Popularity Prediction of Facebook Videos for Higher Quality Streaming

Linpeng Tang[*]

Qi Huang[♭], Amit Puntambekar[♭]

Ymir Vigfusson[†], Wyatt Lloyd[‡♭], Kai Li[*]

PRINCETON UNIVERSITY[*]   EMORY UNIVERSITY[†]   USC University of Southern California[‡]   facebook[♭]
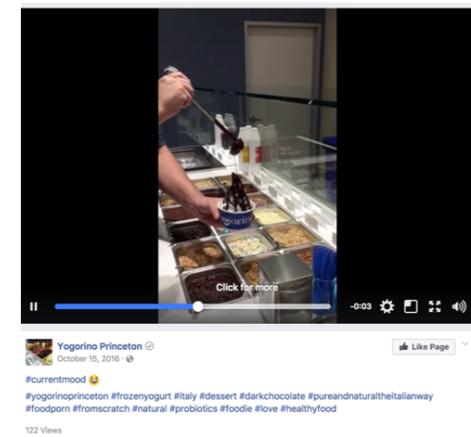
Videos are Central to Facebook
8 billion views per day
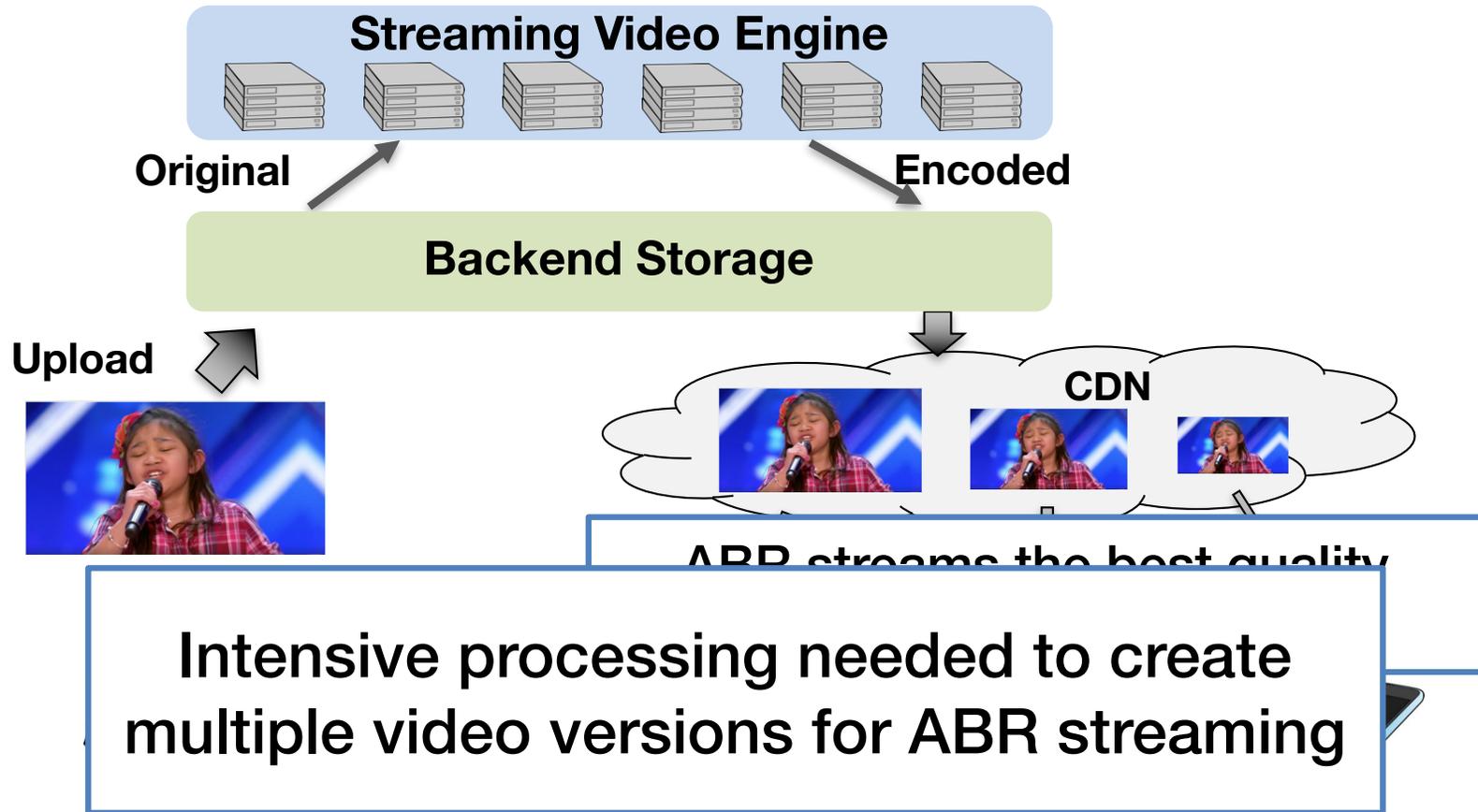

9-year old singing on
America's Got Talent
44M views

Black bear roaming
in Princeton
3.8K views

Small shop making
frozen yogurt
122 views

Workflow of Videos on Facebook

Streaming Video Engine

Original

Encoded

Backend Storage

Upload

CDN

ABR streams the best quality

Intensive processing needed to create multiple video versions for ABR streaming

3

Better Video Streaming from More Processing

- Better compression at the same quality
- QuickFire: 20% size reduction using 20X computation
- More users can view the high quality versions

Better Video Streaming from More Processing
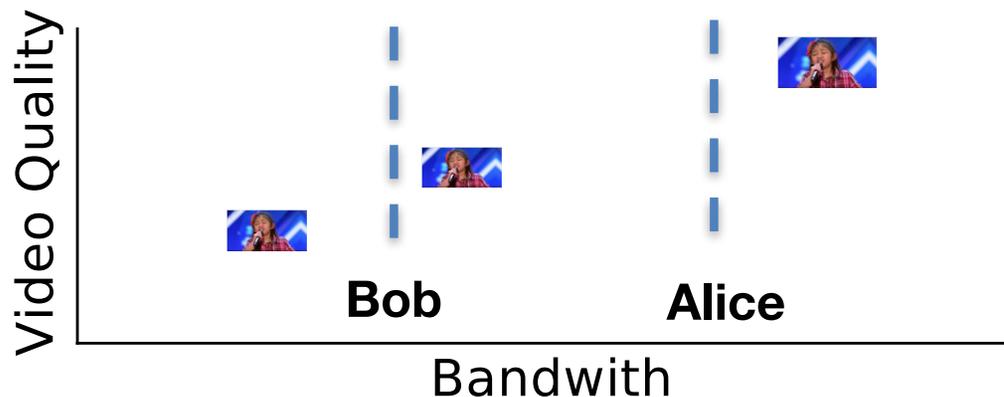
- Better compression at the same quality
- QuickFire: 20% size reduction using 20X computation
- More users can view the high quality versions


