Efficient Recompression for Storage Saving in VOD Streaming Services

Zhijun Ryan Lei, Klaus Schneider, Anand Vadera, Mohanish Penta, Chen Fan
Video Infrastructure, Meta
Thibault Vatter,
University of Applied Science and Arts Western Switzerland

Mile High Video - 2024
Denver, USA, February 11-13, 2024

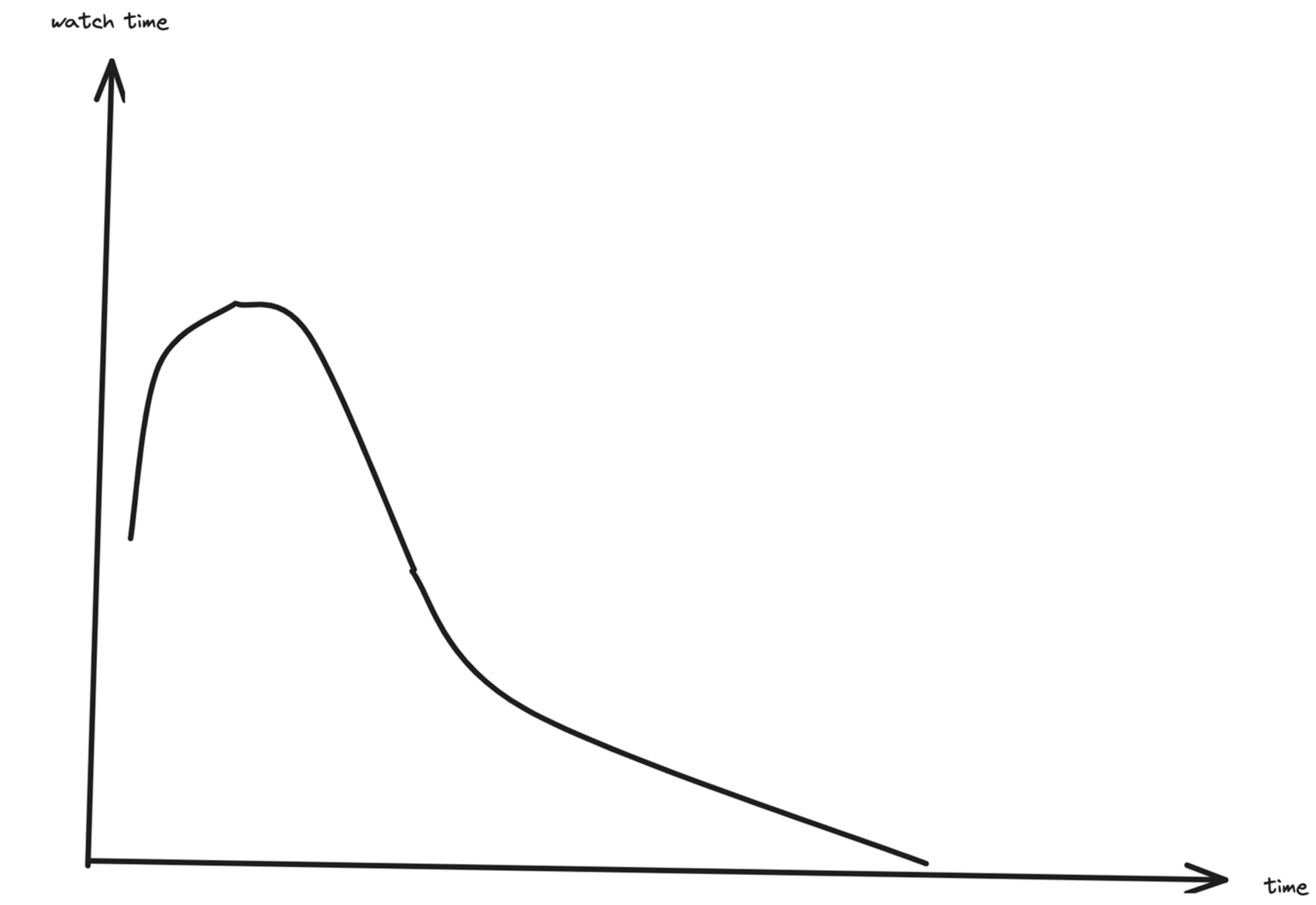
Meta

Agenda

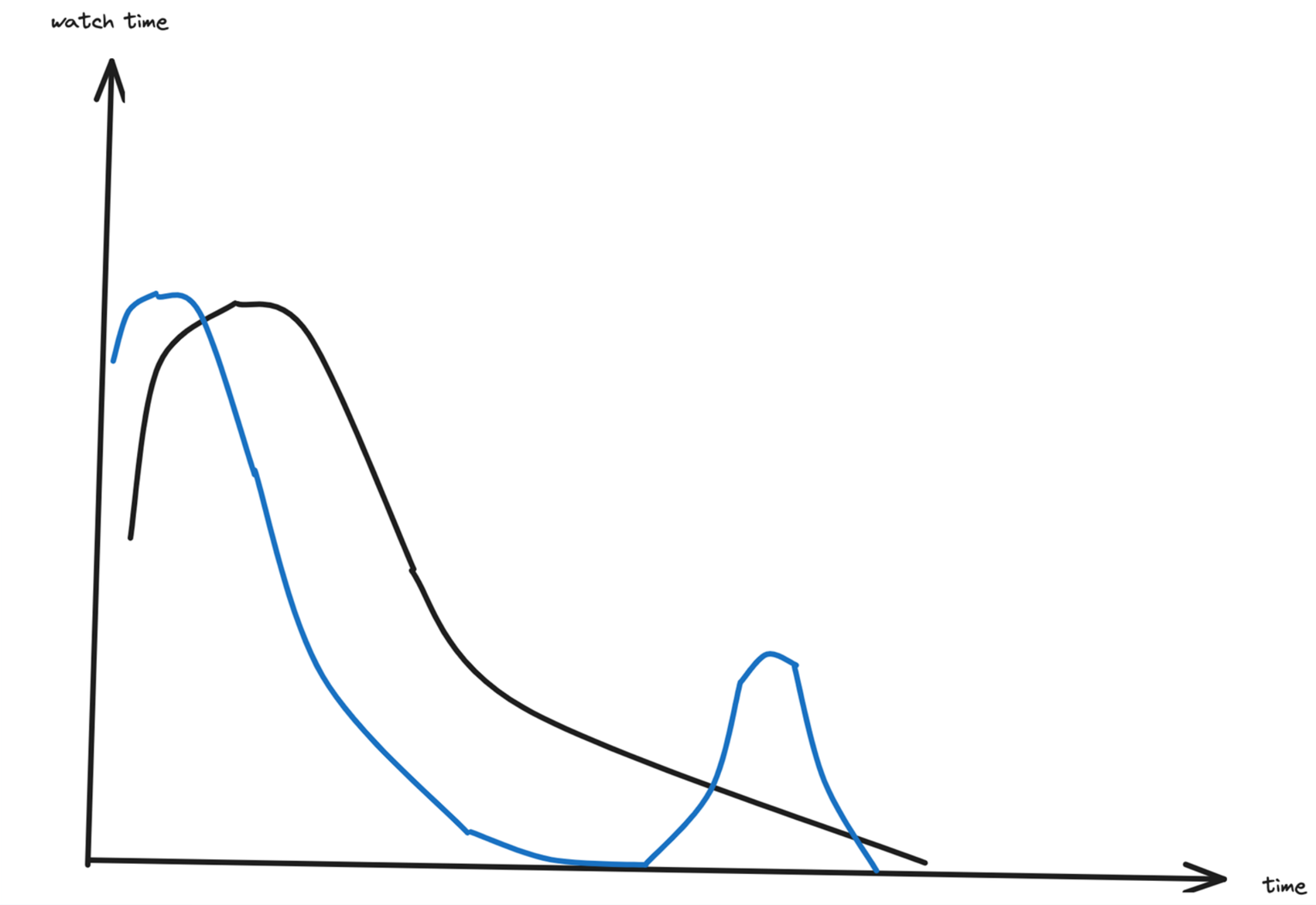
1. Video Watch Time and Life Cycle
2. Improving Storage Efficiency
3. Source Recompression for Storage Saving
4. ML Based Predictor

