GeaNaVid

Gea, Na or Vid: verifying claims against Swiss federal voting booklets

Niklas Wichter

Team Apertool · Hack Apertus 2026 · Track 2A — OST: Multilingual NLI over Swiss Voting Booklets

- **Track:** Track 2A — OST: Multilingual NLI over Swiss Official Voting Booklets
- **Event:** Online
- **Team:** Apertool — Niklas Wichter
- **Demo:** https://geanavid.apertool.ch/#video · `make run` in `track_2a/`

State of 14 October 2026. All numbers are on OST v1.1 (v1.0 without the 7 faulty rows OST removed on 6 October) and come from runs stored in `runs/v1/`. Ablations, retrieval recall, calibration, error analysis and the full tables are in `docs/EXPERIMENTS-v1.md`; the improvement rounds, including the rejected ones, in `docs/ITERATIONS.md`.

1. Summary

Swiss voters meet many claims about federal proposals; the official voting booklet is the authoritative source, but it runs to 40–90 pages in three languages. GeaNaVid checks a claim in German, French or Italian against a booklet and returns entailment, neutral or contradiction, for entailment and contradiction together with the passages that decide the label, plus the token usage and the time. We treat the task as document-level natural language inference (Koreeda & Manning, 2021; Yin et al., 2021), close to claim verification with evidence (Thorne et al., 2018), but judged only against the booklet, never against the world.

Our CLI `booklint` parses the PDF without a language model, narrows the booklet to the proposal named in the case's `vote` field, retrieves a few passages with BM25, preferring the section the claim names ("the summary", "the committee", "the voting text"), and asks Apertus v1.5 for a strict JSON verdict with evidence first and label second. Cross-lingual claims are translated into the booklet language by Apertus before retrieval; a neutral answer Apertus is not fully sure of is checked once more, only for a contradiction. On the official OST v1.1 benchmark (493 test pairs of our split), Apertus 8B reaches a macro-F1 of 0.959 [95% CI 0.933–0.980, cluster bootstrap] with 2,159 input tokens per claim, against 0.720 for the whole-booklet baseline (20,242 input tokens). Apertus 70B reaches 0.965 [0.944–0.983] with 1,461 input tokens, against 0.889. With OST's own evidence rule (one of the first five quotes inside the gold passage), our quotes score 0.969 (8B) and 0.963 (70B) of the 324 gold entailment/contradiction cases. In the Beginner task (gold reference given), 8B reaches 0.977 and 70B 0.980. On 546 development claims outside the tuning subset, the defaults reach 0.971 / 0.971, and on a synthetic hold-out from booklets outside OST v1.1 the gap to the baseline carries over (8B 0.874 in its second evaluation, first 0.845, vs 0.665; 70B 0.932 vs 0.872). On older booklets (1998–2018, two synthetic stress sets, 802 claims) task A was 0.79 / 0.82; reading the proposal structure of older layouts raised it to 0.84 / 0.89 (8B / 70B; §4). The output is never a voting recommendation: every yes or no comes with a quote from the booklet; a neutral answer ("blank") cites nothing.

2. System design and architecture

The container follows OST's Solution API Guide (7 October): `<image> --input /data/cases.jsonl --output /output/predictions.jsonl`, endpoint and key from `BASE_URL` / `API_KEY`, booklet paths relative to `/data`. It reads OST's cases (`id`, `vote`, `claim{text, language}`, and `booklet{path, language}` for task A or `reference{text, language}` for task B; flat variants are accepted too) and writes one line per case with `id`, `label`, `label_name`, `evidence[{page, text}]` and `metrics{input_tokens, output_tokens, inference_time_ms}`. If the endpoint names the models differently (`swiss-ai/Apertus-v1.5-...`), the model of the same size is taken from its model list. The default pipeline for the Advanced task (A):