Better Compression
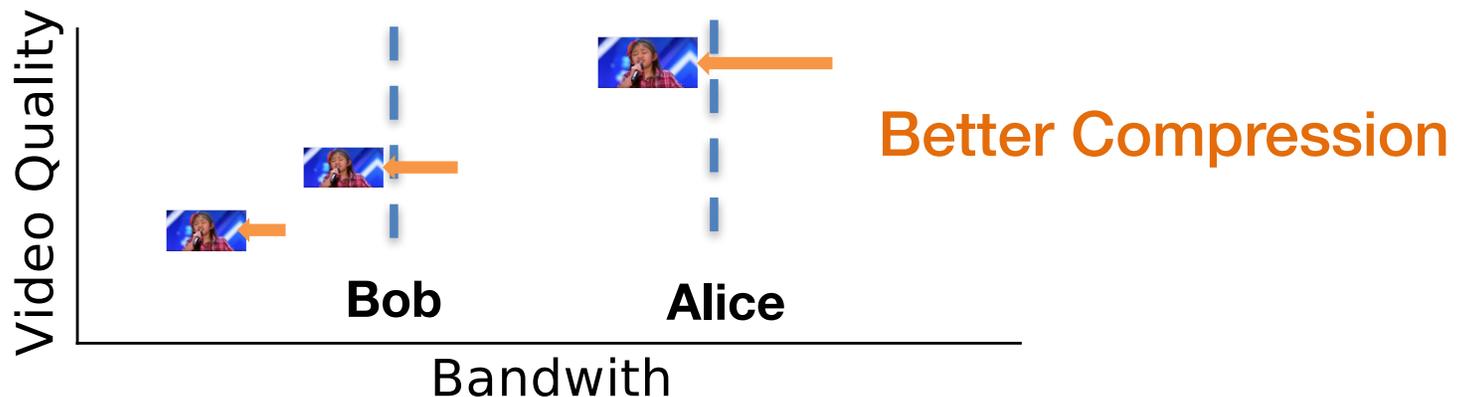
How to apply QuickFire for FB videos

- **Infeasible to encode all videos with QuickFire**
  - Increase by 20X the already large processing fleet

- **High skew in popularity**
 - Reap most benefit with modest processing?

6

Opportunity: High Skew in Popularity

- Access logs of 1 million videos randomly sampled by ID
- Watch time: total time users spent watching a video
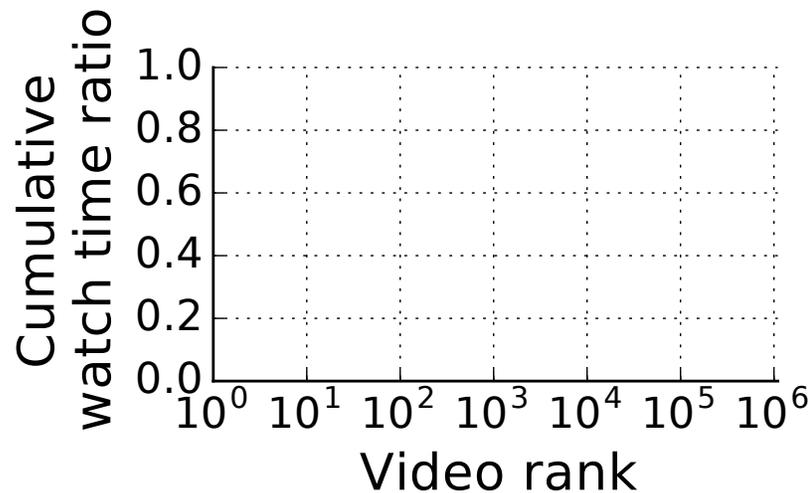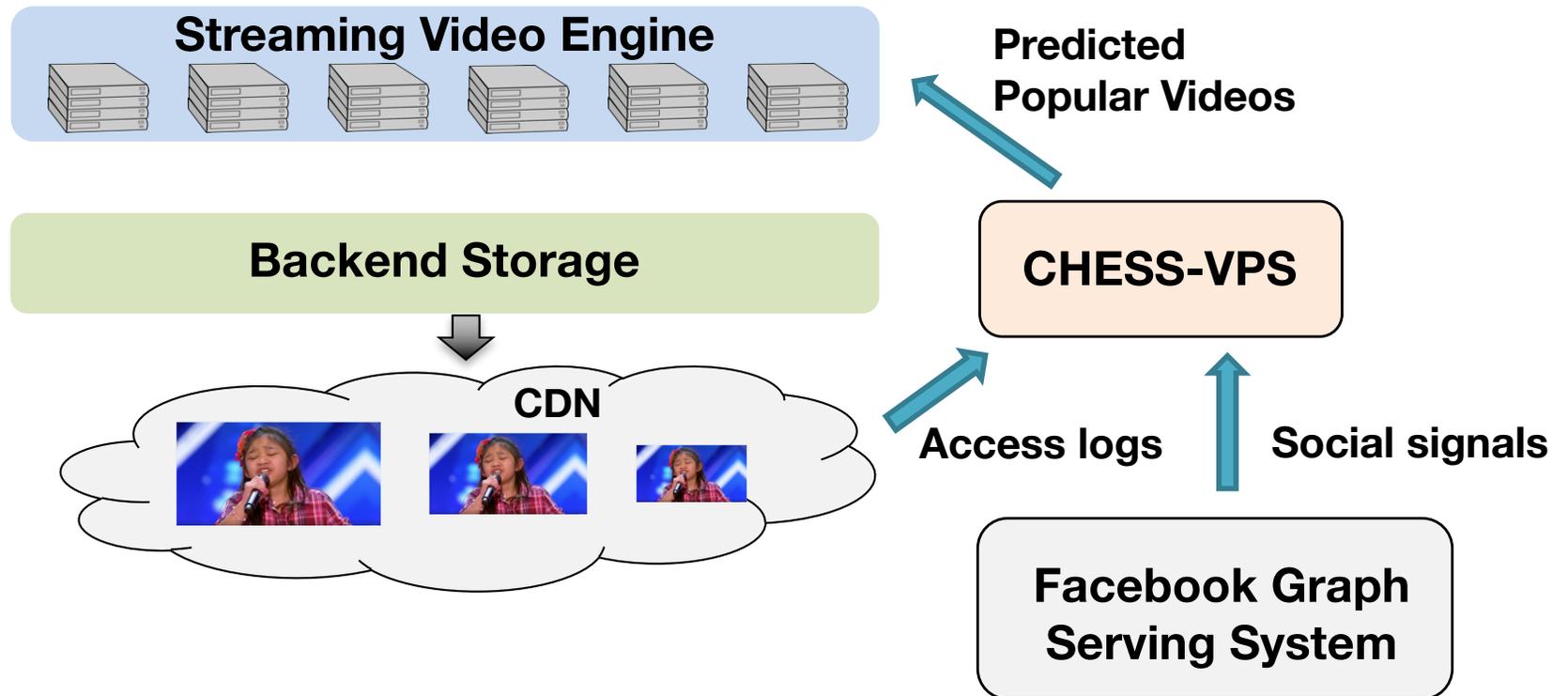
Opportunity: High Skew in Popularity