Video Watch Time and Life Cycle

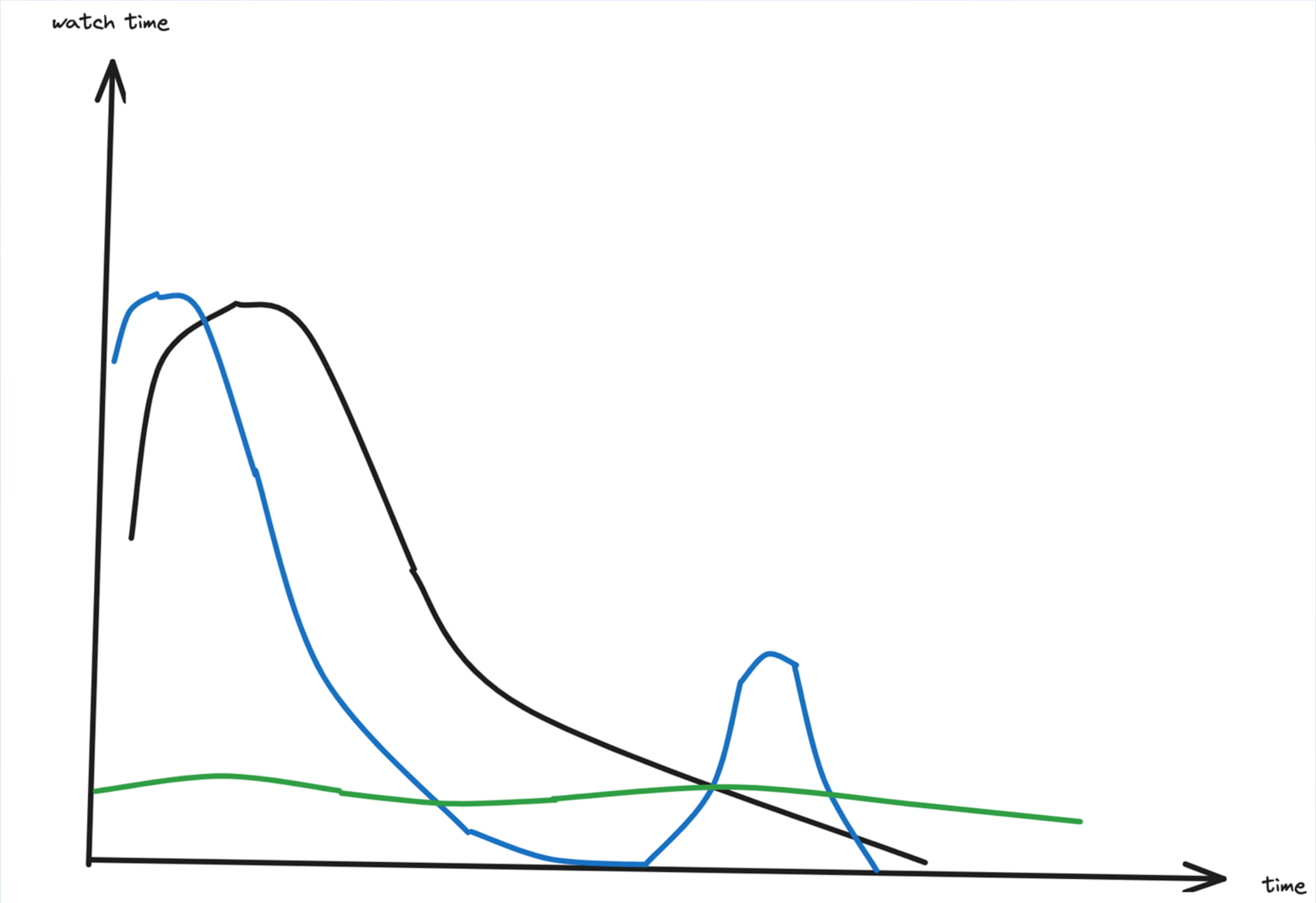
Watch Time by Video Age



Watch Time by Video Age



Watch Time by Video Age

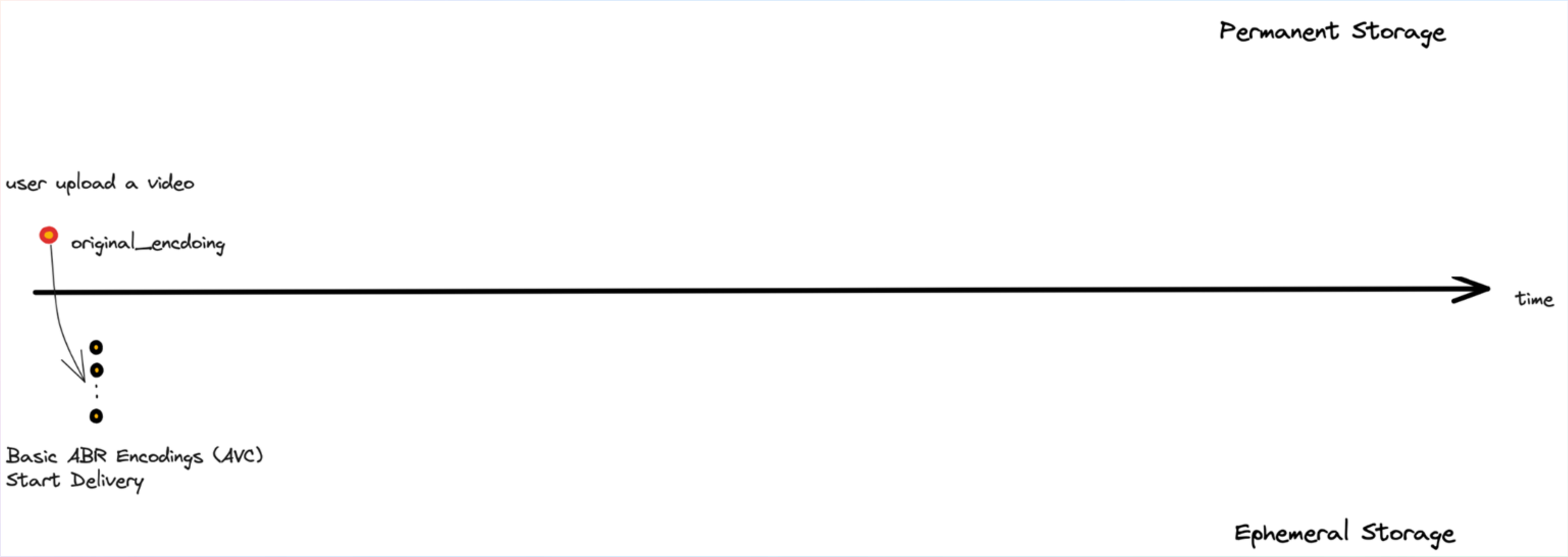


Meta Video Infra - Storage Efficiency

- Trade-off between **user experience** and **resource cost** (storage, compute)
  - -> **Increase user experience** when watchtime is high!
  - -> **Reduce storage footprint** when watchtime is low!

