AI Model Management

Michael McCool, Geoff Gustafson,
Sudeep Divakaran, Muthaiah Venkatachalam

Intel

23 September 2024, TPAC 2024

Summary for WebML Meeting

Agenda

- Breakout
- Issues
- Alternatives
  - Focus on “Common Model Auto-Expedite” alternative
- Next Steps (5m)

AI Model Management Breakout

https://github.com/w3c/tpac2024-breakouts/issues/15

Wednesday, 13:15–14:15 Pacific, 2 Ballroom Level - California A

Agenda

- Review list of issues and add or refine any if necessary
- Prioritize issues, identify shortlist for discussion
- Discuss potential solutions to high-priority issues

Issue Starter Pack

AI Model Management · Issue #15 · w3c/tpac2024-breakouts

- Background model download and compilation.
- Model naming and versioning
- Allowing for model substitution when useful
- Common interface for downloadable and “platform” models
- Storage deduplication
- Model representation independence
- API independence (e.g. sharing between WebNN and WebGPU)
- Browser independence
- Offline usage, including interaction with PWAs
- Cache transparency (e.g. automatic or explicit checking)

Alternatives

1. Do nothing.
2. Do the minimum.
3. Enhance existing caches.
4. Define model-aware caches.
5. Auto-expedite common models.

Alternative 5: Auto-Expedite Common Models

The more common a model is, the less of a tracking risk it is

- Low probability models carry high information
  - **Restrict low probability models to single-origin caching**
- “Built-in” models are “certain” and carry zero information
  - 100% probability given browser+version, which is already known
  - **Would act like they are “preloaded” in the shared cache; load immediately**
- Models with in-between probabilities “budgeted”
  - **Automatically load models from shared cache up to maximum “information budget”**
  - If information budget would be exceeded for a given probe, gracefully degrade to user-expedite prompts (large models) or per-origin caches (small models)
  - ***Note that information budget check needs to happen BEFORE probe is confirmed.***

Next Steps

- Organize community to address problem
  - Obtain consensus on solution alternative
  - Further discussion of alternatives probably needed
  - Do any of the alternatives need standardization?
- Identify standards gaps, for example:
  - Foundation models (shared) + adapters (per-origin)
  - Hybrid model APIs
  - Shared weight representations for WebGPU/WebNN implementations
    - MLTensor is (currently) only for the inputs and outputs of models…

Backup/Extra

References and Links

- Storage Partitioning (see HTTP Caches especially)
- GPU Web Privacy Considerations (shader caches)
- Felten and Schneider, Timing Attacks on Web Privacy, 2000
- Judis, Say goodbye to resource-caching across sites and domains, 2020
- CloudFlare (CDN) Origin Cache Control (can also be enabled in CDNs)
- Background Fetch – related API for large downloads.
- Cache AI models in the browser (Google) – how to use existing per-origin cache mechanisms for AI models
- https://github.com/webmachinelearning/proposals/issues/5
- https://github.com/webmachinelearning/hybrid-ai
- https://github.com/w3c/tpac2024-breakouts/issues/15
- Choose model · Issue #8 · explainers-by-googlers/prompt-api (github.com)

More References and Links

- Fingerprinting:
  - https://coveryourtracks.eff.org/
  - https://amiunique.org/
  - https://blog.amiunique.org/an-explicative-article-on-drawnapart-a-gpu-fingerprinting-technique/ (paper at https://inria.hal.science/hal-03526240/document) - can distinguish identical GPUs via WebGL
- Privacy budgets (pros, cons)
  - https://developers.google.com/privacy-sandbox/protections/privacy-budget
  - https://blog.mozilla.org/en/mozilla/google-privacy-budget-analysis/
    - https://mozilla.github.io/ppa-docs/privacy-budget.pdf (details)
  - Brave, Fingerprinting, and Privacy Budgets | Brave
- Privacy-preserving aggregation using modular arithmetic
  - Privacy-preserving measurement and machine learning (cloudflare.com)
  - Prio: Private, Robust, and Scalable Computation of Aggregate Statistics
  - https://datatracker.ietf.org/doc/draft-ietf-ppm-dap/