1. **Ingest** (once per booklet, no LLM, cached): text blocks with page, proposal, section and committee note.
2. **Languages** of claim and booklet from function words and letters; no language field needed.
3. **Proposal**: the booklet is narrowed to the proposal named by `vote` (BM25 over proposal titles and texts). A `vote` title in another language than the booklet is first translated by Apertus.
4. **Translation**: a cross-lingual claim is translated into the booklet language by Apertus.
5. **Retrieval**: BM25 top-5 over claim and translation inside the proposal, plus same-page neighbours; blocks of the section the claim names (summary, committee, voting text, Federal Council) rank higher.
6. **Apertus NLI**, one call: numbered passages, JSON schema with evidence before label. With 8B the prompt carries three development examples, a verdict with p(label) < 0.9 is re-checked by splitting the claim into atomic statements (two more calls), and a neutral verdict with p(label) < 0.999 is re-checked once with a narrow question: does a passage contradict one detail of the claim (number, actor, position, direction, scope)? With 70B the passage headers are compact (proposal title once, section type instead of the page note). For a cross-lingual committee claim, "Ausschuss" / "commission" in the translation is set back to the booklet's "Komitee" / "comité" / "comitato".
7. **Evidence sentences** (no LLM): up to five verbatim sentences from different passages (cited, retrieved or of the named section), best first, each with its 1-based PDF page.

For the Beginner task (B: claim + reference text) steps 3 and 5 are skipped: the reference is the context, and evidence carries `"page": null` as OST asks. A failed case (missing or scanned PDF, endpoint error) is answered neutral with an `error` field, so a batch never stops.

2.1 Document parsing and context preparation

Parsing. PyMuPDF text blocks are cleaned (soft hyphens, line-break hyphenation, footnote digits that otherwise glue to numbers). Each block keeps its page (1-based PDF index), the proposal from the running header, the nearest bold side heading ("In Kürze", "Empfehlung von Bundesrat und Parlament") and the committee note. Parsing is deterministic and costs no tokens. Booklets without a text layer are reported as failed cases (no OCR yet).

Context preparation. (a) Locating the proposal by the `vote` title keeps 93.7 % of the gold section inside about a quarter of the booklet (median 19 k of 82 k characters). (b) BM25 (Robertson & Zaragoza, 2009) over heading plus text, with number normalisation and prefix stemming, fused across claim and translation by reciprocal rank (Cormack et al., 2009); BM25 needs no GPU and is a strong zero-shot baseline (Thakur et al., 2021). (c) **Sections.** Every block gets a section type from its headings and page notes (legal text, committee, Federal Council arguments, recommendation, explanation; DE/FR/IT keywords), and the claim's wording tells which source it names. On the 998 development claims the gold section contains the explanation part for 99 % of "summary" claims, the committee pages for 93 % of committee claims and the recommendation or Federal Council arguments for 98 % of Federal Council claims. Blocks of the named section get a factor 1.5 in the rank fusion. hit@5 of the gold section is 0.980 on test with translation and sections (0.928 without sections, 0.718 without translation, 0.604 over the whole booklet). (d) The five hits and their same-page neighbours, each prefixed with page, proposal and section, give about 1.1 k tokens of context instead of 20 k for the whole booklet (p95 30 k).

NLI call and evidence. Apertus names the deciding passage numbers before the label, as in evidence-first setups (Koreeda & Manning, 2021; Thorne et al., 2018). The returned evidence is up to five verbatim booklet sentences from different passages, chosen after the decision without an LLM call (idf-weighted word stems and character 5-grams against claim and translation; cited passages ×1.2, sentences of the named section ×2, and its four best blocks are candidates too); `--evidence-unit paragraph` returns whole text blocks. OST scores the first five quotes and counts a case if one lies in the gold passage; five sentences instead of at most two left the first quote unchanged and were chosen on development data (§4).

Not in the default. A page-window ladder (Li et al., 2024), an 8B → 70B cascade (Chen et al., 2023), the parallel language version instead of translation, dense or hybrid retrieval, Apertus reranking, label calibration, number rules, compact passage headers for 8B, fewer few-shot examples, fewer passages without neighbours, splitting more claims ($p < 0.99$), re-checking entailments for a contradiction, translating the passages instead of the claim and a sharper contradiction/neutral definition in the prompt are flags or were dropped; on development data none paid for its cost. The ungated neutral re-check did not pay on test for 70B; the shipped 70B default uses it only below $p < 0.999$ (§4; `docs/EXPERIMENTS-v1.md`, `docs/ITERATIONS.md`).