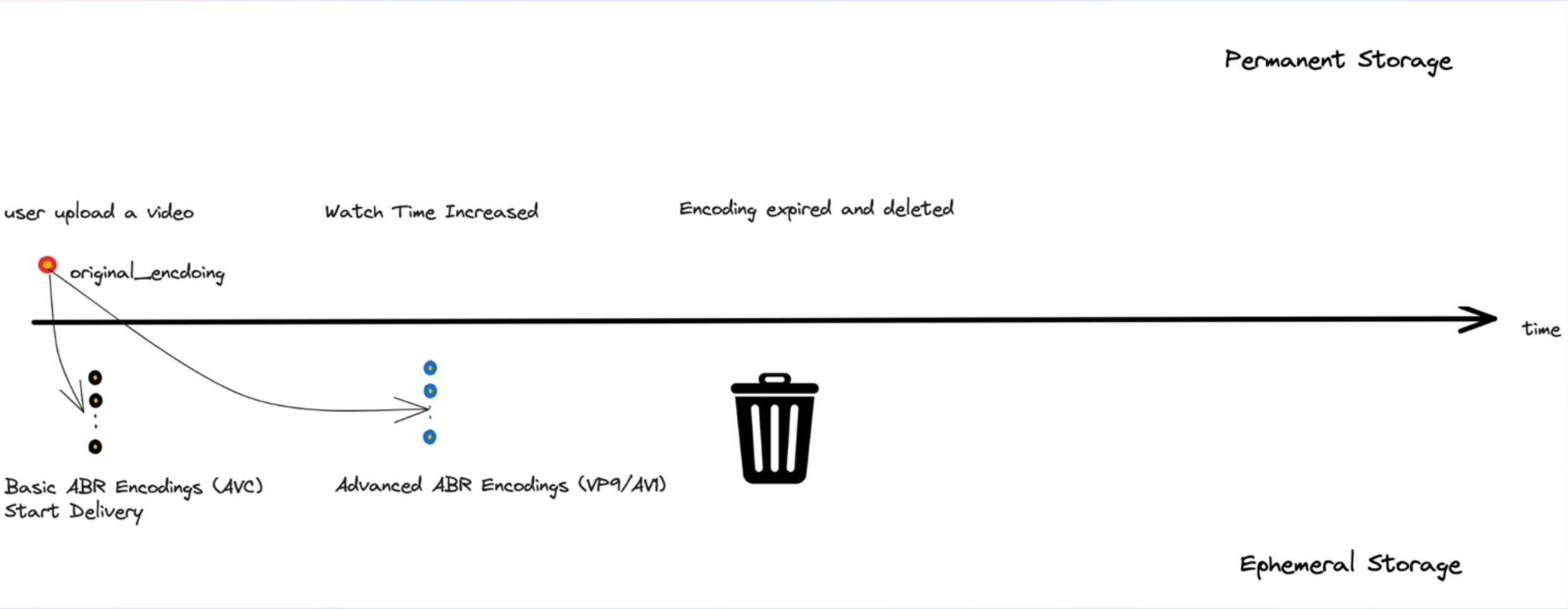
- Storage layout:
  - Permanent high-res/quality encoding (**“Source” encoding**)
  - Ephemeral ABR encodings for delivery

