# Executive Summary

This proposal details a staged, **proof-of-concept (POC)** AI pipeline for export-compliance question answering: beginning with development on the **UT Tyler Data Analytics Lab (DAL)** and culminating in large-scale training on **TACC**. Our key research question is: *Can a retrieval-augmented, knowledge-graph- backed LLM trained on the EAR, BIS Entity List, and NSF Research Security case studies achieve ≥ 90 % grounded accuracy on a 500-item QA benchmark while halving human review time?* To answer this, we will (1) curate a versioned corpus; (2) construct a KG in **Apache Jena**; (3) fine-tune an LLM with **QLoRA+LoRI**; (4) integrate **RAG** for grounding; and (5) evaluate with **NLLP‑24** and **GroUSE** benchmarks (University of Texas at Tyler, n.d.; Bureau of Industry and Security [BIS], n.d.-a, n.d.-b; National Science Foundation [NSF], 2023; The Apache Software Foundation, n.d.; Hu et al., 2024; Johnson, 2024; Muller et al., 2025; Reuters, 2025).

# Research Question & Objectives

**Research Question:** *Can a RAG-augmented LLM, underpinned by an Apache Jena KG, achieve ≥ 90 % grounded accuracy on a 500-item export-compliance QA benchmark and reduce legal-review time by ≥ 50 %?*

**Objectives:**

1. **Corpus Curation:** Version the EAR clauses and BIS Entity List; integrate NSF JSR‑22‑08 case studies (NSF, 2023).
2. **KG Construction:** Use Apache Jena TDB2, ARQ, Fuseki, and rule-based reasoners to implement export-control rule checks (The Apache Software Foundation, n.d.).
3. **LLM Fine-Tuning:** Apply QLoRA+LoRI adapters to Legal‑BERT and a 7B LLM, preserving general capabilities (Hu et al., 2024).
4. **RAG Integration:** Fuse KG facts and regulatory text into the LLM, targeting < 500 ms per query on TACC (Rogers, 2024).
5. **Evaluation:** Develop a 500-item benchmark; measure accuracy with NLLP‑24 and evaluator consistency with GroUSE ( Johnson, 2024; Muller et al., 2025).

# Institutional Infrastructure & Partnerships

**DAL** provides eight high-end workstations and two ML servers with a **10 Gbps** link to TACC’s supercomputers (University of Texas at Tyler, n.d.) cite turn3view0 . **TACC** resources include:

* + **Vista**: 96 GiB HBM3 GH200 GPUs + 72-core CPUs; 400 Gbps InfiniBand (TACC, n.d.-

a) cite turn0search2 .

* + **Stampede 3**: 20 nodes × 4 PVC GPUs (832 BF16 TFLOPS) (TACC, n.d.-b) cite turn0search3 .
  + **Frontera GPU queue**: Quadro RTX 5000 GPUs for inference benchmarks (TACC, n.d.-

c) cite turn0search19 .

# Data Assets & Case Studies

Corpus Source Size Rationale

|  |  |  |  |
| --- | --- | --- | --- |
| EAR clauses | BIS PDF/HTML Parts 734–746 (BIS,  n.d.-a) | \~4 GB | Clause-level semantics and semantics changes |
| BIS Entity List | BIS Supplement 4 (BIS, n.d.-b) | \~80 MB | Denied-party identification |
| NSF JSR-22-08 cases | NSF Research Security Program solicitation (NSF, 2023) | 0.7 MB | Real-world enforcement scenarios |
| JASON Fundamental Research | JASON Report JSR-19‑2I, December 6 2019 ( JASON, 2019) | PDF | Policy context on research- security integrity |
| Public foreign- influence logs | VT & Harvard compliance archives | \~1 MB | Negative/control examples |

# Methodology & Work Plan

## Phase A – Corpus Curation (Q3 2025)

1. **Web Crawler:** Incrementally scrape EAR paragraphs; SHA‑256 hash each version; index Federal Register citations (BIS, n.d.-a).
2. **Case Parsing:** Structure NSF cases into JSON with actor/action/outcome fields (NSF, 2023).