3. Use of Apertus

- **Model:** swiss-ai/Apertus-v1.5-8B and swiss-ai/Apertus-v1.5-70B (Project Apertus et al., 2025)
- **How it is used:** inference only: the NLI verdict with evidence selection, the translation of cross-lingual claims and of foreign `vote` titles, and (8B) the split of unsure claims into atomic statements and the contradiction re-check of unsure neutral verdicts.
- **Where it runs:** any OpenAI-compatible Apertus endpoint: `BASE_URL` and `API_KEY` (as OST injects them; `LLM_BASE_URL` / `LLM_API_KEY` for our own runs), model `LLM_NAME`; temperature 0, JSON schema, `logprobs` with 10 alternatives.

The label probability is read from the first token of the label value. No other model takes part in the default path. Other models only supported development and evaluation data, never the pipeline: Claude (Anthropic) drafted the claims of the synthetic hold-out as a development aid (a closed model, used only to write evaluation data); only open-weight models judged labels blindly: Nemotron 3.5 Lightning and a panel of GLM-5.3, DeepSeek V3.2 and Qwen3-235B (hold-out), Gemma 4 31B and Qwen3-235B (stress sets, written by Nemotron 3 Ultra); none of them runs in the pipeline or produces a reported OST number. Every call is cached on disk by a hash of model, messages and parameters, which makes reruns free and reproducible.

4. Evaluation

Data. OST v1.1 (MIT, `OSTswiss/MNLIoverSwissVotingBooklets`) has 1,488 claim-booklet pairs, 60 booklets from 20 voting dates (2020–2026), balanced labels and about two thirds cross-lingual; it is v1.0 without 7 rows whose vote text was missing (removed by OST on 6 October; `data/v1/v1.1-removed.json`). We split v1.0 by voting date (seed 2026), so the DE/FR/IT versions of a booklet never cross splits: 998 development pairs (13 dates; 995 in v1.1), of which a stratified subset of 450 (`dev_s`, 449 in v1.1) was used for all ablations, and 497 test pairs (7 dates, 21 booklets; 493 in v1.1). Dropping the removed rows changes no prediction, so all v1.1 numbers are recomputed from the stored runs without new calls (`experiments/v1/replay_v11.py`).

How often test was used. Variants were developed on `dev_s`, but test was evaluated 26 times over the project, for 24 configurations: 14 with 8B, 10 with 70B, the 70B compact headers twice (4 and 14 October); the 26th evaluation, on 14 October, only reproduced the stored labels (see "Older booklets"). Add cold runs on 50 test cases (all stored in `runs/v1/test/` and `runs/v1/test_v11/`), and at least three default decisions used them: the section step was kept because test agreed, and compact headers and the ungated neutral re-check were not adopted for 70B on 4 October because test disagreed. Test is therefore not untouched. Two less exposed checks: (a) **`dev_rest`**, the 546 development claims outside `dev_s` from the same 13 dates. It served only as a veto: an option chosen on `dev_s` stayed only if `dev_rest` did not get worse; it never selected between variants. The claim-cue → section mapping was read off all development claims. Scores: **0.971 (8B) / 0.971 (70B)**, 0.968 / 0.972 without claims that also occur in another split;

(b) the synthetic hold-out below (0.874 / 0.932; the 8B value is its second evaluation, after the re-check was added). Over all 20 voting dates (dev + test, 1,488 claims) the defaults score 0.967 / 0.971 with a per-date spread of 0.87–1.00 (s.d. 0.030 / 0.028; `runs/v1/generalization.json`): a hidden set of a few voting dates can land several points lower or higher than our test split.

Tokens and time (14 October). Because OST scores tokens and time relative to the other teams, one more round cut both, with the same protocol: choose on `dev_s`, confirm once on `dev_rest`, then run once on test (the 24th test evaluation). (1) 8B re-checks a neutral verdict only when p(label) < 0.999: only 8 of 154 neutral answers on `dev_s` qualify, 6 of the 7 correct flips stay; `dev_s` 0.973 → 0.971, `dev_rest` 0.971 → 0.971, test 0.957 → 0.959, input tokens −19 %. (2) 70B uses compact passage headers: `dev_s` 0.966 → 0.973, `dev_rest` 0.965 → 0.967, test 0.967 → 0.963 (within the interval), tokens −7 %. Rejected on `dev_s`: compact headers for 8B (0.960), two or one few-shot examples (0.967 / 0.951 at −11 % / −18 % tokens).