Video Life Cycle

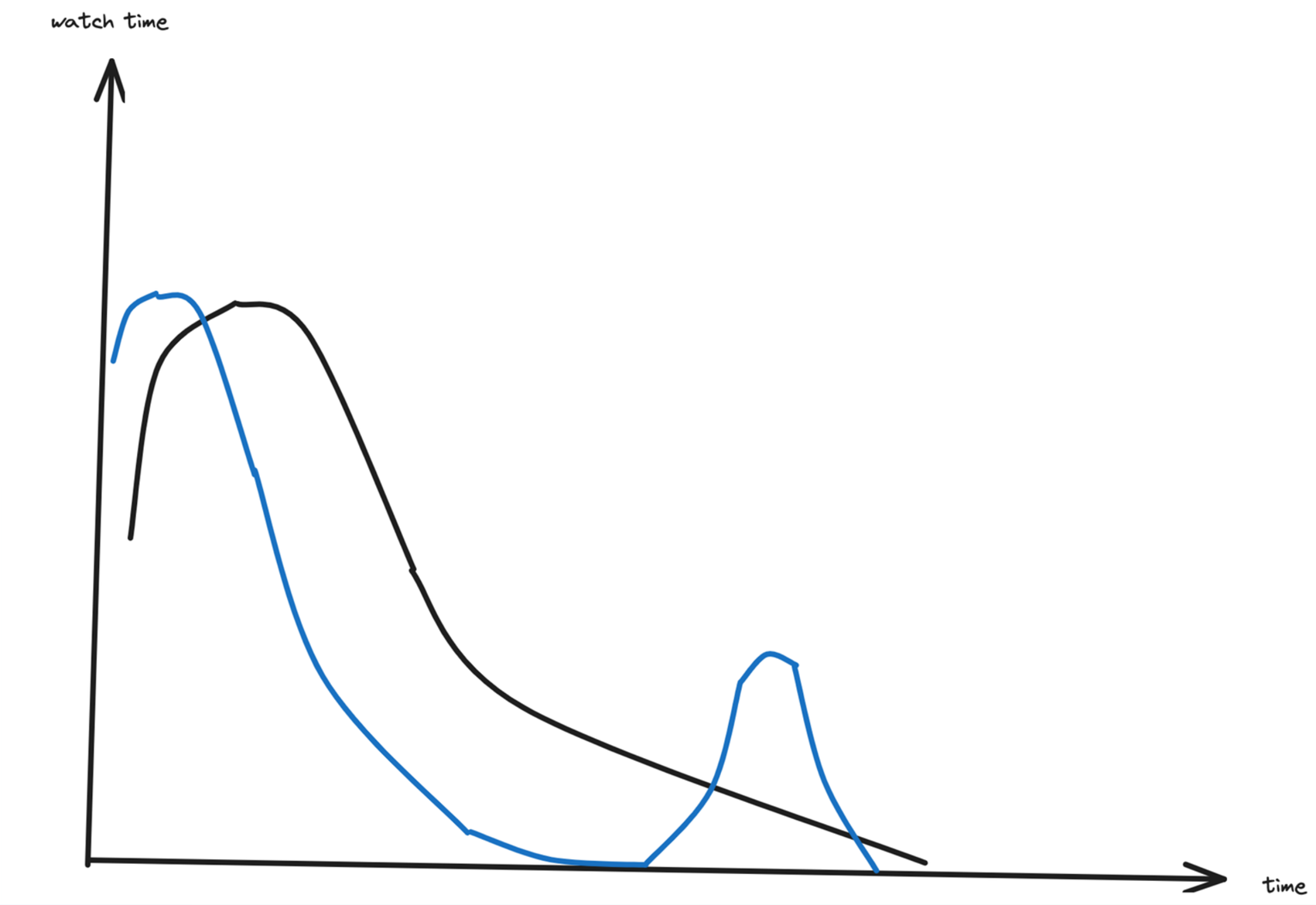


Video Life Cycle

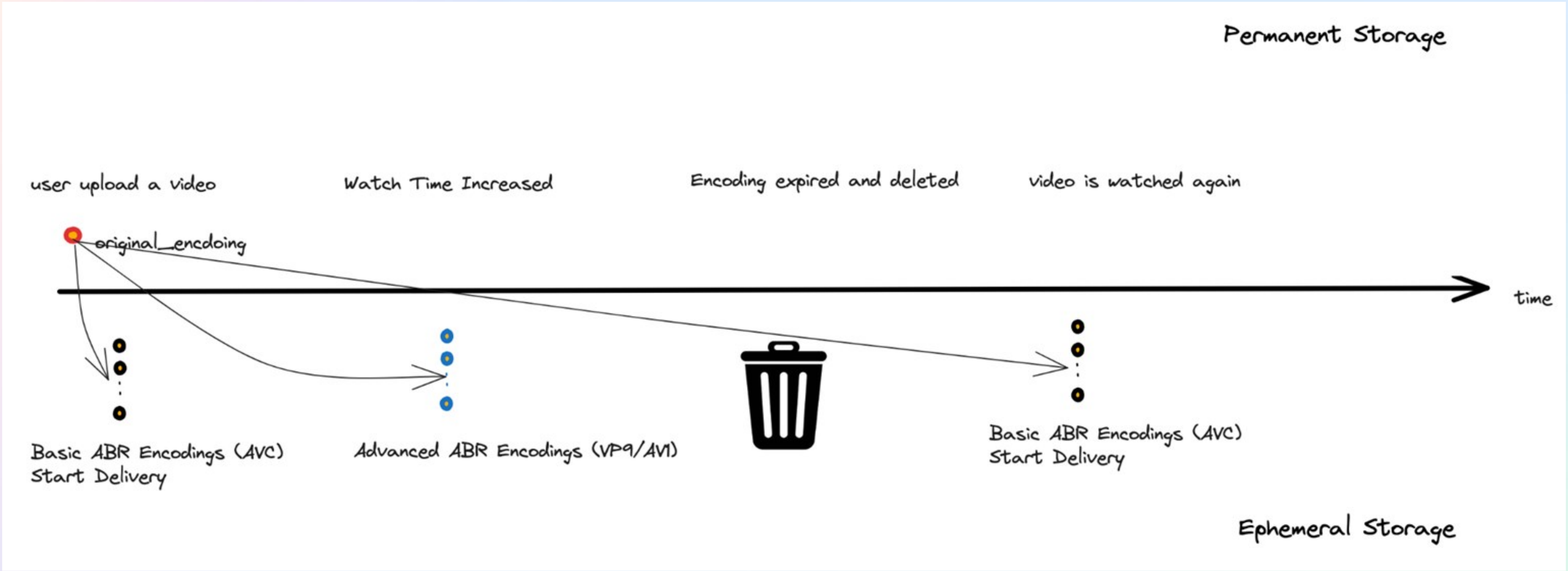
Video Life Cycle



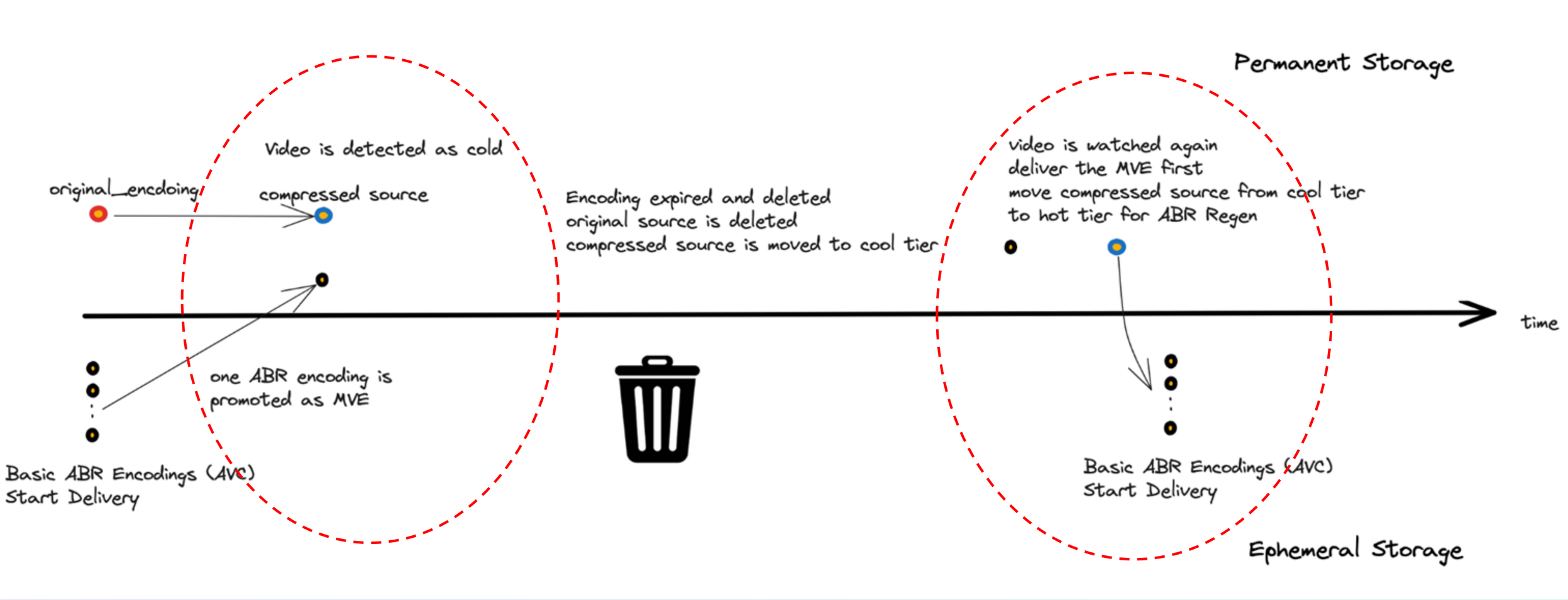
Watch Time by Video Age



Video Life Cycle



Further Improving Storage Efficiency



When video is detected as cold: Original_encoding —> compressed_source
When cold video is watched again:

- **Deliver the Minimum Viable Encoding (MVE) first**
- **Deliver the compressed_source, or regen ABR encoding from compressed_source**

