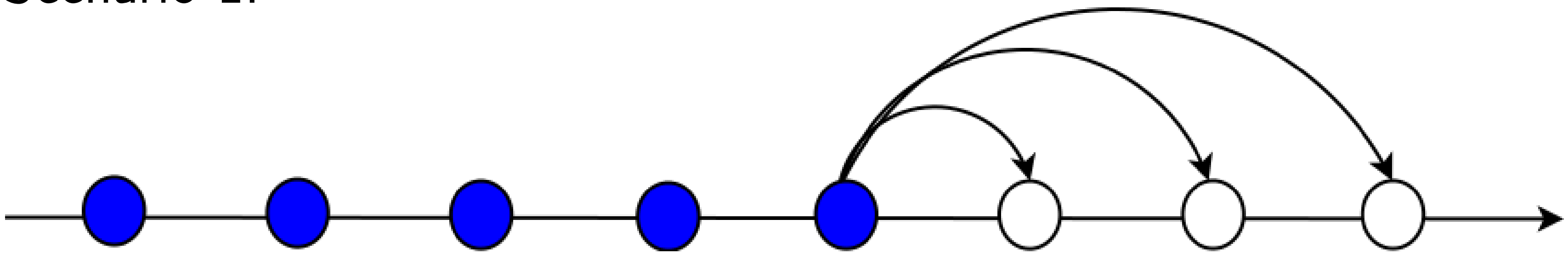
Scenario 2:

- Only WC projects completed at every p ($p > 1$) time steps are used for training.
- At each time step, predict the next c WC projects.



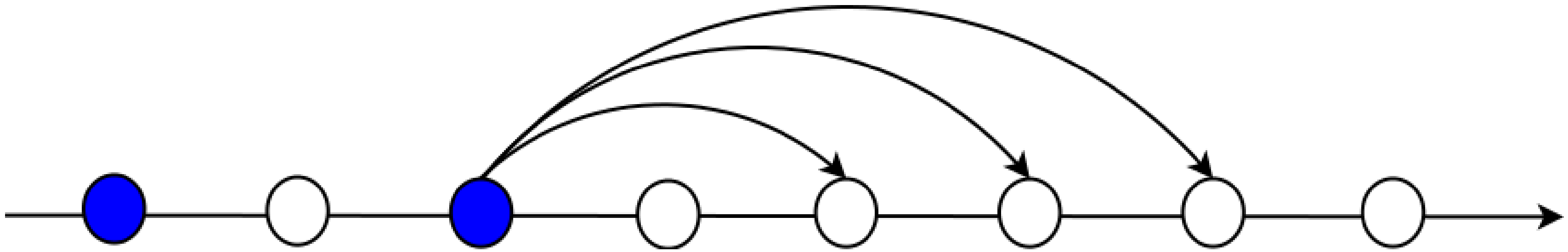
Online Learning Scenarios

Scenario 1:



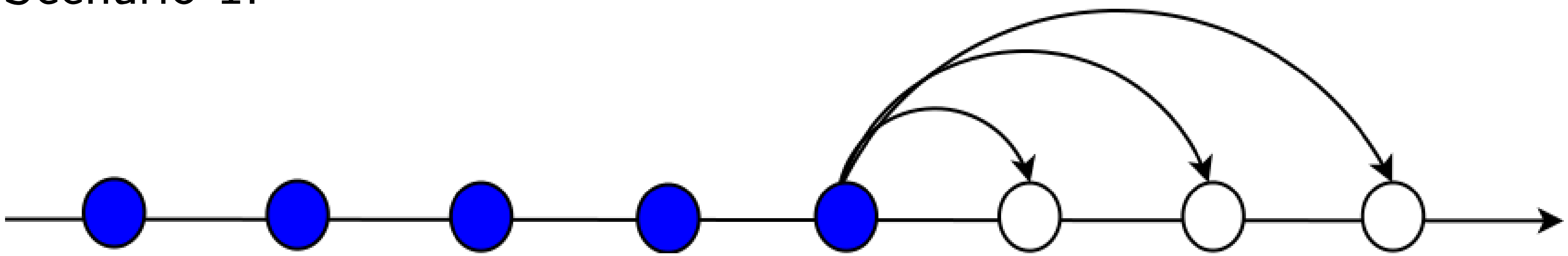
Scenario 2:

- Only WC projects completed at every p ($p > 1$) time steps are used for training.
- At each time step, predict the next c WC projects.