Error analysis (14 October). On development data only (995 claims, defaults before this round, `experiments/v1/error_types.py`, `runs/v1/error-types-dev.json`), 8B makes 29 errors and 70B 30. Missed contradictions dominate: 20 (8B; 11 answered entailment, 9 neutral) and 19 (70B; 7 / 12) of them have gold contradiction, which is a third of the data; 19 / 21 are given with p ≥ 0.999. Numbers (13 / 16), dates (5 / 7), negation (5 / 4) and cross-lingual pairs (22 / 20) occur about as often among the errors as in the data; in 2 / 2 no quote lies in the gold passage. Countermeasures were tried on `dev_s` only. A number check (claim numbers missing from the passages) never fires: all 42 numeric entailments find their numbers. Re-checking every numeric entailment for a contradiction costs 8B 0.971 → 0.924 and 70B 0.973 → 0.960; another or a fourth example costs 8B 0.967 / 0.951. Kept, for 70B only: re-check a neutral verdict with p < 0.999 and accept the contradiction only at p ≥ 0.999: `dev_s` 0.973 → 0.978, `dev_rest` 0.967 → 0.971 (11 re-checks, 2 flips), then test once (the 25th evaluation) 0.963 → 0.965, +43 input tokens. The same re-check without the p < 0.999 gate had lost 0.002 on test on 4 October, so this is a second test look at a related option. The six few-shot rows lie in `dev_rest` (three used by 8B), so `dev_rest` is not fully clean for 8B.

Copies. v1.1 contains rows that are exact copies of another row (same claim, booklet and label), 122 of them in test (371 distinct pairs). We keep them as published; a plain bootstrap then counts copies as independent and its intervals are too narrow. Without copies the defaults score 0.957 (8B) and 0.962 (70B); resampling each group of copies as one unit gives [0.933–0.980] and [0.944–0.983] (`runs/v1/test_v11/final-v5-ci.json`; the method in `experiments/v1/duplicates_ci.py`). Two auxiliary sets written by closed LLMs are used for no reported number: 24 unreviewed edge cases (Gemini 3.8 Flash via the Antigravity CLI), demo presets only, and never evaluated because 16 of them concern proposals of the test dates; and archived synthetic development sets from before v1.0 (Claude Opus 5.5, GPT-5.5). A unit test checks that no stored run contains an edge case; sources and per-row provenance are in `data/README.md`.

Metrics as in OST's `evaluate.py` (starter repository, 8 October) and slide 28: macro-F1 over the three labels, separately for task A (Advanced) and B (Beginner), with 95 % bootstrap intervals (2,000 resamples, each group of copies as one unit, `runs/v1/test/duplicates-ci-v11.json`) and by language; OST's evidence score for task A (share of gold entailment/contradiction cases where one of the first five quotes, each at most 5,000 characters, matches the gold `reference_string` with fuzzy partial ratio ≥ 90; `experiments/v1/ost_evidence.py` reimplements it); input tokens per claim (OST's proxy counts them); mean and p95 of `inference_time_ms`. OST's validity thresholds (macro-F1 ≥ 0.60 on A, ≥ 0.70 on B) are passed by every setup below.

Main results (test v1.1, 493 pairs, macro-F1; input tokens per claim, 8B / 70B where they differ; `runs/v1/test/report-tables-v11.json`, the two default rows `runs/v1/test_v11/*/final-v5.summary.json`):

Setup	8B	70B	Input tok.
Baseline: whole booklet	0.720	0.889	20,242
Beginner: gold section	0.940	**0.980**	2,093
Beginner + 3 examples (8B)	**0.977**	–	2,675
Adv.: BM25, whole booklet	0.760	–	1,368
Adv.: proposal + BM25	0.776	0.817	1,310
Adv.: + translation	0.939	**0.951**	1,637 / 1,631
Adv.: + 3 ex. + decomp.	0.940	–	2,259
Adv.: + sections	0.953	0.967	2,130 / 1,532
Adv.: + neutral re-check	0.957	0.966	2,667 / 2,071
Default 8B: re-check only if p < 0.999	**0.959**	–	2,159
Default 70B: sections + compact headers + gated re-check	–	**0.965**	1,461

A dash means the step was not run for that model: the three examples and the decomposition are 8B steps (examples cost 70B 0.02 on `dev_s`). Defaults: 8B three examples, decomposition and the gated neutral re-check; 70B sections with compact headers and the gated neutral re-check (both since 14 October; `runs/v1/test_v11/*/final-v5.*`). The 8B pipeline beats the 8B baseline by +0.239 and even the 70B baseline; 70B gains +0.076. Both need 1.5–2.2 k instead of 20 k input tokens per claim. The section step added +0.011 (8B) and +0.015 (70B) on development data and +0.013 / +0.017 on test; the neutral re-check with the committee words added +0.016 / +0.009 on development data and +0.004 / −0.002 on test, so 70B did not use it (a decision that used test) until the stricter gate above.