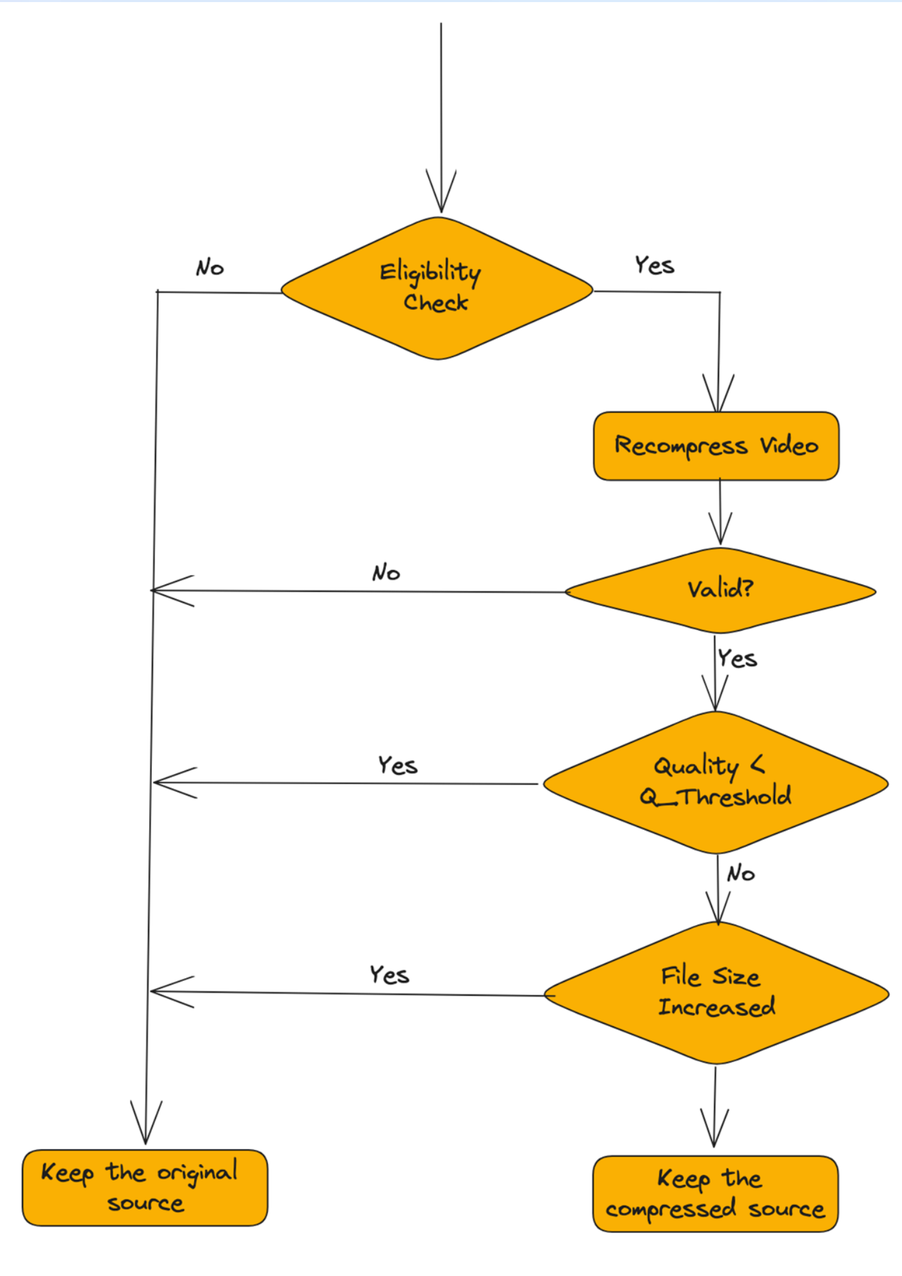
Source Recompression for Storage Saving

Requirement and Trade Offs for Source Recompression

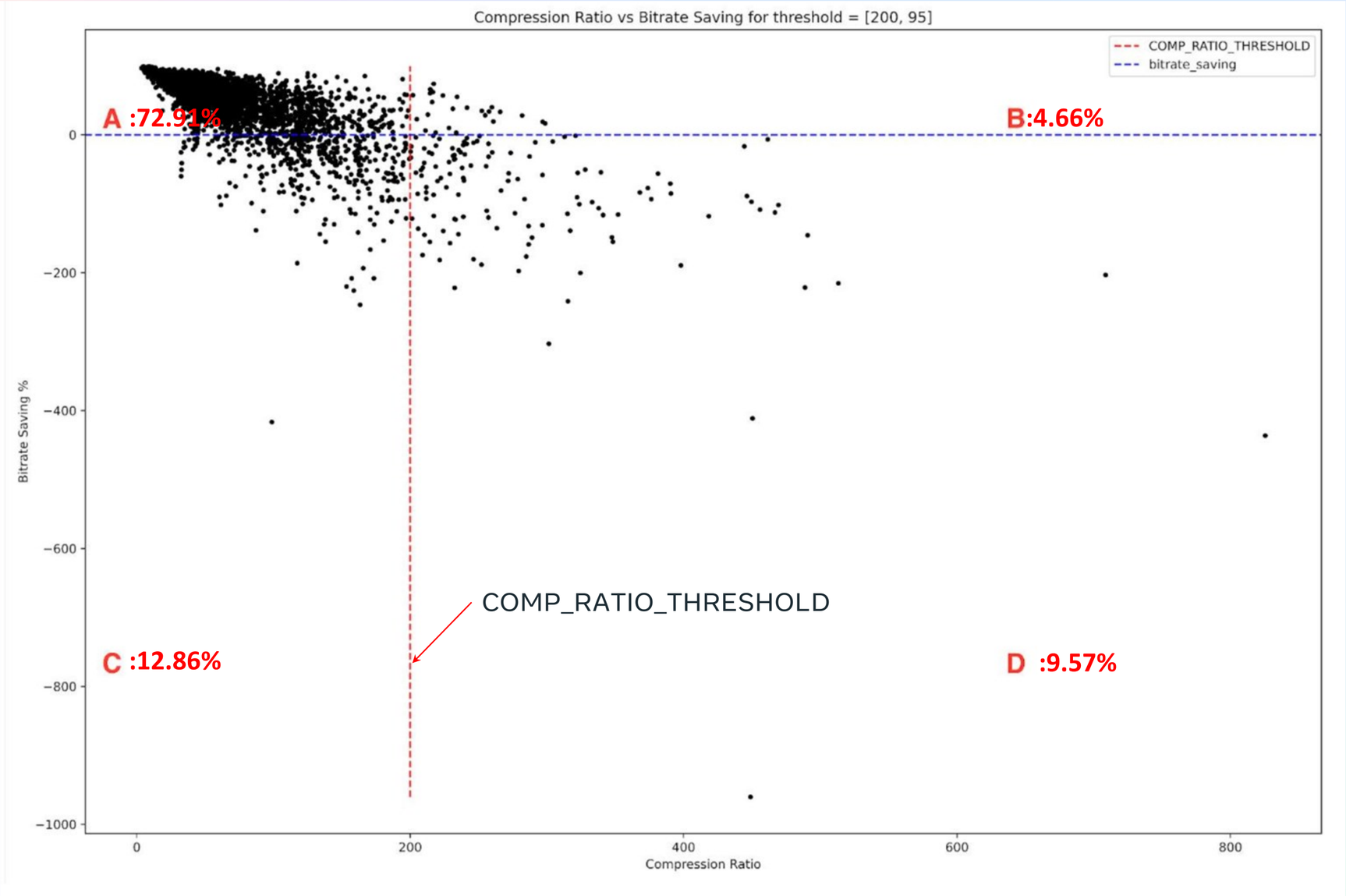
- Minimize the quality loss between compressed and the original source.
- Maximize storage saving.
- Minimize the compute cost for recompression.
- Deliver compressed source vs ABR regen