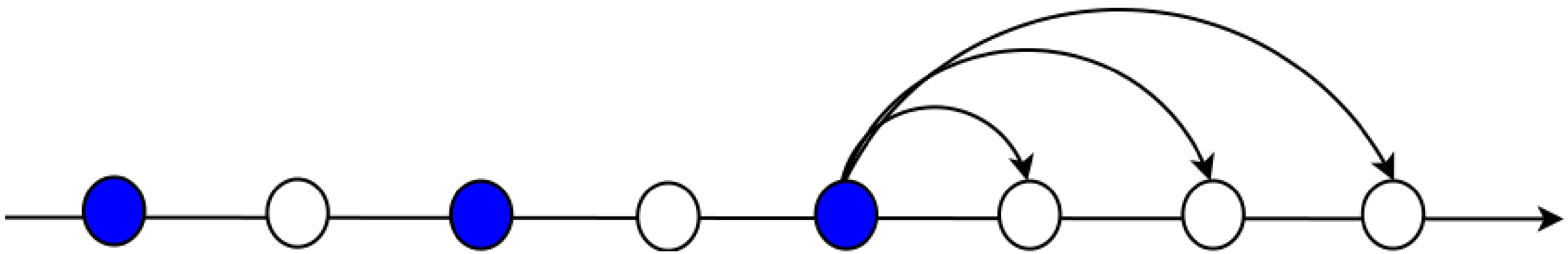
Online Learning Scenarios

Scenario 1:



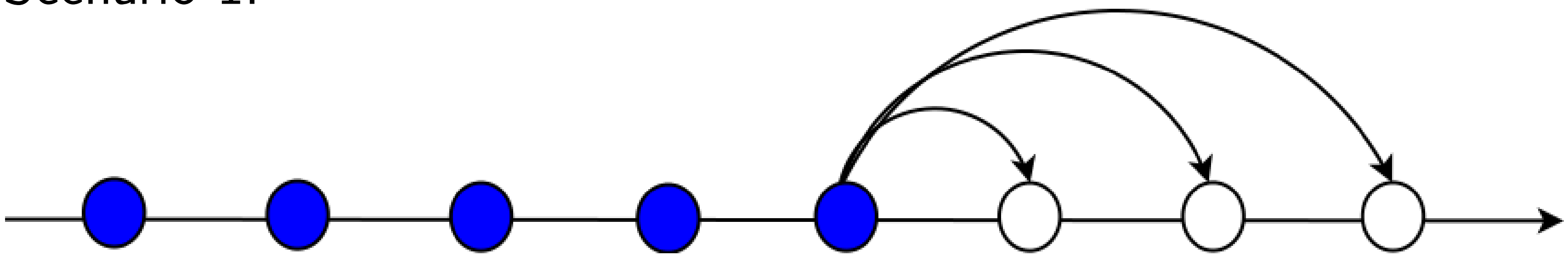
Scenario 2:

- Only WC projects completed at every p ($p > 1$) time steps are used for training.
- At each time step, predict the next c WC projects.

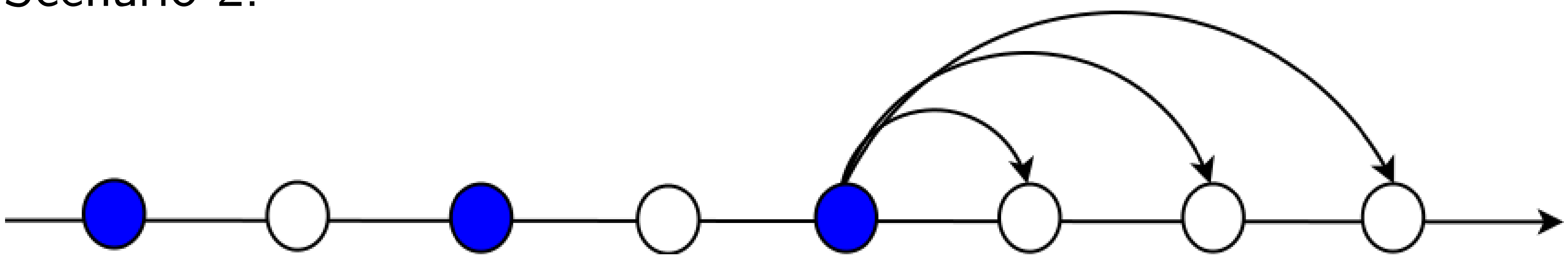


Online Learning Scenarios

Scenario 1:

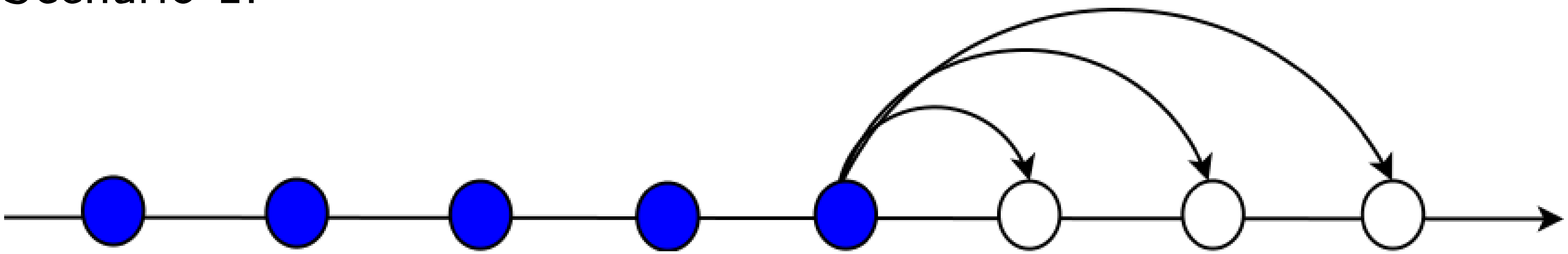


Scenario 2:

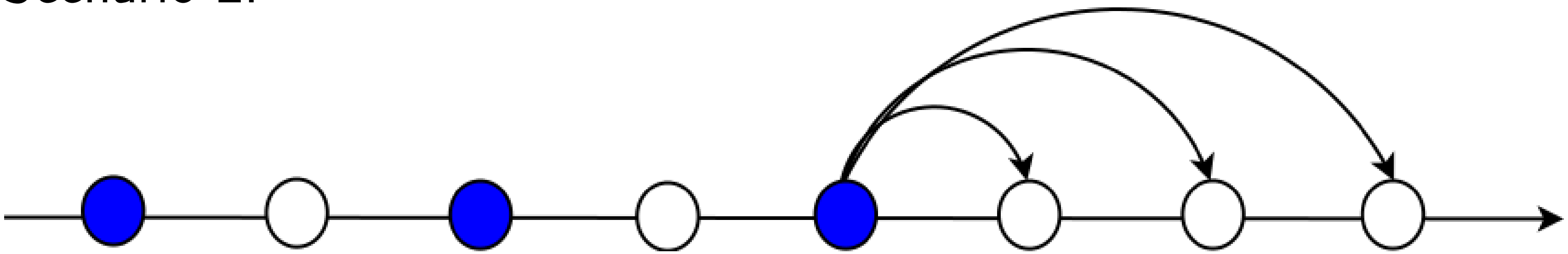


Online Learning Scenarios

Scenario 1:



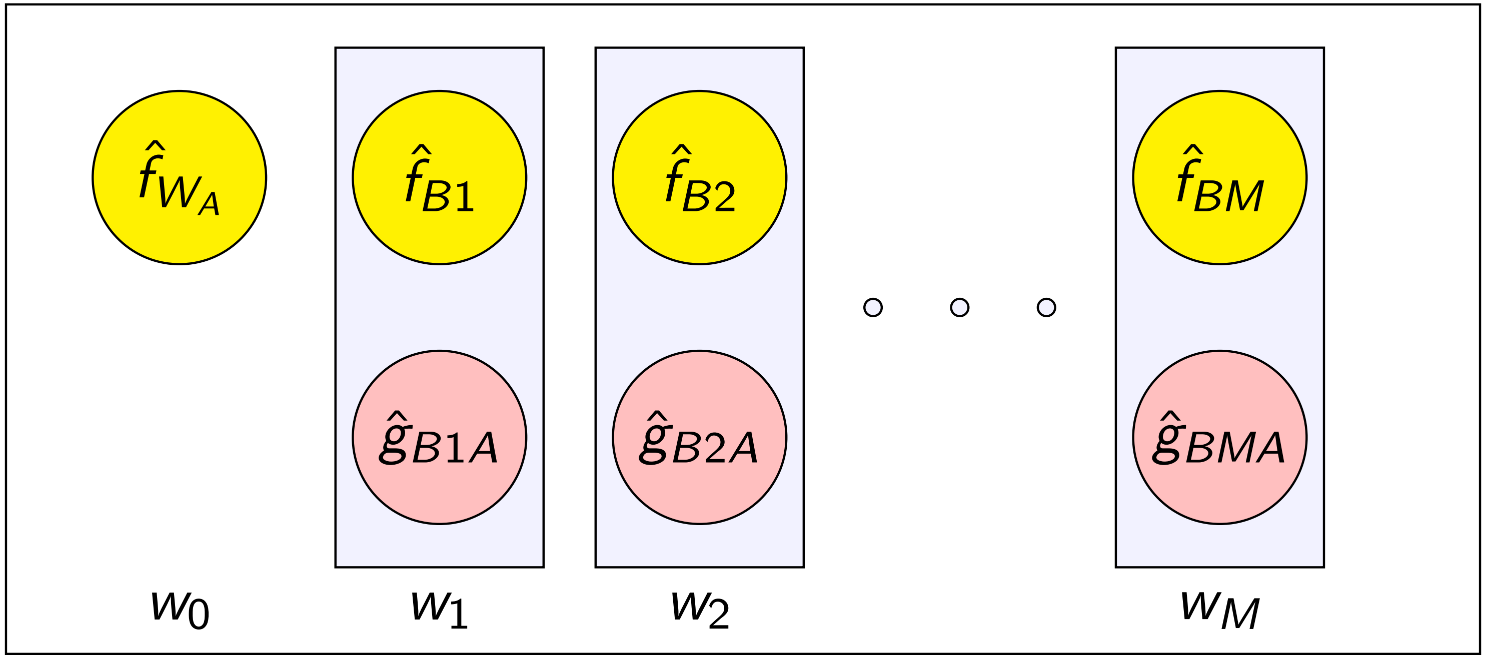
Scenario 2:



CC projects form pre-defined sets.

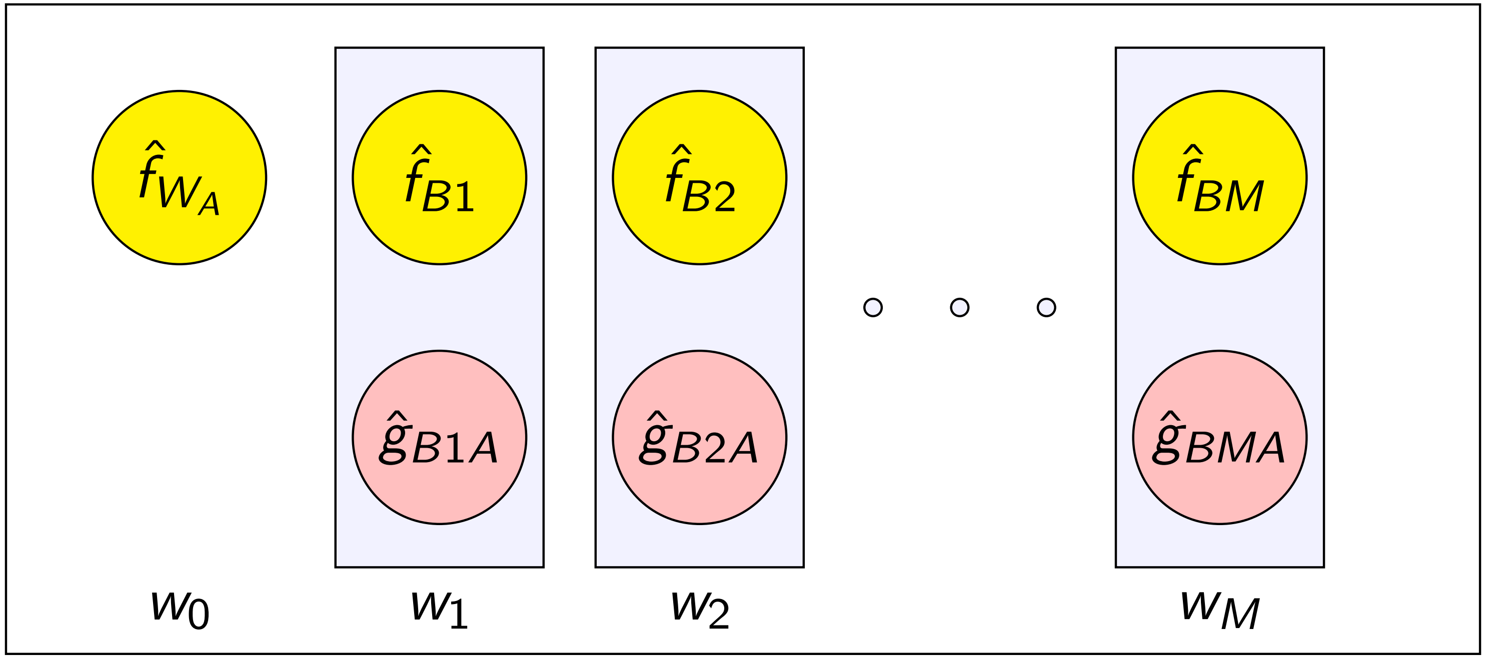
Dynamic CC Mapped Model Learning (Dycom)