- We can serve most watch time even with a small fraction of videos encoded with QuickFire

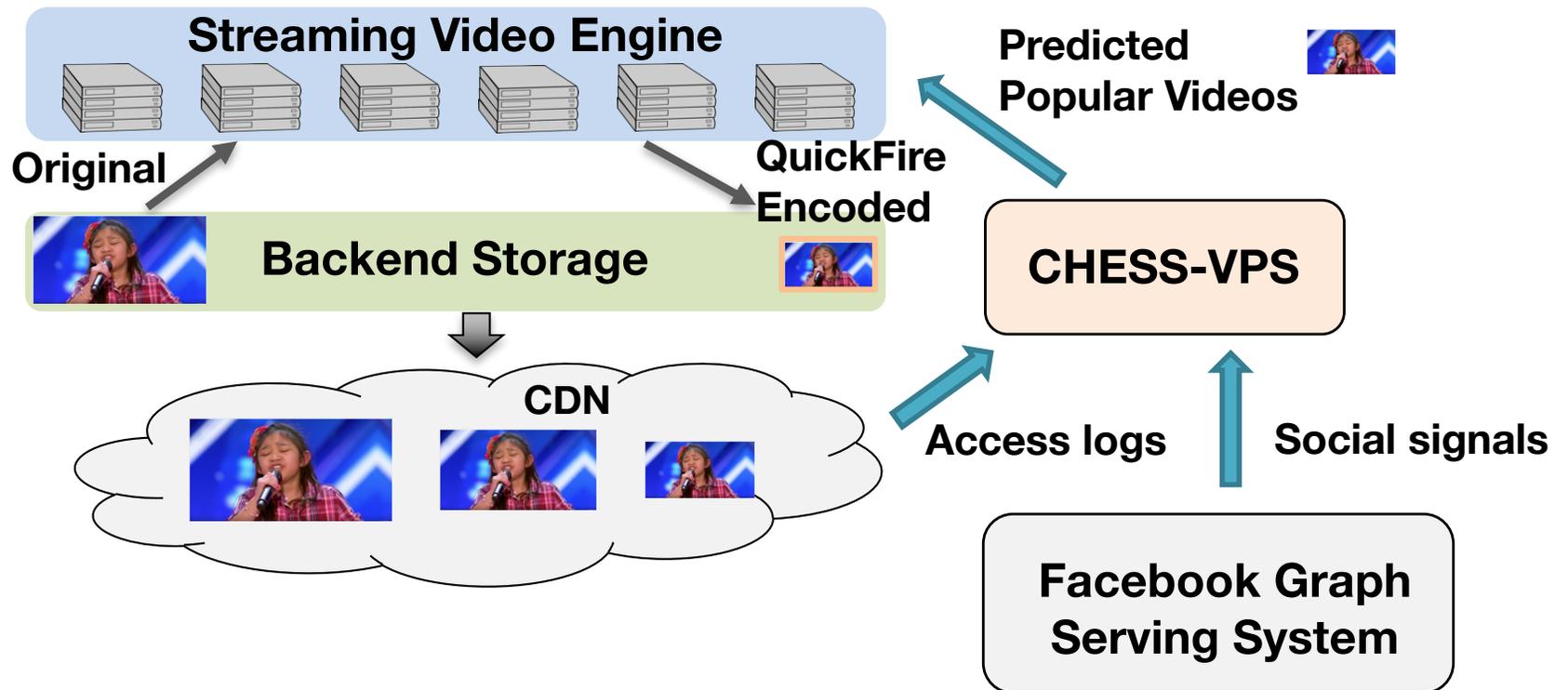- Can we predict these videos for more processing?

80%+ watch time

Cumulative watch time ratio vs. Video rank
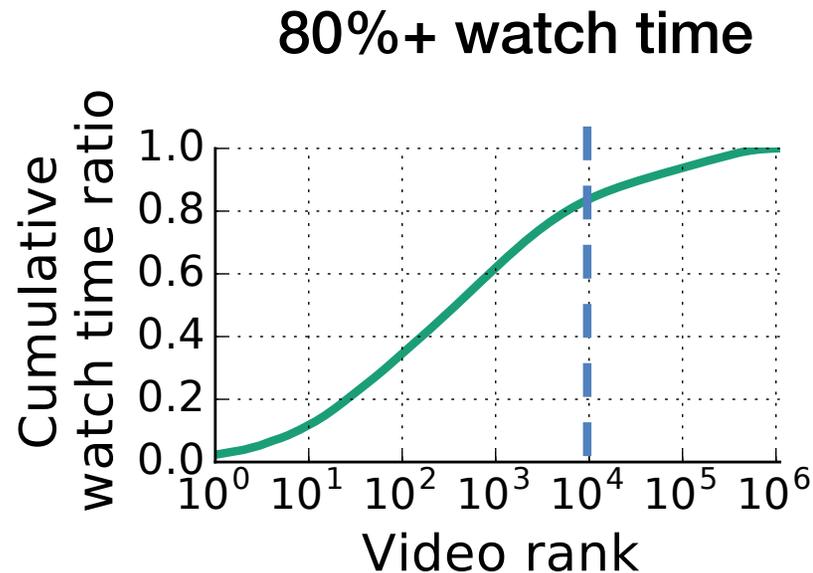
CHESS Video Prediction System

- Popularity prediction is important for higher quality streaming
  - Direct encoding on videos with the largest benefit