By language (test v1.1, macro-F1; claim or booklet language, n in brackets; `runs/v1/test_v11/final-v5-languages.json`):

Group (n)	Adv. 8B	Adv. 70B	Beg. 8B	Beg. 70B
All (493)	0.959	0.965	0.977	0.980
Claim DE (163)	0.950	0.963	0.955	0.962
Claim FR (165)	0.945	0.964	0.982	0.989
Claim IT (165)	0.982	0.969	0.994	0.988
Booklet DE (152)	0.980	0.974	0.973	0.987
Booklet FR (186)	0.966	0.977	0.983	0.989
Booklet IT (155)	0.926	0.940	0.974	0.965
Same lang. (163)	0.993	0.980	0.979	0.987
Cross-ling. (330)	0.943	0.958	0.976	0.977

The whole-booklet baseline drops on cross-lingual pairs (8B 0.794 same vs. 0.684 cross); translation closes most of the gap. With the claim language detected instead of read from the data, the Advanced results are identical (detection right on all test claims).

Synthetic hold-out. To check that our choices did not fit OST's 20 voting dates, we built 89 claims on five proposals of three voting dates outside v1.0 (2018–2019, older layouts; claims drafted with Claude as a development aid, labels confirmed blindly by Nemotron and an open panel of GLM-5.3, DeepSeek V3.2 and Qwen3-235B, one row dropped; not human-annotated, never mixed with OST numbers) and evaluated them once after all defaults were fixed: whole booklet 0.665 (8B) / 0.872 (70B), GeaNaVid 0.845 / 0.932 at 9 % / 7 % of the tokens, Beginner 0.855 / 0.931 (`data/holdout-synth/`). The 8B neutral re-check, added later, was evaluated there once more: 0.874 (contradiction F1 0.735 ⟶ 0.793).

Older booklets (stress sets, 14 October). OST v1.1 covers 2020–2026 only. Two synthetic sets cover 26 other voting dates from 1998 to 2018 (802 claims, `data/stress-synth/`): claims written by Nemotron 3 Ultra from one booklet page each, kept only where two blind judges of other model families (Gemma 4 31B, Qwen3-235B) agree; no human checked them, and the judges dropped about half of the generated neutral claims, so the scores are rather too high. On the first set (10 dates) task A fell to 0.788 (8B) / 0.791 (70B), while task B held (0.889 / 0.976). Causes: page headers that name no proposal (retrieval over the whole booklet) or are font-encoding garbage (the wrong proposal), and the section boost, whose headings older layouts lack. Fix, without LLM calls: when the headers give fewer than two readable proposal names, proposals start at lines such as "Erste Vorlage / Premier objet / Primo oggetto" or at the overview's page ranges ("Seiten 4–11"); garbled text is dropped; no section boost; 70B reads 7 instead of 5 passages there. Each set was split by voting date before it was used (seeded); changes were developed on the first set's dev half only, every other half was run once:

Task A macro-F1, 8B / 70B	Before	First run	After the fix below
Set 1, dev half (157)	0.760 / 0.757	0.864 / 0.955	0.877 / 0.955
Set 1, held out (141)	0.819 / 0.828	**0.864 / 0.915**	0.849 / 0.922
Set 2, half 1 (252)	0.824 / 0.845	**0.861 / 0.881**	0.861 / 0.881
Set 2, half 2 (252)	0.767 / 0.813	**0.728 / 0.777**	0.776 / 0.841
All 802	0.793 / 0.815	–	0.836 / 0.890

The last half exposed a regression: in a booklet with a single proposal (2007) the overview fallback took explanation and voting text for two proposals and the title picked the voting text; the fallback now needs two proposal ordinals on the overview. That fix was found on held-out data, so the last column is not a clean estimate; the bold first runs are. Over all 802 claims input tokens rise from 2,267 / 1,543 to 2,341 / 1,779 and the evidence score (gold page as passage) from 0.62 / 0.63 to 0.77 / 0.78; task B is unchanged (0.889 / 0.963). No OST booklet takes these paths: an offline replay of all OST splits gives the stored labels (the 26th test evaluation, 0.959 / 0.965). The first set's held-out half is not fully clean either: its first report, read before the split, described every date's failure. Rejected: re-checking every neutral 8B answer (set 1 dev 0.858, `dev_s` +0.002 at +21 % tokens).

