

Technical report — Sovereign Market Brief

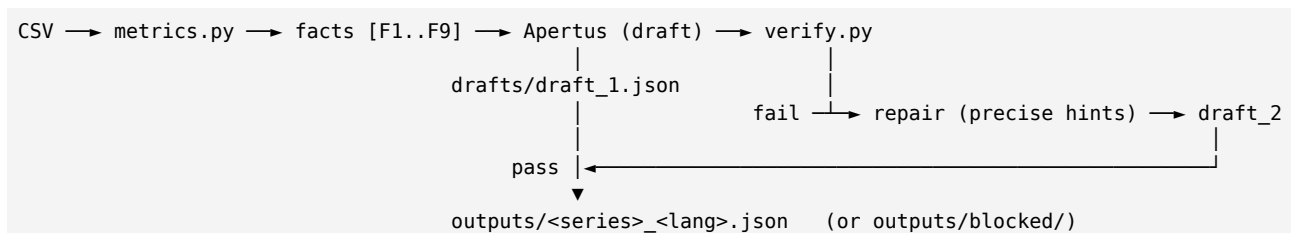
- **Track:** Track 2B — Sovereign Market Brief
- **Event:** Online
- **Team:** SimonLin — Simon Lin
- **Demo:** <https://github.com/simonlin1212/sovereign-market-brief/releases/download/v1.0/demo.mp4> (1:26)

1. Summary

Financial teams need short multilingual market notes, but LLMs invent numbers and usually run on foreign cloud APIs. Sovereign Market Brief lets **Apertus** write the prose while Python owns every number: metrics become numbered facts, Apertus must cite them, and a deterministic verifier checks numbers, direction words, dates, high/low order, required coverage and the disclaimer. Failed briefs are blocked (not published); a repair call gets precise hints (e.g. “265 should cite [F9]”). Every draft is saved.

Results (Apertus v1.5 8B, Q4_K_M GGUF, CPU, final run 2026-10-15): 15 briefs (3 FX series × 5 languages): 7 passed on the first draft, 3 after one repair, **10 published, 5 blocked**. All numbers and dates in the 10 published briefs were checked by hand against the facts: no numeric or date error found. Unconstrained baseline (same model, raw CSV, no facts/verifier): 12 of 16 numbers matched no true value (EN), 3 of 8 (DE).

2. Architecture



Components: `metrics.py` (stdlib), `llm.py` (OpenAI-compatible; `LLM_NAME` / `LLM_BASE_URL` / `LLM_API_KEY`), `verify.py`, `brief.py` (≤ 2 calls; exit 0 with summary), `baseline.py` (ablation). `make run` starts a pinned llama.cpp Docker image plus the app.

Target architecture (mandatory)

(a) On-premise and **(b) air-gapped** after the one-time download of Docker images and GGUF weights. Runtime traffic is only app ↔ local llama.cpp. Pointing `LLM_BASE_URL` at a Swiss-hosted endpoint would also allow (c); not tested here.

3. Use of Apertus

- **Base model:** `swiss-ai/Apertus-v1.5-8B`.
- **Text-only conversion** used for tooling compatibility: `andreamartin/apertus-v1.5-8b-text` by **Prof. Andreas Martin (FHNW)** (<https://huggingface.co/andreamartin/apertus-v1.5-8b-text>).
- **Quantisation we run:** Q4_K_M GGUF by community user *Colby* (`Colby/apertus-v1.5-8b-text-Q4_K_M-GGUF`), SHA-256 `a037df8d87ff6caacee794ee85f55342f2152e0d359b7389033300c3bee5f299`.
- **Serving:** llama.cpp server-b11610 (`ghcr.io/ggml-org/llama.cpp:server-b11610@sha256:89f180f16...`), CPU, `--jinja`.
- **How it is used:** inference (constrained multilingual generation). No other model is used.
- **Where it runs:** local weights (GGUF) served by llama.cpp on CPU — directly on the host for \$5.1, and inside Docker via `make run` (\$6).

4. Data

ECB euro FX reference rates (EXR.D.{CHF,USD,CNY}.EUR.SP00.A), 2025-10-01 → 2026-10-14, 265 daily observations each. Source: ECB Data Portal. No personal data. data/ < 100 MB.

5. Evaluation

Hardware: shared 8-vCPU Linux VM, no GPU. Unit tests: `python3 -m unittest discover -s tests` (58 cases: numbers, direction in EN/DE/FR/IT/ZH incl. both directions in one clause, written and detached dates, high/low labels and dates, metric/horizon misuse incl. “return and volatility were both X”, double negatives, wrong German terms, value-based coverage, negated disclaimer, Unicode hyphens, unsigned negative changes, wrong currency, repair hints). The verifier binds each number to the fact it matches and checks the direction / high-low / metric / horizon / date words in the same clause.