Dycom – a dynamic adaptive online learning instance of the proposed framework:



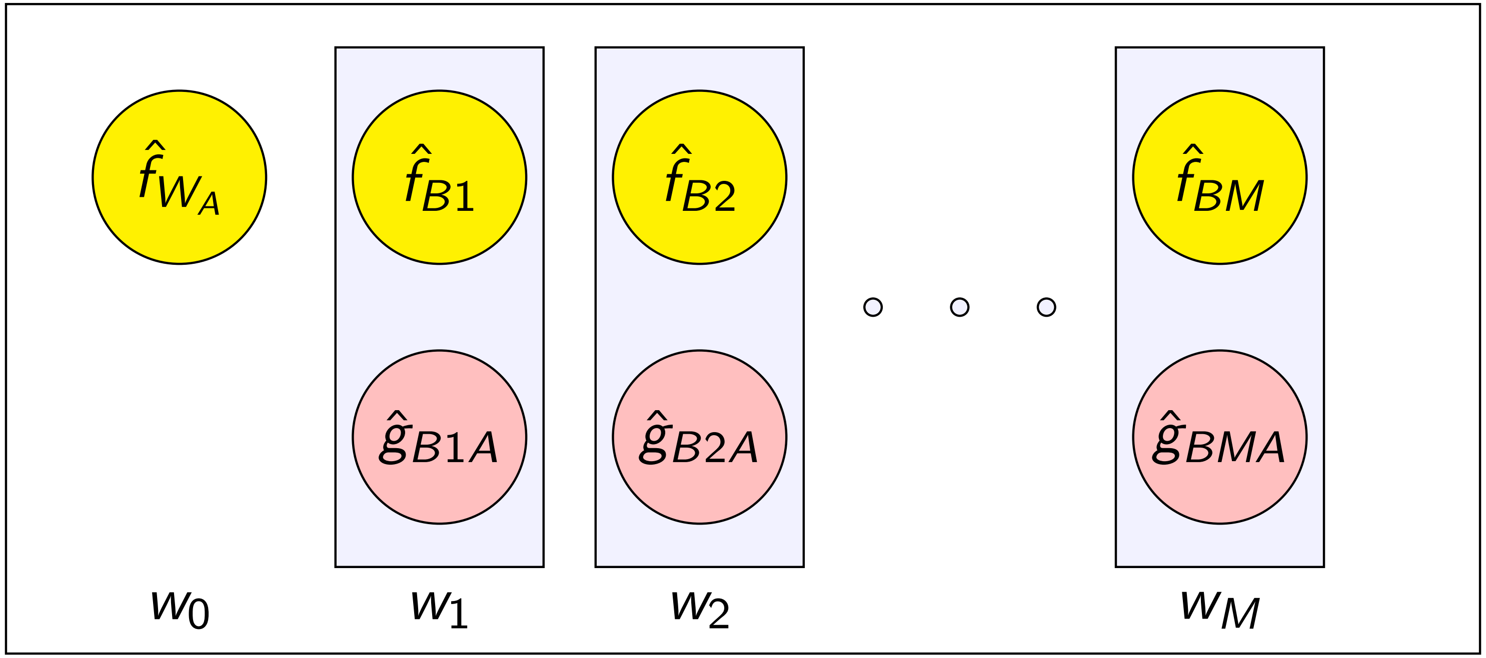
- Dycom uses ensembles of models.
- The ensemble's prediction is the weighted average of its base model's predictions.