- Goal of CHESS video prediction system
  - Identify videos with highest future watch time
  - Maximize watch-time ratio with budgeted processing
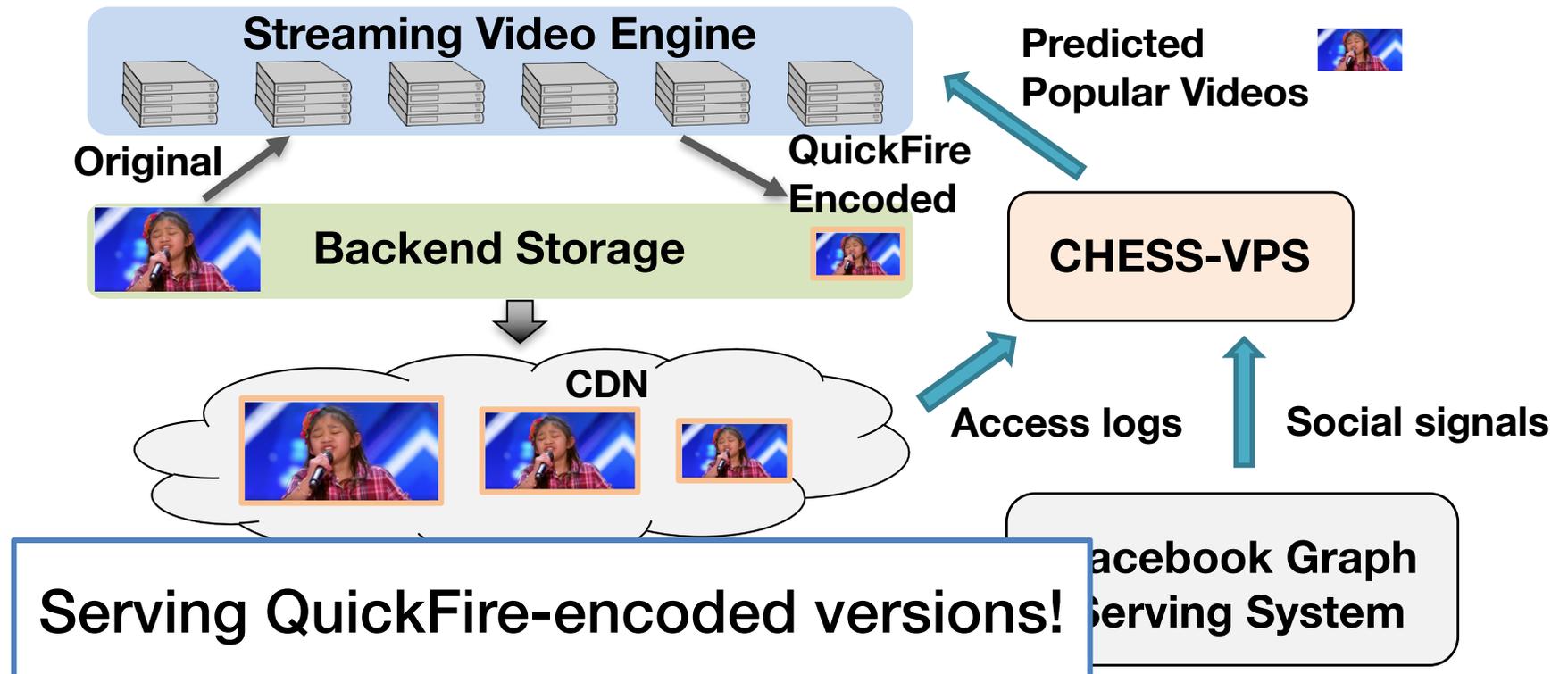
CHESS Video Prediction System

CHESS Video Prediction System

CHESS Video Prediction System

Streaming Video Engine

Original

QuickFire Encoded

Backend Storage

CDN

Predicted Popular Videos

CHESS-VPS

Access logs

Social signals

Facebook Graph Serving System

Serving QuickFire-encoded versions!

12

Requirements of CHESS-VPS

- Handle working set of ~80 million videos
- Generate new predictions every few minutes

- Requires a new prediction algorithm: CHESS!

CHESS Key Insights

- Efficiently model influence of past accesses as the basis for scalable prediction

- Combine multiple predictors to boost accuracy

Efficiently model past access influence

- Self exciting process
  - A past access makes future accesses more probable, i.e. provides some *influence* on future popularity

Influence of past accesses

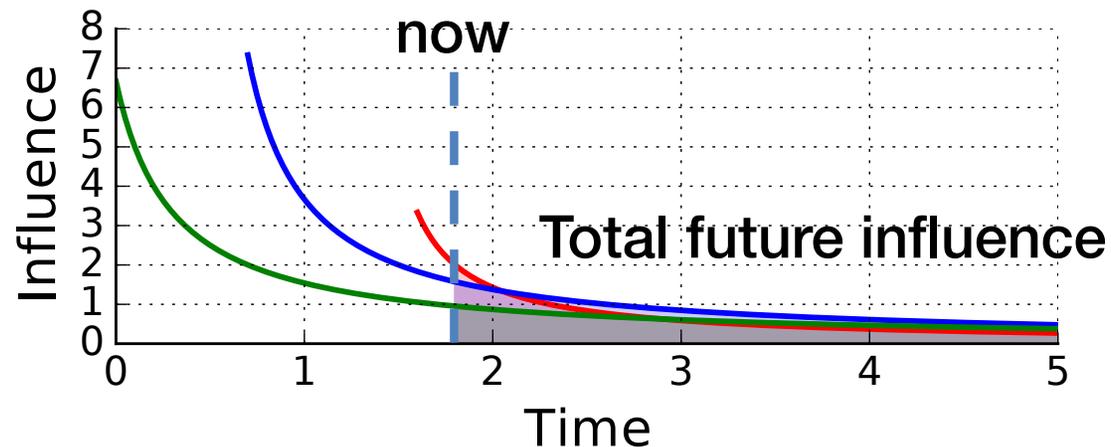(Graph: x-axis "Time" from 0 to 5, y-axis "Influence" from 0 to 8)
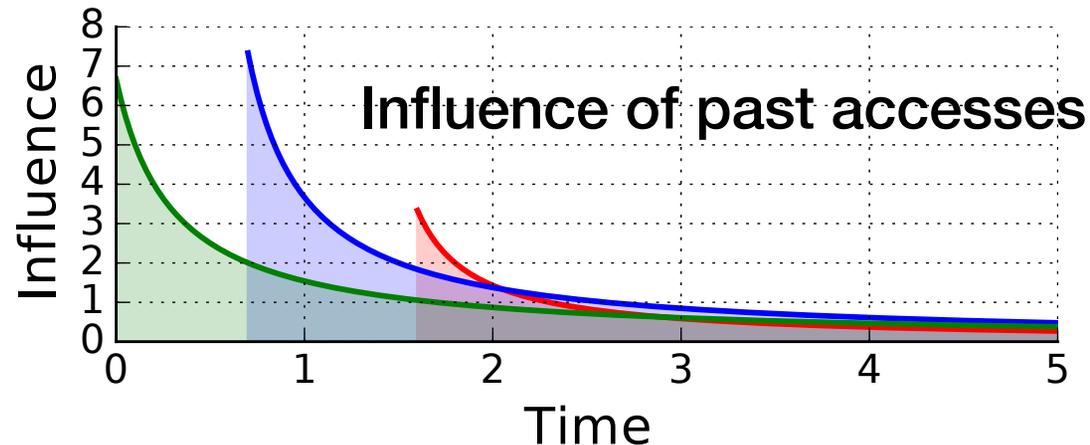
Efficiently model past access influence