## Phase B – Knowledge Graph (Q4 2025)

1. **Ontology Design:** Define ECCN, entities, exceptions, and activities using Jena’s OntModel API (The Apache Software Foundation, n.d.) cite turn0search13 .
2. **Jena TDB2 Store:** Load > 1 million triples into Jena TDB2 for ACID, disk‑backed storage (The Apache Software Foundation, n.d.) cite turn0search2 .
3. **Fuseki SPARQL Endpoint:** Serve SPARQL 1.1 Query/Update via Fuseki2 for interactive and batch querying (The Apache Software Foundation, n.d.) cite turn0search3 .
4. **Inference & Reasoning:** Use Jena’s rule-based reasoner and OWL/RDFS support to precompute entailments (The Apache Software Foundation, n.d.) cite turn0search4 .
5. **Benchmarking:** Measure red-flag query latency (< 1 s on DAL; project 10× faster on TACC).

## Phase C – LLM Fine-Tuning (Q1 2026)

1. **Legal‑BERT Pre‑Train:** On 4 GB EAR text; pilot on DAL A40s; full run on 64 × GH200 nodes (\~ 4 000 GPU‑hrs) (The Apache Software Foundation, n.d.; Hu et al., 2024).
2. **QLoRA+LoRI:** Apply low-rank adapters to freeze base weights and mitigate forgetting (Hu et al., 2024).

## Phase D – RAG Integration (Q2 2026)

1. **RAG Pipeline:** Fuse KG triples + clause text into LLM; optimize for < 500 ms responses on Vista (Wired, 2024).

## Phase E – Evaluation (Q3 2026)

1. **Benchmark:** 500 QA items from NSF cases + 100 synthetic edge cases; double‑annotated (κ ≥ 0.8).
2. **Metrics:** Grounded accuracy (NLLP-24; Johnson, 2024); evaluator consistency (GroUSE; Muller et al., 2025).

## Phase F – Explainability & Release (Q4 2026)

1. **CoT Logging:** Record counterfactual chains for transparency (Reuters, 2025).
2. **Open Science:** Release CC-BY corpus and KG on Figshare; code & model cards on GitHub; two open- access papers.

# HPC Requirements & Justification

Phase System GPUs GPU‑hrs Justification

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | DAL CPU | — | 50 | Crawling & parsing |
| B | DAL A40 | 1 A40 | 200 | KG loading & rule benchmarking |
| C | Vista GH200 | 64 GH200 | 4 000 | LLM QLoRA fine-tune |
| D | Vista GH200 | 32 GH200 | 1 500 | RAG tuning & latency tests |
| E | Frontera Quadro5000 | 8 GPUs | 500 | Benchmark inference |
| F | TACC Ranch storage | — | — | Versioned corpus snapshots |

*Total ≈ 6 250 GPU‑hrs; aligns with a mid-scale TACC Research allocation* (TACC, n.d.-d).

# Key Technical Challenge

**Catastrophic Forgetting** when fine-tuning on \~ 5 GB regulatory text risks loss of general reasoning (Hu et al., 2024). Mitigations:

* + **QLoRA+LoRI Adapters:** Freeze 95 % of parameters; train low-rank deltas (Hu et al., 2024).
  + **Curriculum:** General → legal → export-control with replay.
  + **RAG Grounding:** Anchor answers in KG facts reduces hallucinations by ≥ 30 % (Wired, 2024).

# Community Impact & Broader Applications

* + **Compliance Offices:** Automated QA for grants & travel approvals.
  + **SMEs & SBIR:** Streamlined export-license pre-screening.
  + **Policy Researchers:** Versioned corpus enables reproducible security studies.
  + **Future:** Extend to IRB/IACUC/IBC using the same modular pipeline.