Dynamic CC Mapped Model Learning (Dycom)



- CC training examples are available beforehand, and separated into M sections according to, e.g., productivity.
- Each CC section is used to create one offline CC model $\hat{f}_{Bi}$, $1 \leq i \leq M$.
- This allows to deal with CC data sets with mixed data from different companies and to adopt simple mapping functions.

Dynamic CC Mapped Model Learning (Dycom)



- Whenever a WC training example arrives, use it to:
  - train the WC model $\hat{f}_{W_A}$;
  - create M mapping examples, one for training each mapping function $\hat{g}_{BiA}$; and
  - update weights that represent how much we can trust each base model.

Dynamic CC Mapped Model Learning (Dycom)

Mapping functions:

- Each WC training example has the format $(\hat{f}_{Bi}(\mathbf{x}), y)$.
- Dycom assumes that $\hat{f}_A(\mathbf{x}) = \hat{g}_{BiA}(\hat{f}_{Bi}(\mathbf{x})) = \hat{f}_{Bi}(\mathbf{x}) \cdot b_i$, where b_i is a factor to be learnt.

$$b_i = \begin{cases} 1, & \text{if no mapping training example has been received yet;} \\ \dfrac{y}{\hat{f}_{Bi}(\mathbf{x})}, & \text{if } (\hat{f}_{Bi}(\mathbf{x}), y) \text{ is the first mapping training example;} \\ lr \cdot \dfrac{y}{\hat{f}_{Bi}(\mathbf{x})} + (1 - lr) \cdot b_i, & \text{otherwise.} \end{cases}$$

where lr is a smoothing factor which allows tuning the emphasis on the most recent examples.

Dynamic CC Mapped Model Learning (Dycom)

Weight update:

- At each new WC training example, weights of loser models are multiplied by β, $(0 < \beta \leq 1)$.
- So, loser models will have their weights reduced.
- Loser models are the ones who did not provide the best prediction for the current WC training example.

Dycom's Evaluation – Objective

To determine whether Dycom is able to maintain or improve performance in comparison to a corresponding WC model while using less WC training examples than this model.

Approaches compared:

- RT vs Dycom-RT.
- Dycom is set to use $p = 10$, i.e., it is trained with only 10% of the WC training examples used by RT.
- $c = 10$.

Dycom's Evaluation – Experimental Setup

Databases:

- ISBSG'2000: 119 WC, 168 CC.
- ISBSG'2001: 69 WC, 224 CC.
- ISBSG: 187 WC, 826 CC.
  - Input attributes: development type, language type, development platform and functional size.
  - Target: effort in person-hours.
- CocNasaCoc81: 60 WC Nasa projects, 63 CC projects.
  - Input attributes: 15 cost drivers and KLOC.
  - Target: effort in person-months.
- KitchenMax: 145 WC Kitchenham projects, 62 CC projects.
  - Input attributes: functional size.
  - Target: effort in person-hours.
- We are looking for more databases!!!

CC projects were divided into 3 subsets according to their productivity.

Dycom's Evaluation – Main Performance Measures

- $MAE = \frac{1}{T} \sum_{i=1}^{T} |\hat{y}_i - y_i|$;
- StdDev = standard deviation of MAE across time steps.
- $SA = \left(1 - \frac{MAE}{MAE_{rguess}}\right) \cdot 100$,
- $RMSE = \sqrt{\frac{\sum_{i=1}^{T} (\hat{y}_i - y_i)^2}{T}}$;
- $Corr = \frac{\sum_{i=1}^{T} (\hat{y}_i - \bar{\hat{y}})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{T} (\hat{y}_i - \bar{\hat{y}})^2} \sqrt{\sum_{i=1}^{T} (y_i - \bar{y})^2}}$, where $\bar{\hat{y}}$ and $\bar{y}$ are the average predicted and average actual efforts, respectively;
- $LSD = \sqrt{\frac{\sum_{i=1}^{T} \left(e_i + \frac{s^2}{2}\right)^2}{T-1}}$, where s^2 is an estimator of the variance of the residual e_i and $e_i = \ln y_i - \ln \hat{y}_i$;

