youtube-learning v2 — FORGE Pipeline

youtube-learning v2 — FORGE Pipeline **Status:** Active (side-by-side with v1) **Author:** ALAI, 2026 **MC Ref:** #9908, #9918, #9919, #9920, #9922 --- ## 1. Pregled / Overview youtube-learning v2 replaces single-pass Ollama summarization with a 3-pass FORGE-routed extraction pipeline that produces implement-ready dossiers per video. Instead of 498-character bullet summaries, v2 generates structured JSON with hardware specs, CLI commands, costs, gotchas, key numbers, code snippets, Q&A pairs — plus full transcripts indexed into LightRAG knowledge graph and ALAI relevance scoring for draft MC task generation. The pipeline routes inference through the FORGE tier router (localhost:8400) with automatic circuit breaking, tier escalation, and per-pass telemetry logging to routing-outcomes.db. All processing is local ($0 constraint), batched at ≤10 videos/min to respect LightRAG backpressure, with semaphore enforcement to serialize video processing. --- ## 2. Arhitektura / Architecture ```mermaid flowchart LR A[YouTube URL] --> B[yt-dlp fetch transcript] B --> C{Acquire Lock
/tmp/youtube-v2.lock} C --> D1[Pass 1: TLDR
tier:1 llama3.1:8b] D1 --> D2[Pass 2: Extract
tier:2 qwen2.5-coder:32b
chunked] D2 --> D3[Pass 3: ALAI Relevance
4D formula local] D3 --> E[LightRAG Ingest
POST /documents/texts
transcript + JSON] E --> F[HiveMind Intel
summary post] F --> G[Release Lock] G --> H{score >= 7?} H -->|Yes| I[Draft MC JSON
/tmp/youtube-actionable/] H -->|No| J[Complete] I --> J D1 -.-> R[FORGE Router
localhost:8400] D2 -.-> R R -.-> ANVIL[ANVIL
llama3.1:8b
qwen2.5-coder:32b] R -.-> FORGE[FORGE
qwen3:32b
circuit:open] D1 -.log.-> DB[(routing-outcomes.db)] D2 -.log.-> DB D3 -.log.-> DB E -.checkpoint.-> SQLITE[(youtube-lightrag-ingest.sqlite)] ``` --- ## 3. Tier Routing Odluke / Tier Routing Decisions | Pass | task_type | tier | model | typical latency | rationale | |------|-----------|------|-------|-----------------|-----------| | Pass 1 TLDR | youtube-tldr | T1 | llama3.1:8b | 8-10s | Fast 3-sentence summary for HiveMind post and Pass 3 input. ANVIL at 181 tok/s. | | Pass 2 Extract | youtube-extract | T2 | qwen2.5-coder:32b | 30-75s per chunk | Structured JSON extraction (7 required keys). Long-pole pass. ANVIL at 28 tok/s. Escalates to T3 qwen3:32b when FORGE circuit closes. | | Pass 3 Relevance | youtube-alai-relevance | local | N/A | <1s | 4D scoring formula (KW 30% + TS 25% + PG 30% + DP 15%) against 8 ALAI projects. Runs locally without LLM. | **Circuit state (2026-04-28):** FORGE circuit=open (MC #9916), all T2/T3 requests fall back to ANVIL. T1 always ANVIL. When FORGE TCP-refused issue resolves, T2 escalates to T3 qwen3:32b automatically. --- ## 4. Modulna Mapa / Module Map | File | Purpose | |------|---------| | `~/system/tools/youtube-learning-v2.js` | Main pipeline — orchestrates 3 passes, lock/unlock, routing-outcomes logging. | | `~/system/tools/lib/youtube-lightrag-ingest.js` | LightRAG batch insert + SQLite checkpoint dedup. Fire-and-forget POST