Process Flow for Source Recompression

- Eligibility check
- Transcode
- Post validation



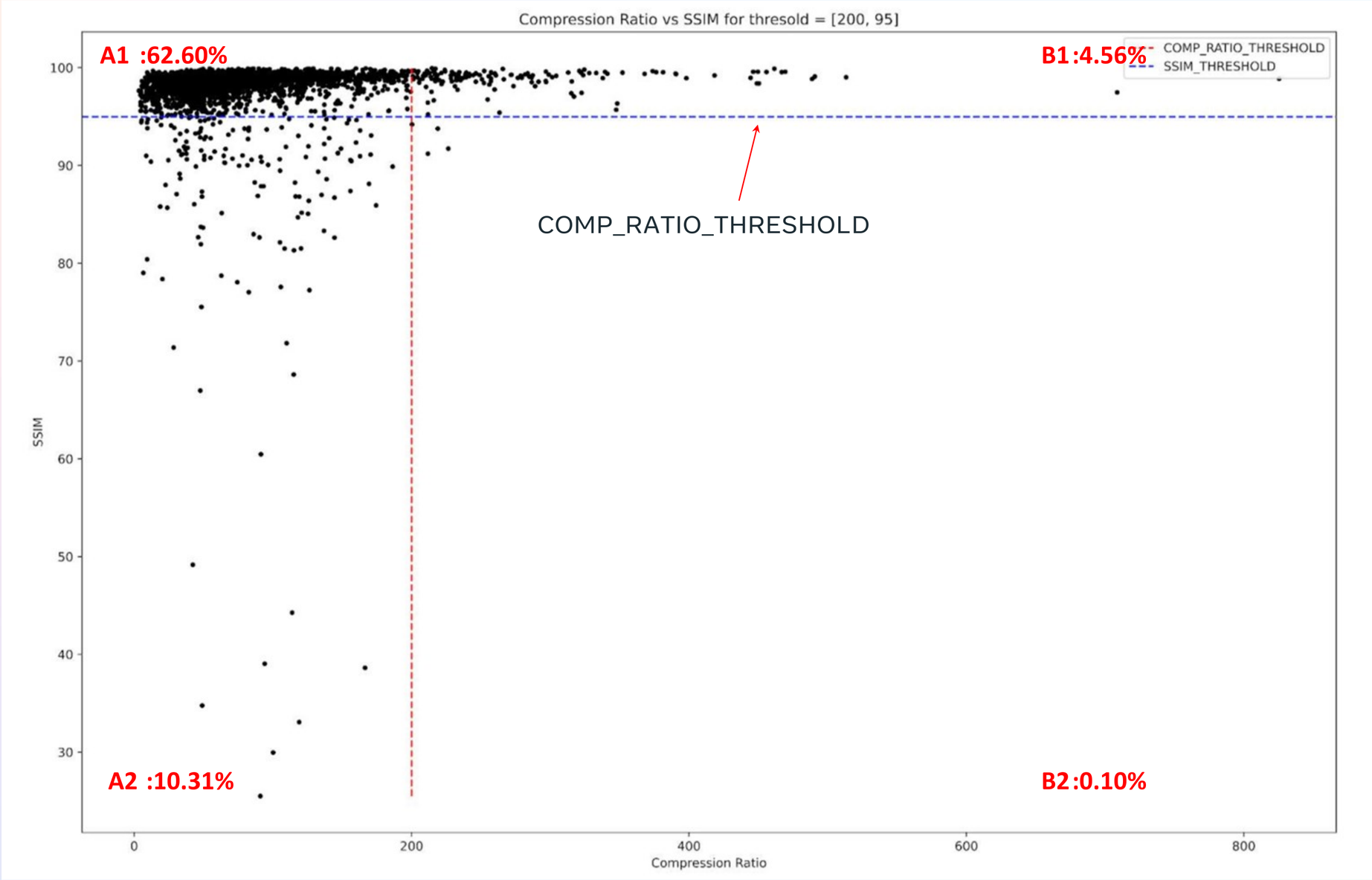
Eligibility Check Based on Initial Compression Ratio



$$Compression\ ratio = \frac{width \times height \times factor \times frame\_rate}{encoding\ bit\ rate}$$

A	video is recompressed and (filesize) is reduced.
B	recompression is skipped, bitrate saving opportunity is missed.
C	recompression is applied but filesize increased. Compute is wasted.
D	recompression is skipped.