Evidence (task A). With OST's rule, our quotes score **0.969 (8B) and 0.963 (70B)** on the 324 gold entailment and contradiction cases of test v1.1 (`runs/v1/ost-evidence-test_v11-v5.json`); the misses are mostly gold entailments or contradictions that we answer neutral, which cite nothing. Up to 4 October we returned at most two sentences (0.932 / 0.920); five sentences from different passages were chosen on `dev_s` (8B 0.920 ⟶ 0.964, 70B 0.917 ⟶ 0.954; whole paragraphs instead of sentences: 0.954 / 0.944), confirmed on `dev_rest` (0.893 ⟶ 0.940, 0.888 ⟶ 0.923) and then run once on test. The first sentence is unchanged by this: it lies in the gold section for 0.862 / 0.865 of the answers and on a gold page for 0.872 / 0.874 (current defaults). OST does not check pages yet; ours are the 1-based PDF index, which equals the printed page number on 2,083 of the 2,084 pages of the 60 booklets where one is printed (`experiments/v1/page_numbering.py`). In a read sample of 30 first sentences (8B, before the section step), 22 decide the claim, 7 partly, 1 does not.

Efficiency as OST measures it. Cold runs on 50 seeded `dev_rest` cases in OST's input format, no response cache, one request at a time, old and new default interleaved case by case so both see the same endpoint (`experiments/v1/cold_latency.py`, `runs/v1/dev_rest_v11/cold-*-1014.json`); `inference_time_ms` is the wall time of one case (booklet parsed beforehand):

Model	Input tok. old ⟶ new	Mean ms old ⟶ new	p95 ms old ⟶ new
8B	2,320 ⟶ 1,979	543 ⟶ 366	972 ⟶ 545
70B	1,383 ⟶ 1,279	1,407 ⟶ 1,342	2,056 ⟶ 2,015
70B + gated re-check	1,276 ⟶ 1,319	1,338 ⟶ 1,274	2,050 ⟶ 2,197

The macro-F1 on these 50 cases is the same for old and new in the first pair (8B 0.980, 70B 0.956); the gated re-check, measured in a later session, flips one of them (0.937 → 0.956). Our own code takes 48 ms per claim (parsing is cached per booklet); the rest is Apertus. The endpoint's prefix cache cannot be switched off and its speed varies between sessions (the earlier 8B default measured 457, 648 and up to 1,530 ms per case on other days), so only same-session comparisons are meaningful. OST scores tokens and time with its own proxy and relative to the other teams; those scores exist only in OST's evaluation.

End-to-end as OST runs it (14 October, current defaults, `runs/v1/test_v11/ost-docker-full-1014/`): a fresh clone, `docker build --platform linux/amd64`, and `docker run -e BASE_URL -e API_KEY -v data:/data:ro -v output:/output <image> --input /data/cases.jsonl --output /output/predictions.jsonl` on the whole test split in OST's format, every row once as task A (booklet) and once as task B (reference), 986 cases (`experiments/v1/ost_cases.py`), no response cache, the two models run side by side (70B rerun alone after the gated re-check): exit code 0, 986 lines, no error for either model. OST's `evaluate.py` gives (a second 70B run the same evening, `…-1014b`, gave task A 0.965, 312 of 324, task B 0.974):

Model	Task A macro-F1 (493)	Task A evidence	Task B macro-F1 (493)	Task A input tok.	Task A ms mean / p95
8B	0.961	314 of 324	0.977	2,163	469 / 1,443
70B	0.965	312 of 324	0.976	1,460	1,059 / 2,344

The labels equal those of the stored runs except for 1 of 986 (8B) and 2 of 986 (70B) cases (the endpoint is not fully deterministic at temperature 0). Times here include parsing each of the 21 booklets on first use (and, for 8B, two runs sharing the endpoint), so the 8B times and the 70B p95 are higher than in the cold runs above (the 70B mean is lower, 1,059 vs 1,274 ms: the endpoint load differed). An earlier check on only 40 + 40 of these cases had shown 8B task A at 0.897: that sample happened to hold 4 of the 8B errors (10 % instead of 4 %), with labels identical to the full run; 40 cases are too few for a macro-F1.