- Self exciting process
  - A past access makes future accesses more probable, i.e. provides some *influence* on future popularity
  - Prediction: sum up total future influence of all past accesses
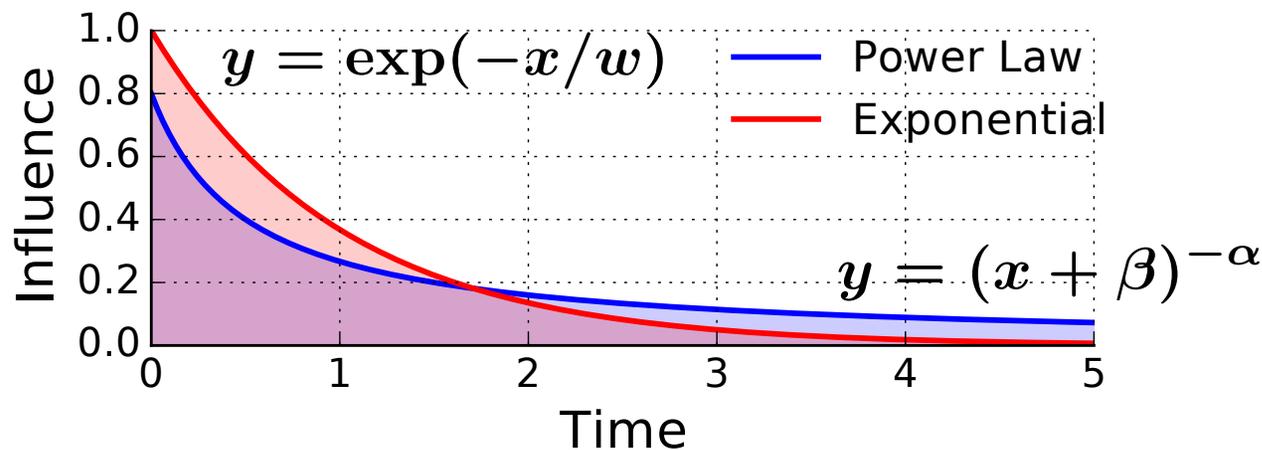
Efficiently model past access influence

- Influence modeled with kernel function
- Power-law kernel used by prior works
  - Provides high accuracy
  - Scan all past accesses, O(N) time/space not scalable
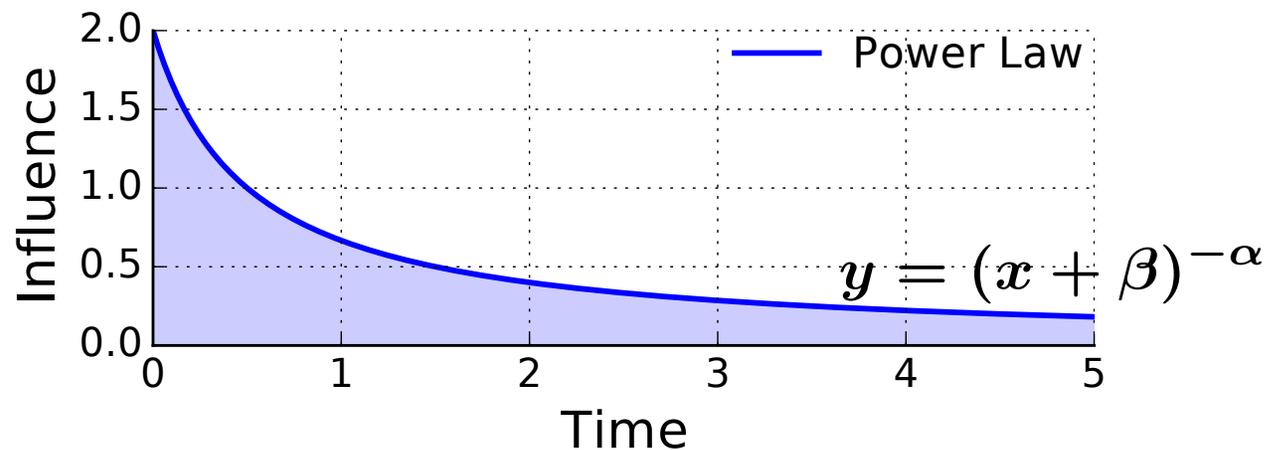
$$y = (x + \beta)^{-\alpha}$$

17

Efficiently model past access influence

- Influence modeled with kernel function
- Power-law kernel used by prior works
- Key insight: use exponential kernel for scalability

$$y = \exp(-x/w)$$

$$y = (x + \beta)^{-\alpha}$$

Power Law
Exponential

Influence

Time

18

Efficiently model past access influence

- Self exciting process with the exponential kernel

$$\tilde{F}(t) = \frac{x}{w} + \exp\left(\frac{-(t-u)}{w}\right)\tilde{F}(u)$$

*Current Access
Watch-time* + *Exponential
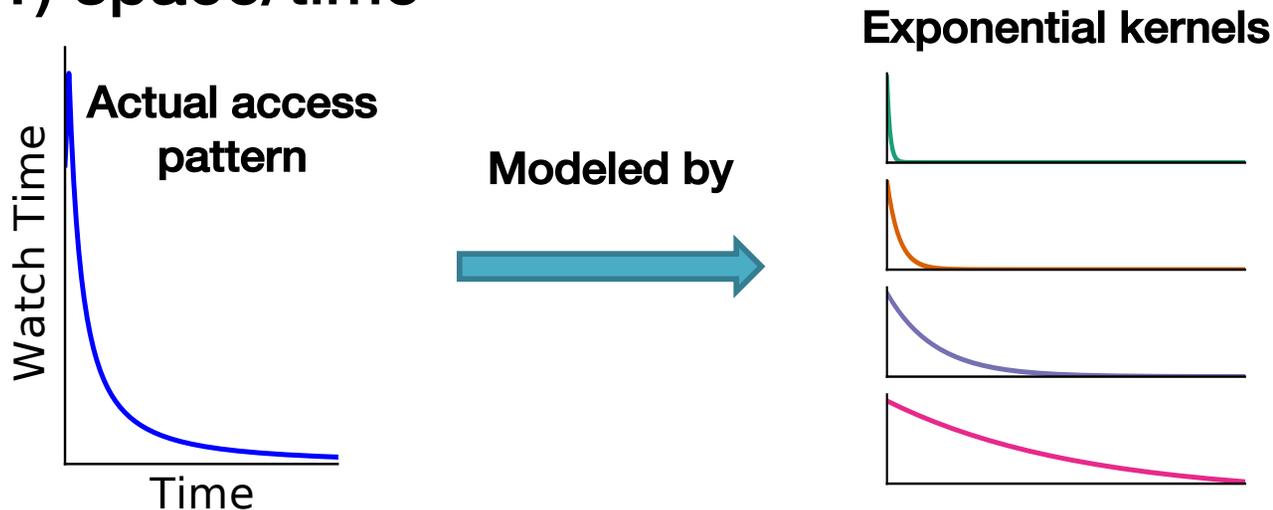Decay* x *Previous
Prediction*