Results – Overall Average Performance

Database	Approach	MAE	StdDev	SA	RMSE	Corr	LSD
KitchenMax	RT	2441.0241	2838.2375	30.1782	4850.3387	0.4350	1.2221
	Dycom-RT	2208.6522	2665.4276	36.8249	4287.4476	0.6416	0.8809
	P-value	3.82E-11	6.35E-01	–	1.46E-12	1.62E-16	4.25E-21
CocNasaCoc81	RT	319.4572	250.2325	33.1366	477.2357	0.6427	0.8623
	Dycom-RT	161.7917	105.7591	66.1365	243.6504	0.8885	0.6671
	P-value	4.04E-06	1.40E-11	–	5.95E-08	4.12E-07	8.82E-04
ISBSG2000	RT	2753.3726	1257.4586	37.0471	4133.1006	0.3554	1.4592
	Dycom-RT	2494.6639	1249.8400	42.9622	3741.8009	0.4515	1.1589
	P-value	4.72E-02	1.01E-01	–	1.83E-01	8.73E-02	1.27E-06
ISBSG2001	RT	3621.9598	1367.9603	11.9270	5149.6267	0.1658	1.8110
	Dycom-RT	2543.9495	1165.8591	38.1403	3581.6573	0.5691	1.2447
	P-value	3.21E-06	4.16E-01	–	7.88E-06	2.29E-10	6.24E-08
ISBSG	RT	3253.9349	2476.0512	46.2891	4872.9193	0.4412	1.3475
	Dycom-RT	3122.6603	2227.9812	48.4560	4473.6527	0.5817	1.0378
	P-value	5.56E-02	3.54E-01	–	4.18E-02	1.90E-09	2.99E-12

- Dycom's MAE (and SA), StdDev, RMSE, Corr and LSD were always similar or better than RT's (Wilcoxon tests with Holm-Bonferroni corrections).
- So, Dycom was successful in achieving similar or better performance while using much less WC training data.

Research Question Revisited

Research Question

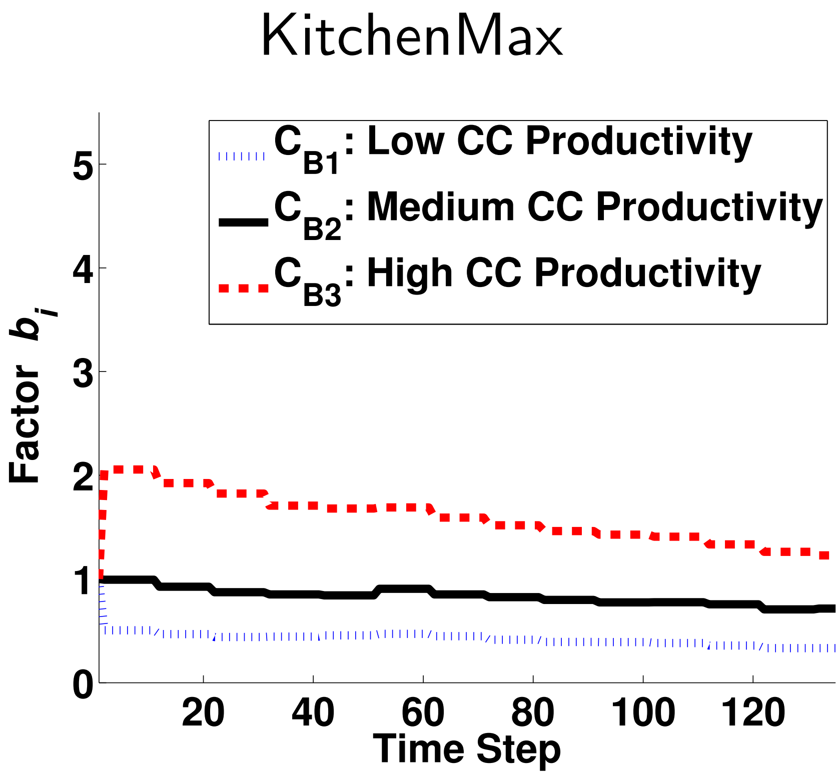
If we map CC models to the WC context, can we reduce the amount of WC training examples necessary for learning while maintaining or improving performance in comparison to a WC model?

- Yes. Overall, Dycom is an example of approach that achieves that.

Insight Provided by Dycom

- Mapping functions learnt by Dycom explain the relationship between the effort of different companies.
- The factor b_i can be plotted to visualise that.
- It can show the need for strategic decision making towards improvement of productivity.
- It can be used to monitor the success of strategies being adopted.