5. Reproducibility

Everything runs in Docker: `make test` (unit tests), `make run` (the default pipeline on the v1.1 test split, `data/v1/test_v11.jsonl`, 493 claims, against the endpoint in `BASE_URL` / `API_KEY` or `LLM_*`) and `make ost` (a sample through OST's container contract, scored like OST's `evaluate.py`); no GPU or local weights are needed. Against the live endpoint the result can differ by a few labels (see below). Every stored run in `runs/v1/` holds its predictions, so the reported numbers can be recomputed without any model call (`experiments/v1/report_tables_v11.py`, `generalization.py`, `ost_evidence.py`). For OST's cases the image runs as OST specifies: `docker run -e BASE_URL -e API_KEY -v <cases>:/data:ro -v <out>:/output <image> --input /data/cases.jsonl --output /output/predictions.jsonl` (details in `track_2a/README.md`). `LLM_NAME` selects the model; the submitted image runs 70B (`swiss-ai/Apertus-v1.5-70B`, resolved via `/models` if the endpoint names it differently): higher macro-F1, fewer tokens and more robust on older booklets than 8B, which is faster (`-e LLM_NAME=apertus-v1.5-8b`). Both models are reported throughout. Settings: prompt `nli-v1`, k = 5, one neighbour, temperature 0, JSON schema. Replaying the stored test runs from the response cache with the current code gives the same labels on all test rows for both models; each run in `runs/v1/` stores its options, split seed and command. At temperature 0 the endpoint is not fully deterministic: a cold rerun of 50 cases gave the same label on 50/50 (8B) and 49/50 (70B). Commit of the submission: `c2dcd83f6c43`.

6. Limitations

493 test claims (371 distinct) from 7 voting dates: differences below about 0.02–0.03 macro-F1 are within the cluster-bootstrap interval (the defaults differ by 0.006, 70B − 8B), and test was evaluated 26 times (§4); per voting date the defaults range from 0.87 to 1.00. The model-aware defaults (examples and decomposition only for 8B, re-check gates per model) were chosen on 450 development claims. The section keywords cover the federal booklet layouts of 2018–2026; older layouts are scoped by their outline (stress set, synthetic, no human labels). Main error sources are topical but unsupported claims, attribution (committee vs. Federal Council) and contradictions that change one detail, which Apertus often calls neutral with high confidence (missed contradictions: 19 of 30 70B errors on development data, 12 of them called neutral; the gated re-check finds a few, a broader one over-calls contradiction). Booklets without a text layer are not handled (OST provides text PDFs). Labels and evidence are an automated reading of the booklet, not a voting recommendation.

OST's Solution API Guide and starter repository (7–8 October) settled page numbering (1-based PDF page), the Beginner page (`null`) and the evidence match (fuzzy quote inside the gold passage). Still open: the weighting of macro-F1, evidence, tokens and time in the final score, and whether the token-counting proxy passes `response_format` and `logprobs` through. Without them the client retries without schema and log-probabilities and the 8B decomposition is skipped (about −0.004 macro-F1 on development data). If the proxy offers neither the configured model name nor a model list, the container cannot find the proxy's name for the model by itself.

7. Lessons learned

- **Narrow first, then read.** Locating the proposal by its title costs nothing; with translation it beats the whole booklet by +0.22 (8B) and +0.06 (70B) with 1.6 k instead of 20 k input tokens. 8B with this context (0.939) beats 70B with the whole booklet (0.889).
- **Translation is the largest single step** for cross-lingual claims (ablation, 8B cross-lingual 0.696 → 0.938; the full defaults reach 0.943).
- **Prompt rules move the boundary instead of sharpening it.** Rules for "neutral" made 8B call real contradictions neutral; three worked examples helped on development data but not on test for the Advanced task. Apertus

is very confident (p > 0.9 for 94–98 % of answers), so probability gates rarely fire and calibration offsets were unstable.