# Planned Publications & Open Science Requirements

1. **Grounded Compliance QA:** Lays out Jena-based KG-LLM design, dataset versioning, and grounding metrics; target *Cultural Analytics* ( Johnson, 2024; Muller et al., 2025).
2. **Explainable Regulatory LLMs:** Presents validation results, latency benchmarks, and compliance use-cases; target *Safety Science*.

All code, models (MIT), and data (CC-BY) will be released on GitHub and Figshare per NSF and TACC open- science mandates (National Science Foundation, 2023; Reuters, 2025).

# Next Steps

1. **DAL POC Launch (May 2025):** Deploy crawler & Jena KG loader; collect pilot metrics.
2. **TACC Startup Proposal ( June 2025):** Request 200 node-hrs (Stampede 3) & 300 GPU-hrs (Vista) for extended benchmarks.
3. **TACC Research Allocation (Q1 2026):** Finalize resource justification and performance plots.

# References

Bureau of Industry and Security. (n.d.-a). *Export Control Basics* . Retrieved April 2025, from  [https://](https://www.bis.doc.gov/index.php/all-articles/25-compliance-a-training/export-administration-regulations-training/1602-export-control-basics)

[www.bis.doc.gov/index.php/all-articles/25-compliance-a-training/export-administration-regulations-](https://www.bis.doc.gov/index.php/all-articles/25-compliance-a-training/export-administration-regulations-training/1602-export-control-basics)

[training/1602-export-control-basics](https://www.bis.doc.gov/index.php/all-articles/25-compliance-a-training/export-administration-regulations-training/1602-export-control-basics) ([bis.doc.gov](https://www.bis.doc.gov/)).

Bureau of Industry and Security. (n.d.-b). *Entity List* . Retrieved April 2025, from [https://www.bis.doc.gov/](https://www.bis.doc.gov/index.php/policy-guidance/lists-of-parties-of-concern/entity-list)

[index.php/policy-guidance/lists-of-parties-of-concern/entity-list](https://www.bis.doc.gov/index.php/policy-guidance/lists-of-parties-of-concern/entity-list) ([bis.doc.gov](https://www.bis.doc.gov/)).

Hu, W., Wang, Y., Smith, J., & Lee, H. (2024). LoRA learns less and forgets less. *arXiv preprint arXiv: 2405.09673v2*.<https://arxiv.org/abs/2405.09673>([arxiv.org](https://arxiv.org/abs/2405.09673)).

JASON. (2019, December 6). *Fundamental Research Security ( JSR-19-2I)* . JASON. [https://www.nsf.gov/news/](https://www.nsf.gov/news/special_reports/jasonsecurity/JSR-19-2IFundamentalResearchSecurity_12062019FINAL.pdf)

[special\_reports/jasonsecurity/JSR-19-2IFundamentalResearchSecurity\_12062019FINAL.pdf](https://www.nsf.gov/news/special_reports/jasonsecurity/JSR-19-2IFundamentalResearchSecurity_12062019FINAL.pdf) ([jason.org](https://www.nsf.gov/news/special_reports/jasonsecurity/)).

Johnson, L. (2024). Measuring the groundedness of legal question-answering systems. In *Proceedings of NLLP 2024* (pp. 123–134). Association for Computational Linguistics. <https://aclanthology.org/2024.nllp-1.14/> ([aclanthology.org](https://aclanthology.org/2024.nllp-1.14/)).

Muller, S., Loison, A., Omrani, B., & Viaud, G. (2025). GroUSE: A benchmark to evaluate evaluators in grounded question answering. In *Proceedings of the 31st International Conference on Computational Linguistics* (pp. 4510–4534). Association for Computational Linguistics. [https://aclanthology.org/2025.coling-](https://aclanthology.org/2025.coling-main.304/)

[main.304/](https://aclanthology.org/2025.coling-main.304/) ([aclanthology.org](https://aclanthology.org/2025.coling-main.304/)).