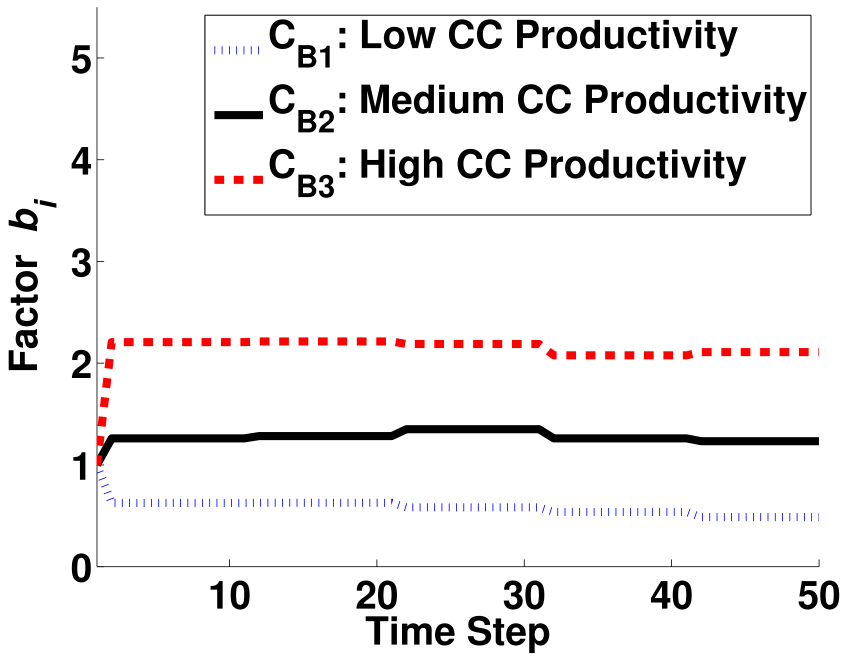
Insight Provided by Dycom



$$\hat{f}_A(\mathbf{x}) = \hat{f}_{Bi}(\mathbf{x}) \cdot b_i$$

- The company needs more/less effort than the high/low productivity CC section.
- In the beginning, the company requires twice the effort of the high productivity CC section.
- In the end, this improves to 1.2 times.

CocNasaCoc81



$$\hat{f}_A(\mathbf{x}) = \hat{f}_{Bi}(\mathbf{x}) \cdot b_i$$

- This company does not improve much with time.
- It needs more effort than the medium productivity CC section.
- Examples from the medium productivity CC section can be used to decide on strategies to improve productivity.

Deciding on Strategies for Improving Productivity

Number of projects with each feature value for the 20 CC projects from the medium productivity CC section and the first 20 WC projects:

Feature / Value	Lang. exp		Virtual mach. exp	
	CC	WC	CC	WC
Very low	1	0	1	0
Low	1	0	4	4
Nominal	8	8	8	16
High	10	12	7	0
Very high	0	0	0	0
Extremely high	0	0	0	0

- Both the company and the medium CC section frequently use employees with high programming language experience.

Deciding on Strategies for Improving Productivity

Number of projects with each feature value for the 20 CC projects from the medium productivity CC section and the first 20 WC projects:

Feature / Value	Lang. exp		Virtual mach. exp	
	CC	WC	CC	WC
Very low	1	0	1	0
Low	1	0	4	4
Nominal	8	8	8	16
High	10	12	7	0
Very high	0	0	0	0
Extremely high	0	0	0	0

- Both the company and the medium CC section frequently use employees with high programming language experience.
- Medium CC section uses more employees with high virtual machine experience. So, this is more likely to be a problem for the company. Sensitivity analysis and project manager knowledge could help to confirm that.

Conclusions

- For the first time, a CC SEE learning scenario was introduced to consider the relationship between the WC and the CC required effort.
- A framework for learning this relationship has been proposed and designed to make best use of CC data in SEE.
- A dynamic adaptive instance of this framework (Dycom) has been proposed and evaluated.
- Dycom was successful in mapping CC models to the WC context, being able to achieve similar or better performance than a corresponding WC model while using a tenth of the WC training examples.
- The learned mapped model can give insights into the behaviour of a company in comparison to others and facilitate strategic decision making towards improvement of productivity.

Acknowledgement

This work was funded by the EPSRC project Dynamic Adaptive Automated Software Engineering (DAASE).

Thank you!