- **A narrower second question beats a gate.** The neutral answers that hide a contradiction have p ≈ 1, but asking only "is one detail contradicted?" turns 5 of 11 such 8B errors on development data into right answers without a single wrong flip. The same question after an entailment flags most correct answers. Asking it only when the first answer has p < 0.999 keeps almost all of the gain with a twentieth of the re-checks: −19 % tokens and −44 % p95 latency for 8B.
- **Cheap evidence.** Choosing the evidence sentence after the decision costs no tokens and lands in the gold section more often than the whole cited passage (early ablation: 0.79 vs. 0.62 for 8B).
- **Read where the claim says it comes from.** "According to the summary" or "the committee thinks" names the section that holds the answer; routing on these words raised evidence hits by 7–8 points and macro-F1 by 0.013–0.017 at fewer tokens, with no model call.
- **Measure what the judges measure.** OST sums input tokens and takes mean and p95 of time per claim; our first numbers (total tokens, warm cache) had to be re-measured cold.
- **Check the input format, not only the dataset.** A `vote` title in another language than the booklet would pick the right proposal only 70 % of the time; translating it raises this to 95–98 % (OST later fixed `vote` to the booklet language).
- **Read the scoring script.** OST counts a case if any of the first five quotes matches; returning five sentences instead of at most two raised the evidence score by about four points without a token.
- **Keep a split nobody looked at.** Test was evaluated 26 times for 24 configurations; the 546 development claims that only confirmed options chosen elsewhere (0.971 / 0.971) are the more honest estimate for unseen claims on the same booklets.

Next steps: extend to Romansh (Stüssi & Vamvas, 2026) and to booklets without a text layer.

License

This report and the documentation: Creative Commons Attribution 4.0 (CC-BY-4.0). Code: Apache-2.0. Our edge-case set (`data/edge_cases.jsonl`), the synthetic hold-out and the stress sets (`data/holdout-synth/`, `data/stress-synth/`): CDLA-Permissive-2.0; their quoted booklet passages are official texts under Art. 5 URG. The OST benchmark and the splits derived from it keep their MIT licence. All HackApertus projects are open-sourced.

References

The reference list is rendered in the PDF from `report/refs.bib`.

Chen, L., Zaharia, M., & Zou, J. (2023). FrugalGPT: How to Use Large Language Models While Reducing Cost and Improving Performance. *Arxiv Preprint Arxiv:2305.05176*. https://arxiv.org/abs/2305.05176

Cormack, G. V., Clarke, C. L. A., & Buettcher, S. (2009). Reciprocal Rank Fusion Outperforms Condorcet and Individual Rank Learning Methods. *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 758–759. https://doi.org/10.1145/1571941.1572114

Koreeda, Y., & Manning, C. (2021). ContractNLI: A Dataset for Document-level Natural Language Inference for Contracts. *Findings of the Association for Computational Linguistics: EMNLP 2021*, 1907–1919. https://doi.org/10.18653/v1/2021.findings-emnlp.164

Li, Z., Li, C., Zhang, M., Mei, Q., & Bendersky, M. (2024). Retrieval Augmented Generation or Long-Context LLMs? A Comprehensive Study and Hybrid Approach. *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, 881–893. https://doi.org/10.18653/v1/2024.emnlp-industry.66

Project Apertus, Hernández-Cano, A., Hägele, A., & others. (2025). Apertus: Democratizing Open and Compliant LLMs for Global Language Environments. *Arxiv Preprint Arxiv:2509.14233*. https://arxiv.org/abs/2509.14233

Robertson, S., & Zaragoza, H. (2009). The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval*, *3*, 333–389. https://doi.org/10.1561/1500000019

Stüssi, E., & Vamvas, J. (2026). How Good is AI on Swiss Voting Booklets? A Multilingual OCR and Alignment Benchmark. *Proceedings of the 11th Edition of the Swiss Text Analytics Conference*, 232–242. https://aclanthology.org/2026.swisstext-1.22/

Thakur, N., Reimers, N., Rücklé, A., Srivastava, A., & Gurevych, I. (2021). BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models. *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*. https://arxiv.org/abs/2104.08663

Thorne, J., Vlachos, A., Christodoulopoulos, C., & Mittal, A. (2018). FEVER: a Large-scale Dataset for Fact Extraction and VERification. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 809–819. https://doi.org/10.18653/v1/N18-1074

Yin, W., Radev, D., & Xiong, C. (2021). DocNLI: A Large-scale Dataset for Document-level Natural Language Inference. *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 4913–4922. https://doi.org/10.18653/v1/2021.findings-acl.435