Efficiently model past access influence

- Single exponential kernel is less accurate than power-law kernel
    - 10% lower watch time ratio

- O(1) space/time to maintain

Single exponential kernel is
less accurate yet scalable

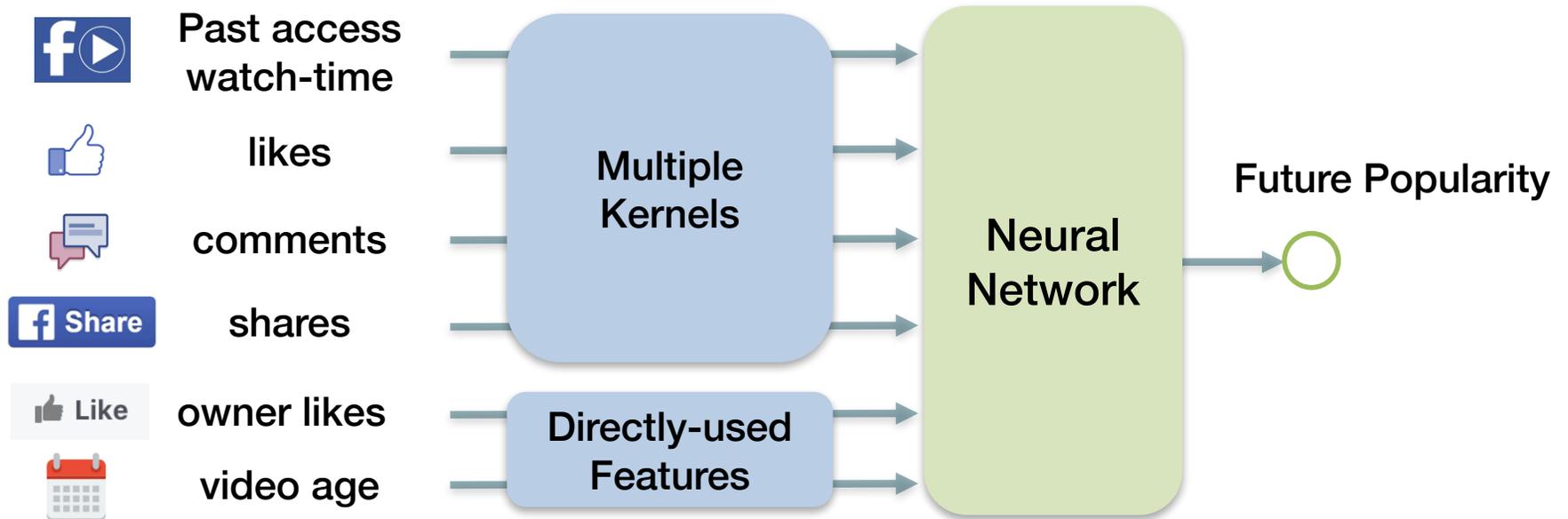# Combining Efficient Features in a Model

- Key insight: maintain multiple exponential kernels
- O(1) space/time

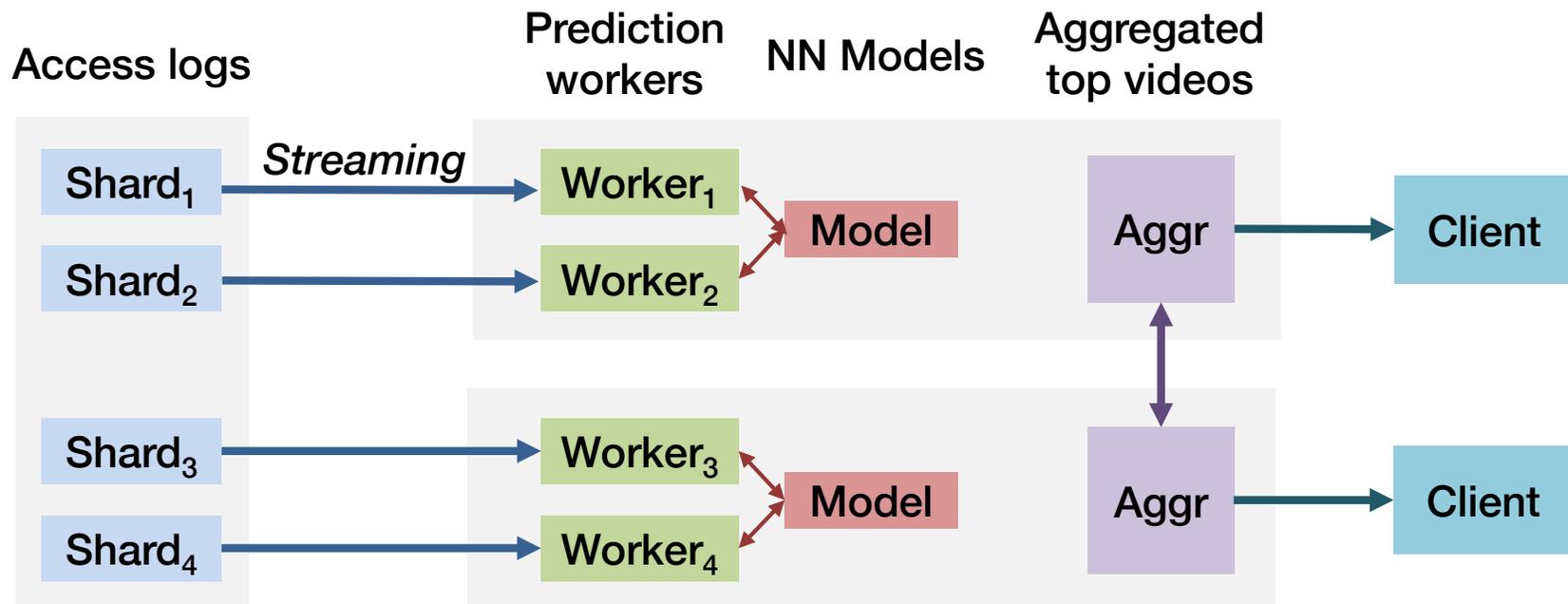Combining multiple exponential kernels is
as accurate as a power-law kernel

Combining Efficient Features in a Model

Raw features

Past access watch-time	→ Multiple Kernels →
likes	
comments	
shares	
owner likes	Directly-used Features
video age	

Neural Network → **Future Popularity**

Social signals further boosts accuracy

CHESS Video Prediction System

Access logs   Prediction workers   NN Models   Aggregated top videos

Shard$_1$ — *Streaming* → Worker$_1$

Shard$_2$ → Worker$_2$

Model → Aggr → Client

Shard$_3$ → Worker$_3$

Shard$_4$ → Worker$_4$

Model → Aggr → Client

Evaluation

- What is the accuracy of CHESS?

