

Early Failure Prediction in Feature Request Management Systems

**Camilo
Fitzgerald**

Emmanuel
Letier

Anthony
Finkelstein

UCL Computer Science

31st August 2011

Feature Request Forums

- Web-based requirements elicitation, analysis and management
- Used to decide whether to develop or reject features suggested by stakeholders:
 - Stakeholders' needs not understood
 - Poor requirements management and decisions

...Process and Product Failures

Upfront Analysis Problem

How much requirements analysis should be devoted to a feature before deciding to reject or implement it?

Too little - More process and product failures

Too much - Wasted time and effort

Approach

- Study failures traceable to early stage requirements decisions
- Generate failure prediction models from historical project data
- Cost-model for estimating the value gained from actioning a set of failure predictions

Overview

Feature Request Failures

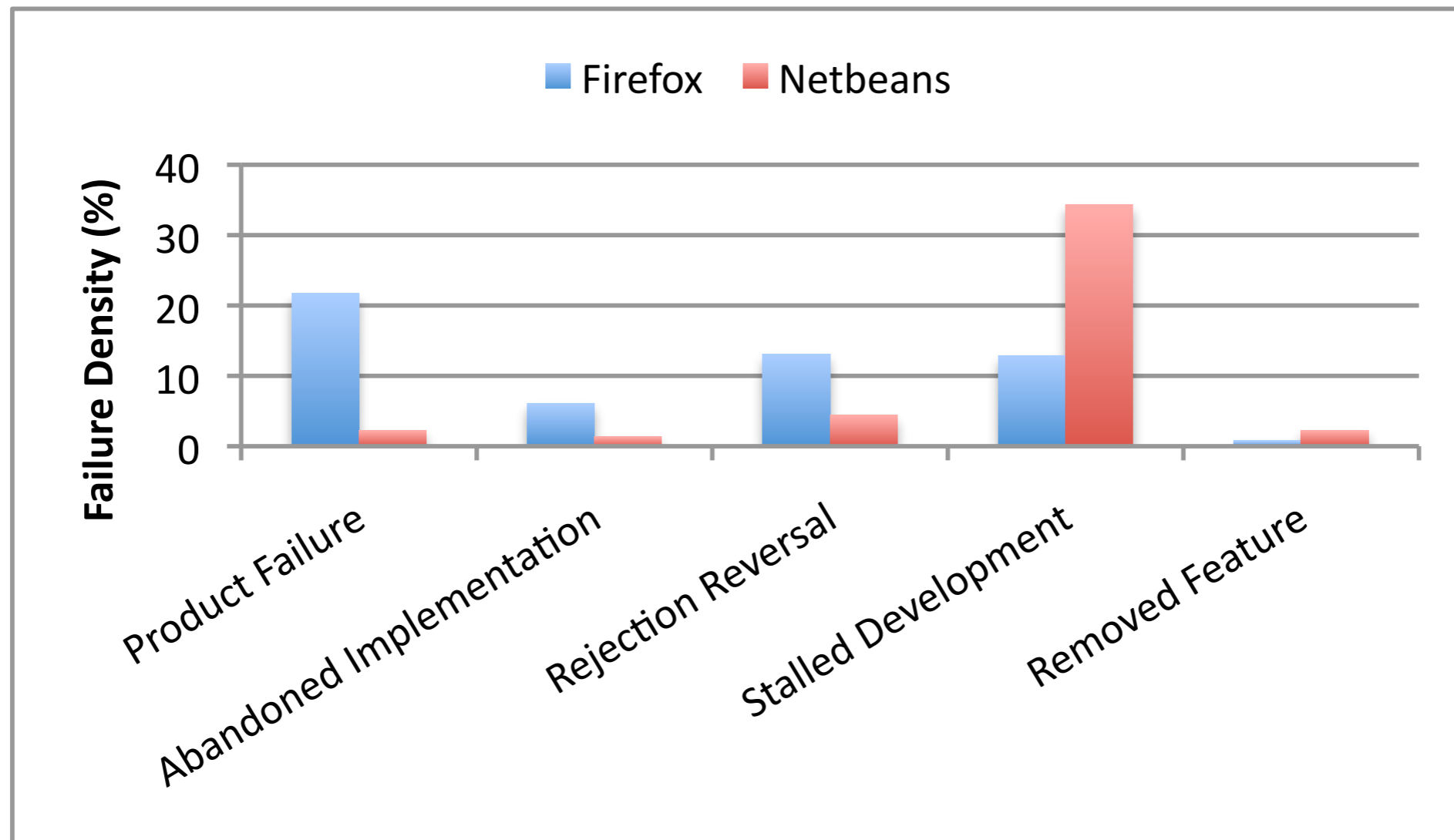
Value of Failure Predictions

Experiments and Results

Feature Request Failures

Product Failure	Bug reported on feature after product integration
Abandoned Implementation	Rejected after code development
Rejection Reversal	Integrated into product after prior rejection
Stalled Development	Assigned for development without progress for a year
Removed Feature	Rejected after product integration

Failures in Firefox and Netbeans



Firefox: 6,397 features / 8.3 years

Netbeans: 24,167 features / 10.3 years

Research Questions

1. Is it possible to make early failures predictions reliable enough to be of value to projects?