/documents/texts. | | `~/system/tools/lib/alai-relevance.js` | 4D scoring formula, draft MC generator, topic cluster dedup, guardrails (weekly cap, triage freeze). | | `~/system/tools/youtube-actionable-digest.js` | Weekly digest CLI: `node youtube-actionable-digest.js --since 7d` → /tmp/youtube-digest-YYYY-MM-DD.md | | `~/system/tools/youtube-learning.js` | v1 pipeline (unchanged, still functional for fallback). | --- ## 5. Stanje i Idempotencija / State & Idempotency **v1 compatibility:** - `~/system/logs/youtube-batch-state.json` — shared state file, tracks processed video IDs. v2 writes to same file. - Format unchanged: `{videos: {: {status:'done', processed_at:, ... }}}` **v2 checkpoint dedup:** - `~/system/state/youtube-lightrag-ingest.sqlite` — table: `ingest_log(video_id PRIMARY KEY, ingested_at, transcript_doc_id, json_doc_id, status)` - Dedup window: 30 days. If `status='success'` and `ingested_at` within 30d, skip LightRAG insert. - `--force-rerun` flag bypasses both youtube-batch-state.json and LightRAG checkpoint. --- ## 6. Failure Modes / Načini Otkazivanja | Scenario | Behavior | Recovery | |----------|----------|----------| | FORGE circuit open (current) | Router falls back to ANVIL for T2/T3. All passes run on ANVIL. Pass 2 latency 30-75s/chunk. | Automatic when MC #9916 resolves. No code change needed. | | Router unavailable (localhost:8400 down) | Client-side circuit opens after 3 failures. Video marked failed, retry next batch. No silent fallback to direct Ollama. | Restart FORGE router: `docker restart forge-router` (ANVIL) or resolve networking. | | Pass 2 timeout (>480s per chunk) | Log error to routing-outcomes.db with error field populated. Skip chunk, continue with remaining chunks. If ALL chunks timeout, return null, mark video failed. | Escalate chunk tier to T3 (when FORGE circuit closes) or increase timeout in code if transcript is unusually large. | | Pass 3 relevance fails | Set `alai_relevance = {score:5, tags:[], mc_priority:'M', rationale:'relevance-unavailable'}`. Pass 1+2 results preserved, video still indexed. | Non-blocking — LightRAG and HiveMind posts succeed regardless. | | LightRAG HTTP 429 or timeout >30s | Mark `status='backpressure'` in checkpoint. Retry on next batch run. No spin loop. | Wait for LightRAG pipeline to drain (check /documents/pipeline_status). Current queue: 119k pending, 4-6 docs/min processing. | | HiveMind socket hang up | Pre-existing issue on qdrant RAG path. LightRAG ingest succeeds, HiveMind post may fail without impact. | Document only — does not block pipeline. | | malformed JSON in Pass 2 | 3-retry budget with stricter prompt (`buildStricterExtractionPrompt()`). If all 3 fail, log parse error, skip chunk. | Check `routing-outcomes.db` error column for "malformed JSON" entries. Escalate to tier T3 if model quality issue. | **FORGE 10.0.0.2 TCP-refused:** Currently down from Mac (MC #9916). Router → ANVIL → FORGE path works. All v2 passes route through ANVIL until network issue resolves. **LightRAG queue depth:** 119,378 docs pending as of 2026-04-28. Query results may be empty for newly ingested videos until background indexing completes. Verify via /documents endpoint and SQLite checkpoint, NOT query response. This is NOT a defect — expected behavior during mass migration. --- ## 7. ALAI Relevance Skoring / ALAI Relevance Scoring **4D Formula (per project, 0-10):** ``` score = round((KW * 0.30) + (TS * 0.25) + (PG * 0.30) + (DP * 0.15) , 1) ``` | Dimension | Weight | Description | |-----------|--------|-------------| | Keyword Overlap (KW) | 30% | Count of project keywords hit in transcript/title/tags, normalized 0-10. | | Tech Stack Overlap (TS) | 25% | Count of tech stack terms hit (from