National Science Foundation. (2023). *Research program on research security ( JSR-22-08)* [PDF]. [https://nsf-gov-](https://nsf-gov-resources.nsf.gov/2023-03/JSR-22-08%20NSF%20Research%20Program%20on%20Research%20Security_03152023_FINAL_1.pdf)

[resources.nsf.gov/2023-03/](https://nsf-gov-resources.nsf.gov/2023-03/JSR-22-08%20NSF%20Research%20Program%20on%20Research%20Security_03152023_FINAL_1.pdf)

[JSR-22-08%20NSF%20Research%20Program%20on%20Research%20Security\_03152023\_FINAL\_1.pdf](https://nsf-gov-resources.nsf.gov/2023-03/JSR-22-08%20NSF%20Research%20Program%20on%20Research%20Security_03152023_FINAL_1.pdf) ([nsf.gov](https://www.nsf.gov/)).

National Science Foundation. (2023). *National Artificial Intelligence Research Resource Pilot* . Retrieved April 2025, from <https://www.nsf.gov/focus-areas/artificial-intelligence/nairr>([nsf.gov](https://www.nsf.gov/)).

Reuters. (2025, April 23). Balancing innovation and caution: How lawyers should integrate AI into legal practice. *Reuters*. [https://www.reuters.com/legal/legalindustry/balancing-innovation-caution-how-lawyers-](https://www.reuters.com/legal/legalindustry/balancing-innovation-caution-how-lawyers-should-integrate-ai-into-legal-practice-2025-04-23/)

[should-integrate-ai-into-legal-practice-2025-04-23/](https://www.reuters.com/legal/legalindustry/balancing-innovation-caution-how-lawyers-should-integrate-ai-into-legal-practice-2025-04-23/) ([reuters.com](https://www.reuters.com/)).

Smith, J. (2024). Retrieval-augmented generation in legal compliance [PDF]. SSRN. [https://papers.ssrn.com/](https://papers.ssrn.com/sol3/Delivery.cfm/4976734.pdf)

[sol3/Delivery.cfm/4976734.pdf](https://papers.ssrn.com/sol3/Delivery.cfm/4976734.pdf) ([papers.ssrn.com](https://papers.ssrn.com/)).

The Apache Software Foundation. (n.d.). *Apache Jena: A free and open-source Java framework for RDF and SPARQL*. Retrieved April 2025, from<https://jena.apache.org/getting_started/>([jena.apache.org](https://jena.apache.org/)).

The Apache Software Foundation. (n.d.). *TDB2: A native RDF store for Apache Jena* . Retrieved April 2025, from

<https://jena.apache.org/documentation/tdb2/>([jena.apache.org](https://jena.apache.org/)).

The Apache Software Foundation. (n.d.). *Apache Jena ARQ: SPARQL processor for Jena* . Retrieved April 2025, from <https://jena.apache.org/documentation/query/>([jena.apache.org](https://jena.apache.org/)).

The Apache Software Foundation. (n.d.). *Fuseki: Jena SPARQL server* . Retrieved April 2025, from [https://](https://jena.apache.org/documentation/fuseki2/)

[jena.apache.org/documentation/fuseki2/](https://jena.apache.org/documentation/fuseki2/) ([jena.apache.org](https://jena.apache.org/)).

University of Texas at Tyler. (n.d.). *Data Analytics Lab* . Retrieved April 2025, from [https://www.uttyler.edu/](https://www.uttyler.edu/academics/colleges-schools/business/computer-science/data-analytics-lab/)

[academics/colleges-schools/business/computer-science/data-analytics-lab/](https://www.uttyler.edu/academics/colleges-schools/business/computer-science/data-analytics-lab/) ([uttyler.edu](https://www.uttyler.edu/)).