2. What attributes of early feature requests discussions can be used as reliable predictors?

Overview

Feature Request Failures

Value of Failure Predictions

Experiments and Results

Valuing Failure Predictions

- A set of failure predictions will have imperfect precision and recall
- A *cost-benefit model* estimates the value gained from actioning predictions

Precision: $\frac{\text{Correct Failure Predictions}}{\text{Predicted Failures}}$

Recall: $\frac{\text{Correct Failure Predictions}}{\text{Total Failures}}$

Expected Value of Predictions

$$ExpectedValue = recall.(\alpha - \frac{1}{precision})$$

$$\alpha = P_s \cdot \frac{C_f}{C_a}$$

$\frac{C_f}{C_a}$ - Ratio of cost of action to cost of failure

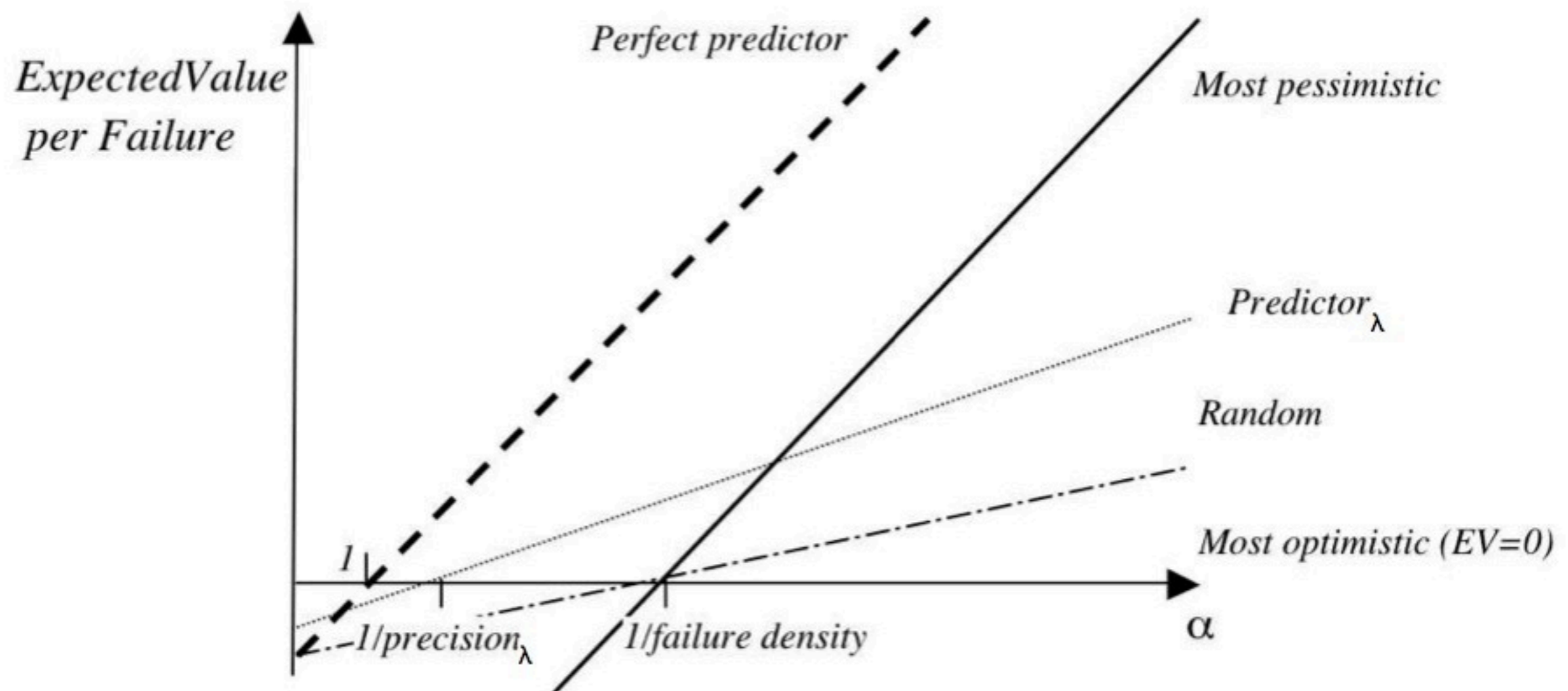
P_s - Probability that an action to prevent a failure will succeed