- How do our design decisions on CHESS affect its accuracy and resource consumption?

- What is CHESS's impact on video processing and watch time ratio of QuickFire?

Evaluation

- What is the accuracy of CHESS?

- How do our design decisions on CHESS affect its accuracy and resource consumption?

- What is CHESS's impact on video processing and watch time ratio of QuickFire?

Metrics

- ## Watch time ratio
  - Ratio of watch time from better encoded videos
  - Directly proportional to benefits of better encoding

- ## Processing time

Metrics

- ## Watch time ratio
  - Ratio of watch time from better encoded videos
  - Directly proportional to benefits of better encoding

- ## ~~Processing time~~ (infeasible to encode all videos)
  - Video length $\propto$ processing time
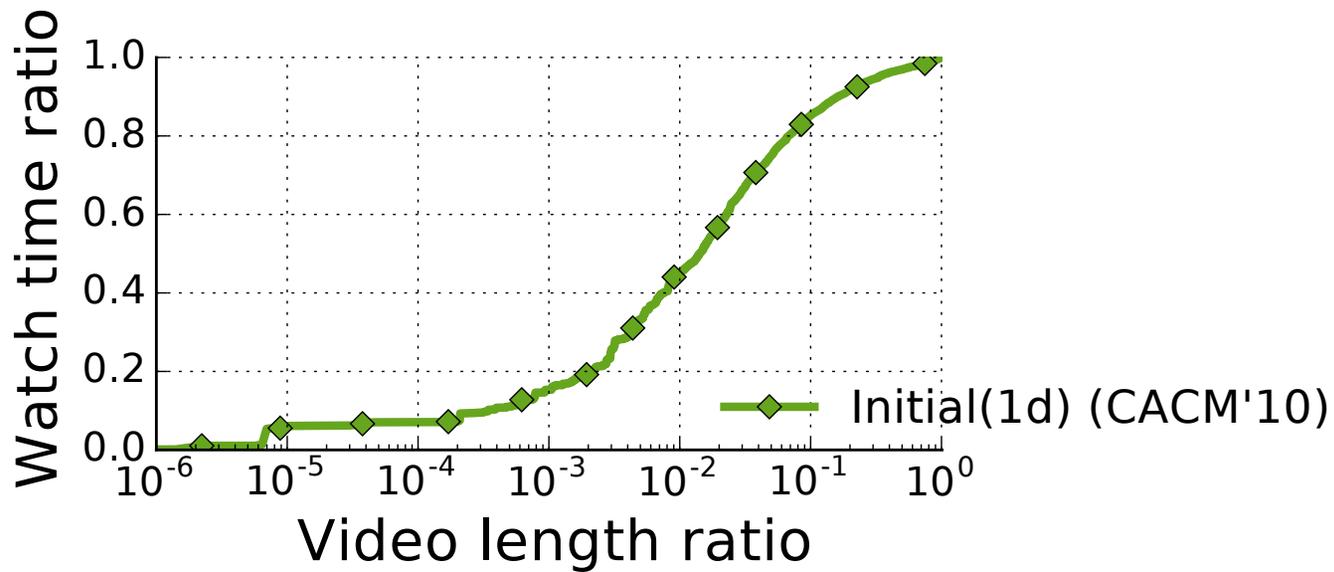 - Video length ratio $\approx$ computation overhead

CHESS is Accurate

- Vary video length ratio (proxy for processing overhead)
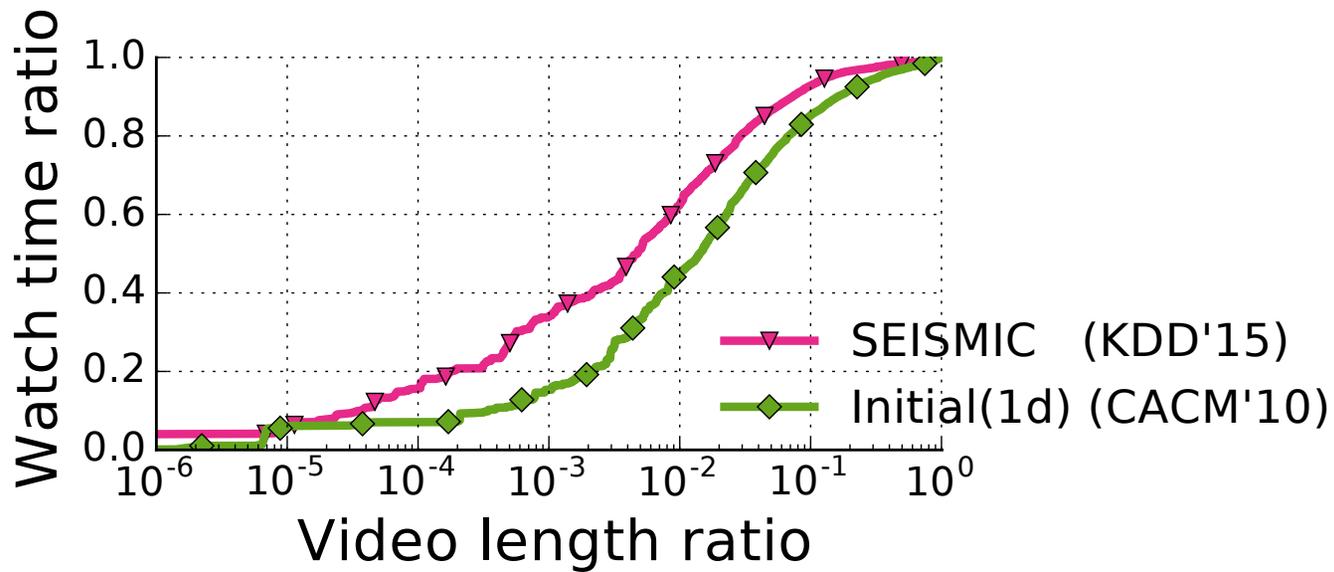- Observe watch time ratio of better encoded videos
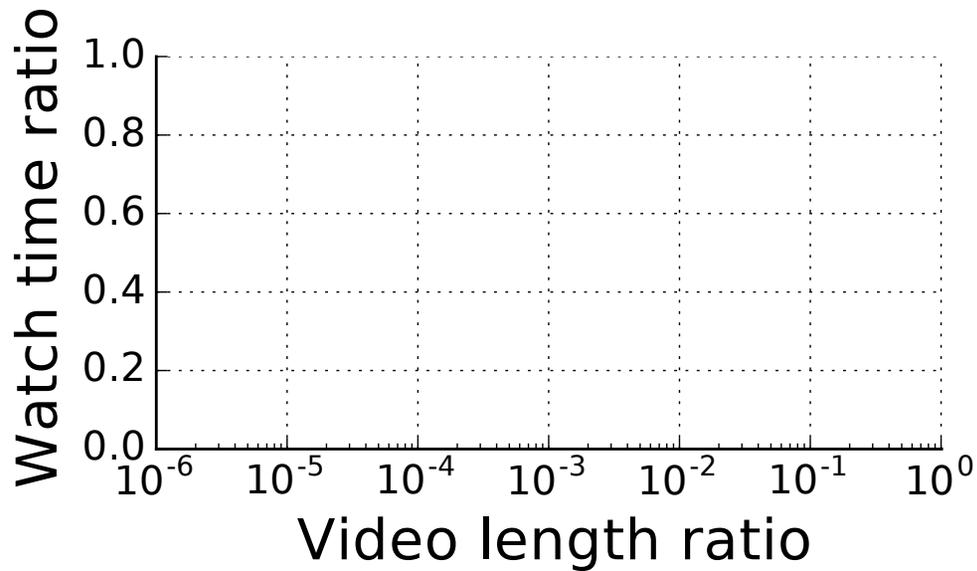
CHESS is Accurate