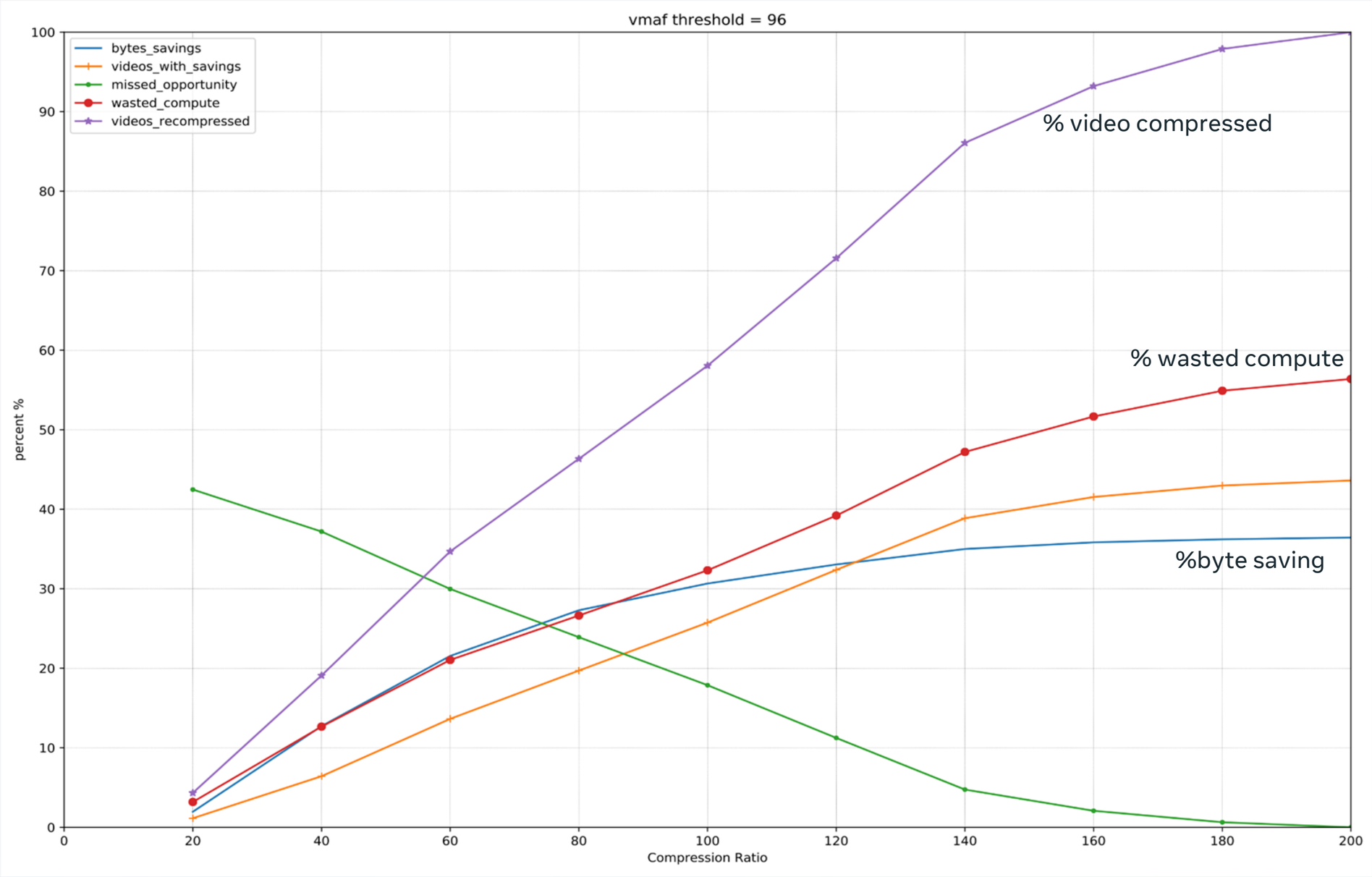
Quality Drop after Recompression



A, B portions can be further divided to A1, A2, B1, B2

A1	Quality is above the Q_THRESHOLD, real bitrate saving is achieved
A2	compute is wasted.
B1	opportunity missed.
B2	recompression is skipped.

Storage Saving Estimate



ML Based Predictor for Accurate Targeting

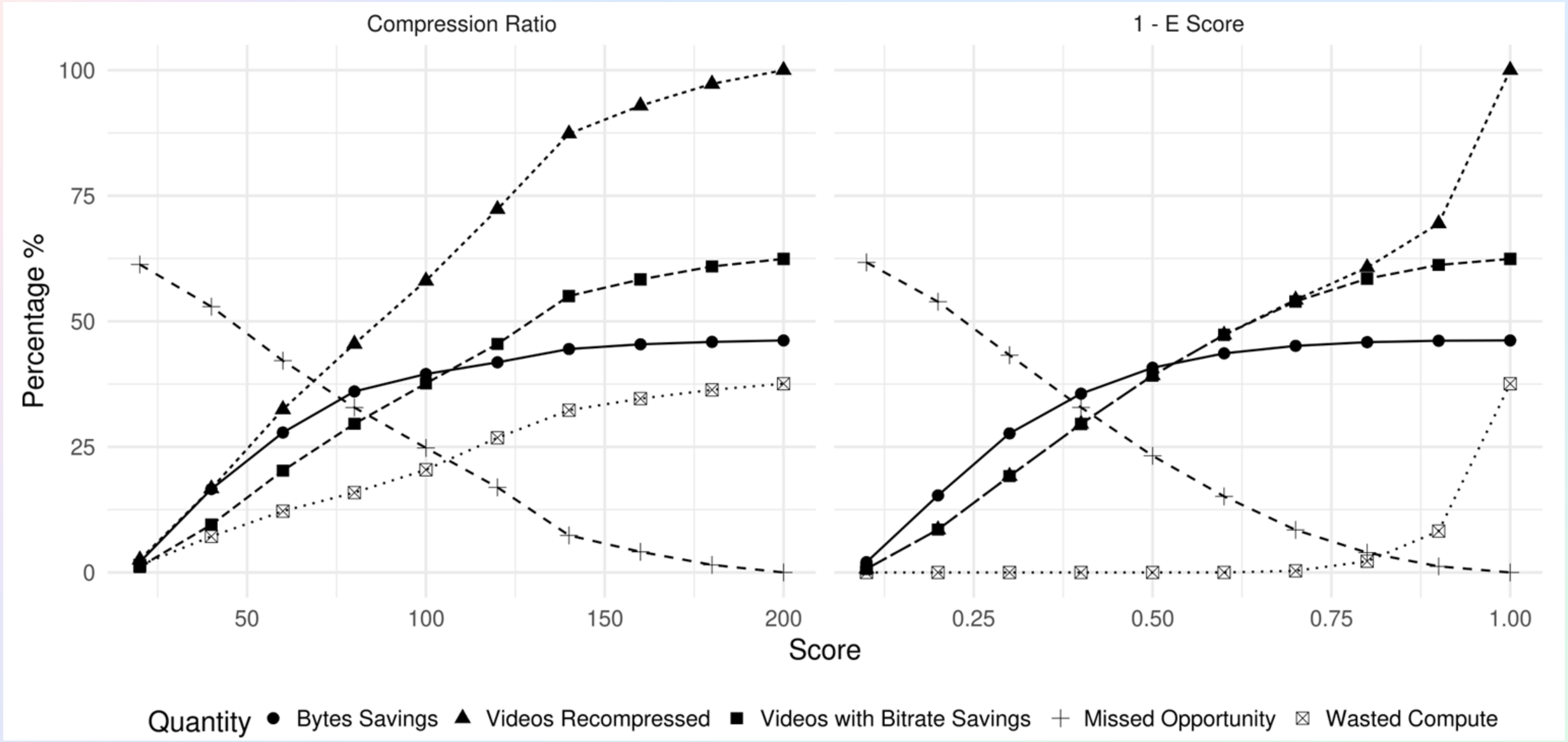
- Risk-Reward Model.
  - **reward model:** predicts the % expected bitrate savings from recompression (**S**).
  - **risk model:** predicts the probability that compute is wasted (**P**).
  - risk - reward score:

$$E = (1 - P) \times S \text{ in } [0, 1].$$

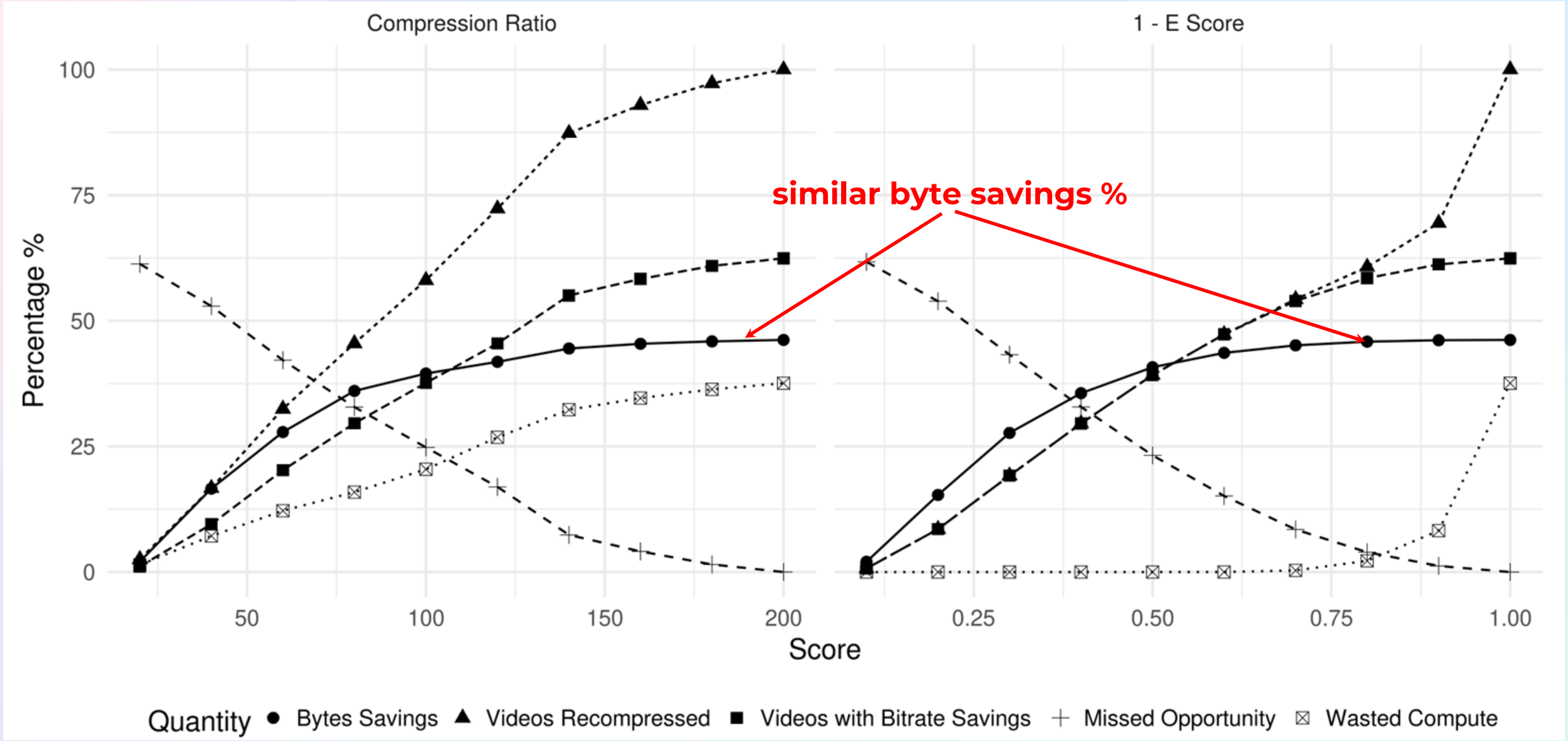
- More input features, such as resolution, duration, etc., are included.
- Multiple ML training methods experimented. Random Forest method produces the best result.

Target	Metric	Linear/logistic	Gradient Boosting	Random Forests
Bitrate savings (S)	RMSE / MAE	0.23 / 0.16	0.22 / 0.15	**0.20 / 0.13**
Wasted compute (P)	F1 Score / AUC	0.56 / 0.56	0.70 / 0. 66	**0.79 / 0.77**

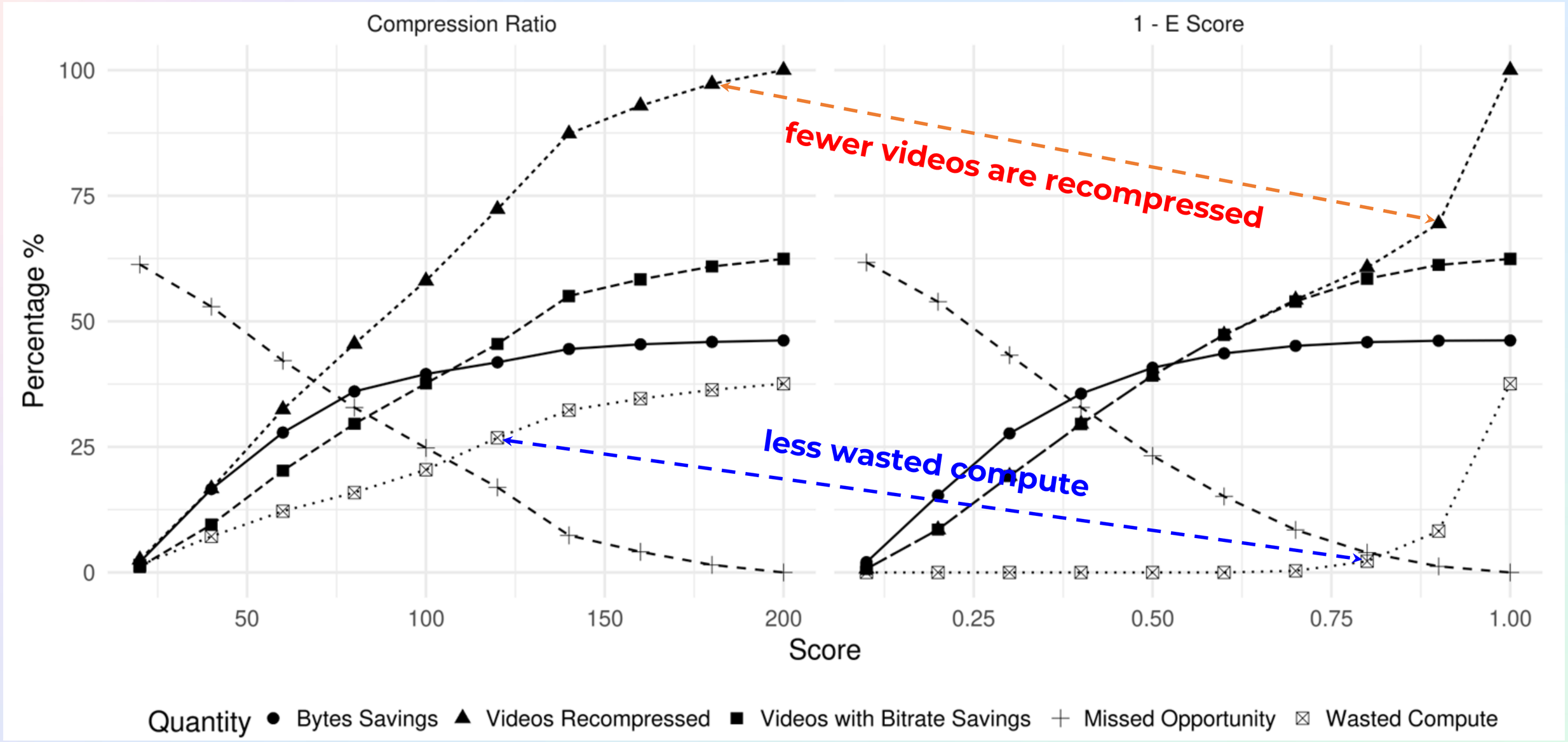
Performance Comparison



Performance Comparison



Performance Comparison



Conclusion

- Storage efficiency is an key area for improvement at Meta
- Source recompression is an important tool to reduce storage cost
- Accurate ML based predictor can help reduce the compute cost

Thank You!

Ryan Lei: ryanlei@meta.com
Klaus Schneider klausschneider@meta.com

https://www.linkedin.com/in/ryanlei/
https://www.linkedin.com/in/schneiderklaus/


Meta