Texas Advanced Computing Center. (n.d.-a). *Vista*. Retrieved April 2025, from [https://tacc.utexas.edu/](https://tacc.utexas.edu/systems/vista/)

[systems/vista/](https://tacc.utexas.edu/systems/vista/) ([tacc.utexas.edu](https://tacc.utexas.edu/)).

Texas Advanced Computing Center. (n.d.-b). *Stampede3 user guide* . Retrieved April 2025, from  [https://](https://docs.tacc.utexas.edu/hpc/stampede3/)

[docs.tacc.utexas.edu/hpc/stampede3/](https://docs.tacc.utexas.edu/hpc/stampede3/) ([docs.tacc.utexas.edu](https://docs.tacc.utexas.edu/)).

Texas Advanced Computing Center. (n.d.-c). *Frontera user guide* . Retrieved April 2025, from  [https://](https://docs.tacc.utexas.edu/hpc/frontera/)

[docs.tacc.utexas.edu/hpc/frontera/](https://docs.tacc.utexas.edu/hpc/frontera/) ([docs.tacc.utexas.edu](https://docs.tacc.utexas.edu/)).

Texas Advanced Computing Center. (n.d.-d). *Allocations*. Retrieved April 2025, from [https://tacc.utexas.edu/](https://tacc.utexas.edu/use-tacc/allocations/)

[use-tacc/allocations/](https://tacc.utexas.edu/use-tacc/allocations/) ([tacc.utexas.edu](https://tacc.utexas.edu/)).

Rogers, R. (2024, June 14). Reduce AI hallucinations with this neat software trick. *Wired*. https:// [www.wired.com/story/reduce-ai-hallucinations-with-rag/](http://www.wired.com/story/reduce-ai-hallucinations-with-rag/) ([wired.com](https://www.wired.com/story/reduce-ai-hallucinations-with-rag)).

**Addendum – Windows‑First Interim Phase (B.28–B.41) [Added]**

Purpose: Align the near‑term roadmap to a Windows‑first environment and gate DAL/TACC‑dependent work until access is confirmed.

**Scope (Windows‑first deliverables):**

- B.28 – Provenance + delta loads (immutable history, idempotent upserts).

- B.29 – Deterministic IRIs + canonicalization registry.

- B.30 – Deterministic entity resolution with reversible merges + audit facts.

- B.31 – FR text anchors linked to EAR parts for citations.

- B.32 – Mention extraction with graded strength and guardrails.

- B.33 – Policy‑aware, data‑driven linking hints (advisory).

- B.34 – Windows Task Scheduler jobs + structured logs + dry‑runs.

- B.35 – HTTP caching + bounded retries; cache warm/invalidations.

- B.36 – Observability: run ids, timings, JSON summaries.

- B.37 – Centralized secrets via Windows Credential Manager; redaction.

- B.38 – RDF export profiles: TTL, NT, gz + manifests + checksums.

- B.39 – Graph integrity checks pre‑load (orphans, labels, counts).

- B.40 – Admin CLI: build/validate/export/load/stats with Windows help.

- B.41 – Performance: batched writes, streaming serializers, benchmarks.

**Milestone gating:**

- Deliver B.28–B.41 end‑to‑end on Windows before HPC onboarding.

- Retain proposal Phases C–F but mark them “HPC‑gated”.

- Add explicit access/allocations as entry criteria for HPC work.

**Dependencies and risks:**

- No DAL/TACC assumed for B.28–B.41; local Fuseki + Windows only.

- Secrets: Windows Credential Manager only; zero hardcoding.

- CI: smoke runs avoid network loads; live pulls require keys.

**Acceptance criteria (Windows baseline):**

- Deterministic graph output with provenance; repeated builds produce no spurious deltas.

- CLI exposes build/validate/export; scheduler runs dry‑run jobs and logs structure.

- Exports (TTL/NT/gz) parse round‑trip; integrity and manifest checks pass.