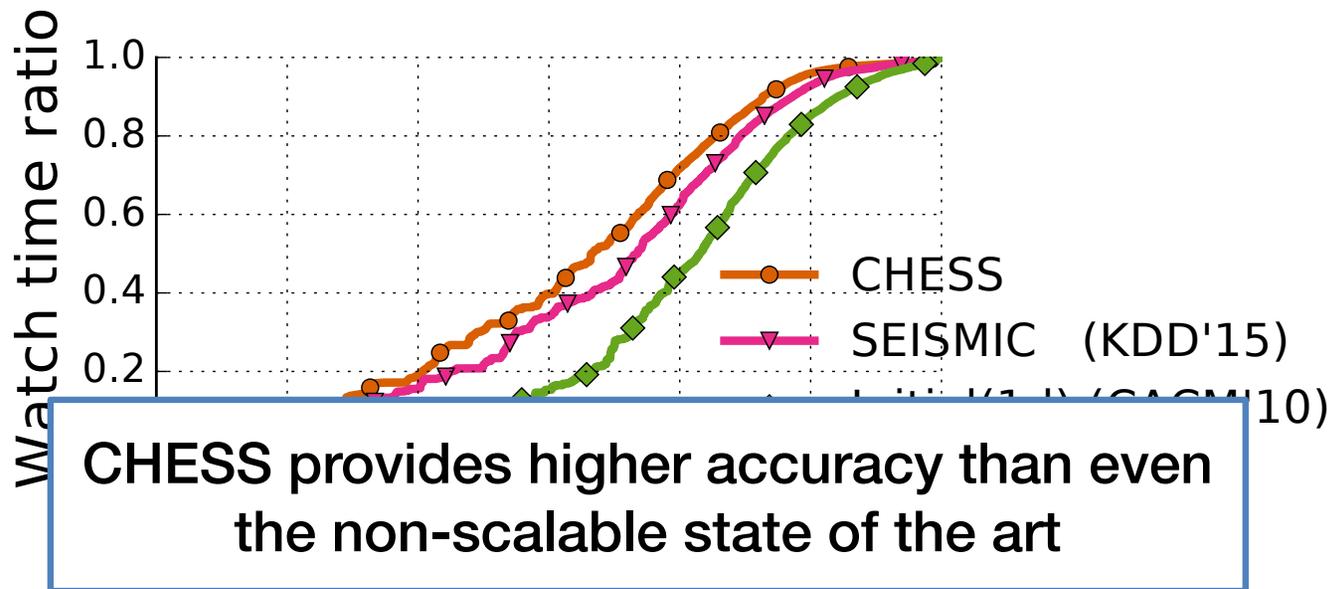- Initial(1d): initial watch time up to 1 day after upload

CHESS is Accurate

- Initial(1d): initial watch time up to 1 day after upload
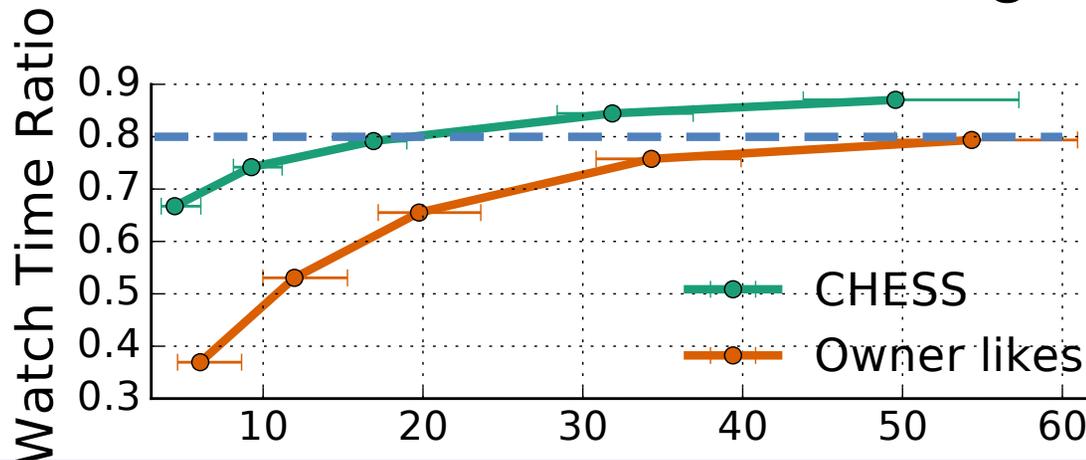- SESIMIC: handcrafted power-law kernel

CHESS is Accurate

- Initial(1d): initial watch time up to 1 day after upload
- SESIMIC: handcrafted power-law kernel

CHESS provides higher accuracy than even
the non-scalable state of the art

CHESS Reduces Encoding Processing

- Predict on whole Facebook video workload in real-time
- Sample 0.5% videos for actual encoding

CHESS reduces CPU by 3x (54% to 17%)
for 80% watch time ratio

Related Work

Popularity Prediction
Hawkes'71, Crane'08, Szabo'10, Cheng'14, SEISMIC'15

CHESS is scalable and accurate

Video QoE Optimization
Liu'12, Aaron'15, Huang'15, Jiang'16, QuickFire'16

Optimize encoding with access feedback

Caching
LFU'93, LRU'94, SLRU'94, GDS'97, GDSF'98, MQ'01

Identify hot items to improve efficiency

Conclusion

- Popularity prediction can direct encoding for higher quality streaming

- CHESS: first scalable and accurate popularity predictor
  - Model influence of past accesses with O(1) time/space
  - Combine multiple kernels & social signals to boost accuracy

- Evaluation on Facebook video workload
  - More accurate than non-scalable state of the art method
  - Serve 80% user watch time with 3x reduction in processing