eg. Product Failures in the Firefox project: $P_s \cdot \frac{C_f}{C_a} = \alpha = 0.3 * 10 = \underline{3}$

Expected Value Baselines

$$\text{Expected Value} = \text{recall} \cdot \left(\alpha - \frac{1}{\text{precision}} \right)$$

$$\alpha = P_s \cdot \frac{C_f}{C_a}$$



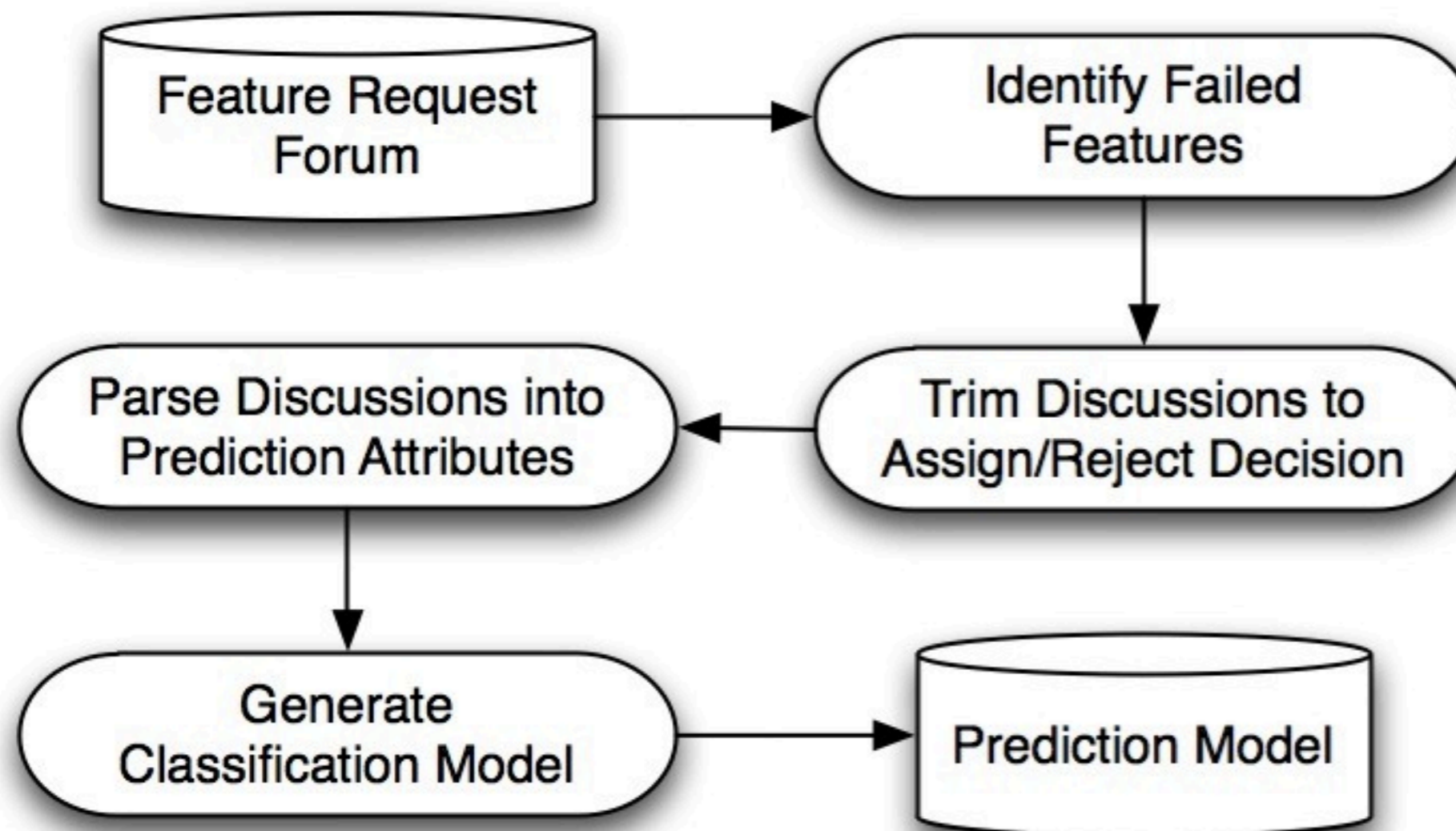
Overview

Feature Request Failures

Value of Failure Predictions

Experiments and Results

Prediction Framework



Prediction Attributes

Text-based

- # Posts in discussion
- # Words per post
- # Words per discussion
- Bag of words
- TF-IDF