MEMORY-products.md), normalized 0-10. | | Priority Gate (PG) | 30% | CEO priority tier: FOCUS (Bilko/Tok/Drop/Lobby) = 10, ACTIVE = 7, RESEARCH = 5, DEPRIORITIZED (LumisCare) = 3. | | Depth Signal (DP) | 15% | Duration proxy: >=45min=10, 20-44min=7, 10-19min=5, 5-9min=3, <5min=1. | **LumisCare hard-cap:** Max score 3 regardless of keyword/tech match (CEO decision 2026-04-17). **Draft MC threshold:** `score >= 7.0` AND `duration >= 600s`. Drafts written to `/tmp/youtube-actionable/.json` with full reasoning, specialist routing from `specialist-mapping.json`, and suggested action. John reviews manually — no auto-creation of live MC tasks. **Safety guardrails:** - Weekly cap: max 10 drafts per 7-day rolling window - Triage freeze: max 3 drafts/day during TRIAGE period (until 2026-05-02)

- Topic cluster dedup: cosine similarity >0.85 on suggested-action text (via bge-m3 embeddings) = skip - Channel dedup: max 2 drafts per channel per month **Score calibration note (V1 finding):** Hardware/infra content (e.g., GB10 cluster video) scores lower than expected — AgentForge 3.5, HOP 2.9 on canary run. Expected range for GPU-infra: 3-5. Fintech tutorials (PSD2/banking APIs): 7-9 on Tok/Drop. Calibration follow-up tracked as MC #9925. --- ## 8. CLI Commands Edge Case **Finding from V1 canary validation (MC #9922):** The `cli_commands` array in Pass 2 JSON is **empty for non-tutorial videos** (e.g., hardware walkthroughs, conference talks, product demos). This is **CORRECT behavior** — the model is non-hallucinating. qwen2.5-coder:32b extracts actual shell commands from transcripts, not mentions of commands or operational guidance. **Example:** GB10 cluster video (uYepcMoqvKQ) returned: - `hardware_specs`: ✓ (8x GB10, RDMA, 160 ARM cores) - `costs`: ✓ ($23k setup, $100/mo Cloud Code) - `gotchas`: ✓ (4 entries) - `key_numbers`: ✓ (5 distinct numbers) - `cli_commands`: [] (empty — no shell commands in transcript) **Do NOT file bug reports for empty `cli_commands` on hardware/demo videos.** Check transcript content first. Tutorial videos (setup guides, how-tos) populate this field richly. --- ## 9. Ops Runbook Delta / Operativni Runbook Dodatak ### Inspect routing outcomes (last 20 passes) ```bash sqlite3 ~/system/databases/routing-outcomes.db "SELECT task_type, tier, model, host, latency_ms FROM routing_outcomes ORDER BY created_at DESC LIMIT 20" ``` **Note:** Table name is `routing_outcomes`, not `outcomes` (correction from V1 finding). ### Clear v2 dedup checkpoint (force re-run) ```bash sqlite3 ~/system/state/youtube-lightrag-ingest.sqlite "DELETE FROM ingest_log WHERE video_id='" ``` ### Force re-run a video (bypass state.json + LightRAG checkpoint) ```bash node ~/system/tools/youtube-learning-v2.js --video --force-rerun ``` ### Check LightRAG queue health ```bash curl -s http://localhost:9621/documents/pipeline_status | jq '{busy, docs, cur_batch, batchs, latest_message}' ``` **Expected during mass migration:** `busy: true`, `docs: 119k+`. New inserts join pending queue. ### Verify video landed in LightRAG (post-ingest) ```bash # 1. Check SQLite checkpoint sqlite3 ~/system/state/youtube-lightrag-ingest.sqlite "SELECT video_id, status, ingested_at FROM ingest_log WHERE video_id='" # 2. Check entity exists in graph (after indexing completes) curl -s "http://localhost:9621/graph/entity/exists?name=" # 3. Query for transcript doc (hybrid mode) curl -s -X POST http://localhost:9621/query \ -H "Content-Type: application/json" \ -d '{"query":"","mode":"hybrid","top_k":10}' | jq ``` ### Disable v2 cutover (revert to v1-only) **Current state:** Both v1 and v2 callable. LaunchAgent `com.john.youtube-nightly-learning` still calls v1. **To cutover:** Update LaunchAgent plist: ```bash # Edit: ~/Library/LaunchAgents/com.john.youtube-nightly-learning.plist # Change ProgramArguments from youtube-learning.js to youtube-learning-v2.js launchctl unload ~/Library/LaunchAgents/com.john.youtube-nightly-learning.plist launchctl load ~/Library/LaunchAgents/com.john.youtube-nightly-learning.plist ``` **Cutover gate:** ALAI calibration (MC #9925) closed AND 7 consecutive nightly batches with ≥90% Pass-2 JSON depth pass rate. ### LightRAG health timeout config Health check timeout must be ≥45s under qwen3:8b load. Insert timeout: 30s (fire-and-forget). ```bash # Check health (NOT a gate — informational only) curl -s --connect-timeout 45 http://localhost:9621/health | jq ``` --- ## 10. v1 → v2 Cutover Plan / Plan Prelaska **Current state (2026-04-28):** Both pipelines operational. v1 serves nightly batch. v2 callable via CLI with `--video` flag. **Cutover conditions (ALL must be met):** 1. MC #9925 (ALAI calibration follow-up) CLOSED — score ranges validated for fintech/hardware/AI content types 2. 7 consecutive nightly batches with ≥90% Pass-2 JSON depth pass (all 7 required keys present) 3. Pressure test complete with 0 crashes (50-video batch at ≤10/min) 4. BookStack documentation published (this page) 5. John approval after manual review of 5 sample drafts from `/tmp/youtube-actionable/` **Cutover steps:** 1. Update LaunchAgent plist (see §9 above) 2. Run first nightly batch via v2 in --preview mode (no MC drafts, verify output only)

3. Monitor routing-outcomes.db for error spikes 4. Enable draft MC generation after 3 clean batches 5. Archive v1 → `youtube-learning-v1-legacy.js` (keep for rollback, do not delete) **Rollback procedure:** ```bash # Revert LaunchAgent plist to youtube-learning.js launchctl unload ~/Library/LaunchAgents/com.john.youtube-nightly-learning.plist # Edit plist back to v1 launchctl load ~/Library/LaunchAgents/com.john.youtube-nightly-learning.plist ``` v1 state.json and HiveMind schema unchanged — rollback is instant. --- ## 11. Reference / Reference **Spec file:** `~/system/specs/youtube-learning-v2-plan.md` **MC tasks:** - #9908 (parent, H-priority) - #9918 (B1 build — youtube-learning-v2.js) - #9919 (B2 build — youtube-lightrag-ingest.js) - #9920 (B3 build — alai-relevance.js + digest CLI) - #9922 (V1 validation — canary report) - #9924 (D1 documentation — this page) - #9925 (calibration follow-up — ALAI score ranges per content type) - #9916 (FORGE TCP-refused network issue — M-priority) **FORGE router endpoint:** `http://localhost:8400/api/generate` **LightRAG endpoint:** `http://localhost:9621` **Routing outcomes DB:** `~/system/databases/routing-outcomes.db` **LightRAG checkpoint DB:** `~/system/state/youtube-lightrag-ingest.sqlite` **Draft MC directory:** `/tmp/youtube-actionable/` **Digest output:** `/tmp/youtube-digest-.md` --- **Document Version:** 1.0 **Last Updated:** 2026-04-28 **Status:** Active — side-by-side with v1, cutover gated per §10

Revision #2
Created 2026-04-28 07:18:25 UTC by John
Updated 2026-05-31 20:06:40 UTC by John