5.1 With facts + verifier (this system)

All 15 briefs were generated with the verifier in `src/verify.py`. The run was resumed once after the `llama.cpp` server process was killed (kernel out-of-memory log on the shared machine); finished briefs were kept. A last review then added two rules (unsigned negative change, wrong currency) and fixed FR/DE/IT direction words; re-checking all 15 saved briefs with that verifier changed only USD-IT (“la variazione è stata del 3,1 %” for −3.1 %), which was therefore regenerated with the final verifier and now passes (“è diminuito del 3,1 %”). The numbers below are the final state; `python3 src/reverify.py` reproduces 10 pass / 5 fail.

Setup	Briefs	Published (1st draft)	Published after repair	Blocked
Apertus 8B Q4_K_M CPU	15	7	3	5

Blocked (from `outputs/blocked/`), all checked by hand — each block is a real defect: CHF-EN — latest rate given the low's date (2026-03-09 instead of 2026-10-14); CHF-FR — repair draft echoed the hint list instead of a brief; CHF-IT — the 1-month value 0.9431 written as a “0,9431 %” decline and no change percentage; USD-ZH — no change percentage and “日志收益率” (mistranslation of log returns); CNY-DE — volatility [F5] omitted.

Wall-clock: 23 LLM calls, median 43 s, range 27–477 s (slowest under heavy shared load).

5.2 Baseline (no facts, no verifier)

Same model; head+tail of the CSV only; asks for the same statistics. We count extracted numbers that do not match the value of **any** fact ($\pm$ tolerance). This is a lower bound on errors: a true value attached to the wrong metric or date is not counted.

Lang	Numbers checked	Matched no fact	Share
EN (CHF)	16	12	75 %
DE (CHF)	8	3	38 %

Small sample (2 briefs); indicative only.

5.3 Manual review of the 10 published briefs

Every number, sign, date, high/low and unit was compared with the fact list: **no factual error found**. Wording issues the verifier does not judge:

1. CNY-IT: “la valuta ha subito una perdita ... deprezzamento” — does not say which currency (the CNY-per-EUR rate fell, i.e. the euro weakened against the yuan).
2. USD-FR / CNY-FR: volatility given without “annualisée”.
3. CHF-DE: states only the 1-month change (3- and 12-month omitted); coverage requires one.

Earlier development runs showed double negation (“declined by −3.08 %”), wrong German terms (“Log-Rückgaben”), dropped disclaimers and wrong citations; these are now verifier rules.

6. Limitations

- Verifier is heuristic: lexicons are finite (e.g. “la valuta ha subito una perdita” is not checked for which currency), clause splitting is rule-based, and prose quality is not judged.

- Docker: **make run verified end-to-end** on the build machine (Docker 26.1.5, Compose 2.26.1): sha256 check OK, pinned llama.cpp:server-b11610@sha256:89f180f1... model container + app container, 15 briefs, summary printed, exit 0 (log: docs/docker_validation.log, outputs: docs/docker_run/). Two host fixes were needed on that shared machine, not repo changes: a stale iptables-legacy FORWARD DROP policy blocked container-to-container traffic, and RAM was short (swap added). The Docker run gave **5 published / 10 blocked** (all 5 CNY briefs passed, none on the first draft) versus 10 / 5 in §5.1 on the host llama.cpp build b11617 with the same code and verifier. We did not isolate the cause (different llama.cpp build and sampling run); blocked Docker briefs we inspected contained real errors (e.g. 3-month +0.8 % written as a decline, “Rückgang ... –6,66 %”), and the 5 published ones have no numeric error. So results vary between runs; in both runs every blocked brief we inspected had a real error and no published brief had a numeric or date error.
- 70B and Swiss-cloud endpoints not tested.

7. Reproducibility

```
cd track_2b
make run          # downloads GGUF once, sha256-checks, docker compose up (verified end-to-end)
# or without Docker:
llama-server -m models/apertus-v1.5-8b-q4_k_m.gguf --jinja -c 4096
export LLM_BASE_URL=http://127.0.0.1:8080/v1
python3 src/brief.py
python3 src/baseline.py --langs en,de
python3 -m unittest discover -s tests
```

8. Next steps

Terminology checks beyond the current lexicon; Swiss equities / filings as inputs; Apertus 70B on a Swiss endpoint; native-speaker review (DE/FR/IT).

License

Code: Apache-2.0. Documentation: CC-BY-4.0.

References

1. swiss-ai/Apertus-v1.5-8B — <https://huggingface.co/swiss-ai/Apertus-v1.5-8B>
2. Prof. Andreas Martin (FHNW), text-only conversion (profile: <https://www.fhnw.ch/en/business/about/portrait-organisation/people/andreas-martin>) — <https://huggingface.co/andreasmartin/apertus-v1.5-8b-text>
3. Colby Q4_K_M GGUF — https://huggingface.co/Colby/apertus-v1.5-8b-text-Q4_K_M-GGUF
4. llama.cpp — <https://github.com/ggml-org/llama.cpp>
5. ECB Data Portal — <https://data.ecb.europa.eu/>