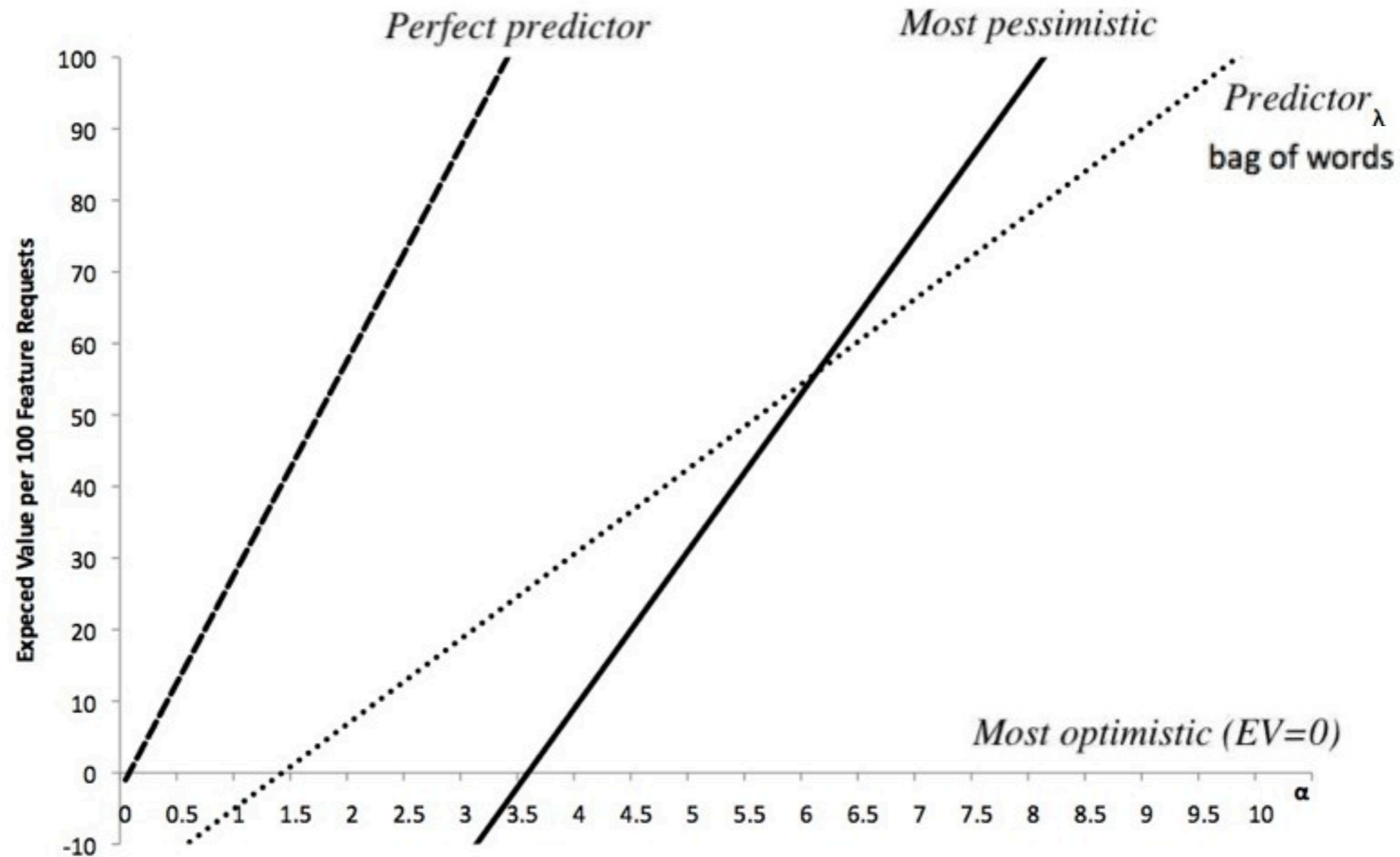
Meta-data

- # Code contributions
- Time elapsed per post
- Time elapsed for discussion
- # Participants
- # and % Posts by reporter
- # and % Posts by assignee

Predictors With Most Expected Value

	Product Failure	Abandoned Implement.	Rejection Reversal	Stalled Development	Removed Feature
Firefox	Posts in Discussion	Most Pessimistic	Most Optimistic	Code Contributions	Most Optimistic
Netbeans	Most Optimistic	TF-IDF	Words in Discussion	Precent by Assignee	Bag of Words

Firefox Product Failure Predictions



Estimated: $P_s \cdot \frac{C_f}{C_a} = \alpha = 0.3 * 10 = \underline{\underline{3}}$

Experimental Findings

- Value could be gained by actioning early failure predictions in 6 of 10 experiments
- Predictions remain valuable even when α is uncertain
- Text-based predictors yielded far more expected value than meta-data predictors

Related Work

- Failure prediction at later stages of the software life-cycle [*Nagappan et al. ICSE'06, Wolf et al. ICSE'09*]
- Guidelines for good feature request management [*Bettenburg et al. FSE'08*]
- Machine learning for prioritisation and clustering of features [*Duan et al. RE'09, Duan & Cleland-Huang ASE'07*]

Conclusion

- Early failure predictions determine when to perform further upfront requirements analysis
- Contributions:
 - Failure definitions
 - Early failure prediction framework
 - Cost-model for actioning predicted failures
- Future work:
 - More case studies
 - Better predictive models