Capstone Proposal

From Skills to KSAs: A Tool-Agnostic Framework for Military-to-Civilian Translation

Proposed by: Kyle Hall

Email: Kyle.hall@gwu.edu

Advisor: Amir Jafari

The George Washington University, Washington DC

Data Science Program

1 Objective:

Demonstrate an end-to-end data science capstone by building a **tool-agnostic KSA extraction pipeline** for selected **USAF AFSCs**, aligning the results to **civilian skill frameworks (ESCO/O*NET)**, and publishing a **graph network** with a **lightweight, static website** that lets users search an AFSC and view KSAs (with provenance). Optionally, incorporate a **preference-learning (RL-light) re-ranker** trained on small batches of SME feedback to improve KSA-to-standard alignment ordering.

Specific objectives (plain language):

1. Pull KSAs from unclassified Air Force documents (AFOCD/AFECD, CFETPs, fact sheets, syllabi) and **keep the page/section** for every item.
2. Turn results into a tidy **K/S/A table** with a short description, type (K|S|A), deduped phrasing, provenance, **ESCO/O*NET** links, and a confidence score.
3. **Publish a graph export** (AFSC ↔ KSA ↔ ESCO/O*NET ↔ SourceDoc) and a **per-AFSC static JSON** bundle that powers a web demo.
4. Ship a **searchable static site** (GitHub Pages/Netlify) where a user selects an AFSC → sees K/S/A tabs, simple filters, evidence toggle, and an **optional** interactive graph view for that AFSC's subgraph.
5. (Optional) Collect ~150–200 SME preferences and train a **pairwise re-ranker** to improve top-k alignment ordering (report uplift with NDCG@10, P@5).

Success criteria: usable AFSC search; evidence-backed KSAs; importable graph exports; and measurable accuracy on a small labeled set (see §5).

---

2 Dataset:

- **Sources (unclassified):** AFOCD/AFECD (e.g., current public releases), CFETPs, fact sheets, training syllabi.
- **Coverage (initial):** 6–8 AFSCs across 2–3 families (e.g., Operations / Intelligence / Maintenance). Pipeline and site are designed to scale to "most AFSCs" later by adding more documents.
- **Parsing:** structure-aware (headings/sections/pages) to support precise **provenance** for each extracted item.

- **Civilian standards for alignment: ESCO** (public), **O*NET** (local copy).
- **Availability:** All datasets are **publicly accessible**; any restricted materials will be excluded or reduced to derived metadata only.
- **Provenance captured:** document title/date, section/page, character offsets (when feasible), and a short evidence quote.

---

3 Rationale:

The DoD lacks a unified, machine-readable KSA layer that ties AFSC-specific competencies to civilian frameworks. This project delivers an **evidence-bearing, queryable KSA graph** and **simple web access**, enabling service members to translate experience, and enabling educators and planners to align curricula and workforce needs.

- **Operational impact:** A reusable KSA layer with provenance reduces manual curation time for career translation, curriculum updates, and program evaluation.
- **Framework bridging:** Dual alignment (ESCO/O*NET) maximizes interoperability with civilian ecosystems.
- **Demonstrable value:** A working web demo + importable graph artifacts show end-to-end feasibility within a capstone timeline.

---

4 Method:

4.1 Pipeline Overview

1. **Ingest:** Collect AFSC-relevant documents; extract text with layout cues; segment by section/page.
2. **Candidate KSA Extraction:** Hybrid approach—pattern cues + POS/NER, embedding-based retrieval, and LLM prompts specialized for K/S/A.
3. **Normalization & Deduplication:** Cluster semantically similar items; canonicalize phrasing; tag K|S|A.
4. **Alignment to ESCO/O*NET:** Vector similarity + lexical overlap + concept heuristics; keep top-k with scores.
5. **Provenance & QA:** Attach (doc_id, section/page, evidence snippet); add sanity checks to reduce ability/skill conflation.
6. **Graph Build & Exports:** Emit **GraphML**, **Neo4j CSV**, and **per-AFSC JSON** bundles used by the web demo.

4.2 Graph Schema (typed)

Nodes: `AFSC` (id, title, family) · `KSA` (id, normalized_text, type ∈ {K,S,A}) · `SourceDoc` (id, title, ref) · `Standard` (id, type ∈ {ESCO,O*NET}, label)
Edges: `AFSC`—[HAS_K|HAS_S|HAS_A]→`KSA`; `KSA`—[SUPPORTED_BY {page,section,quote}]→`SourceDoc`; `KSA`—[ALIGNS_TO {score}]→`Standard`

4.3 Web Demo (static-first)

- **Stack:** Vite/React, Fuse.js for client-side search, Cytoscape.js for the optional graph tab.
- **Data:** `afsc_index.json` for search + `afsc/<AFSC>.json` subgraphs (built offline by the pipeline).

- **UX:** AFSC search → K/S/A tabs with counts → filter by type / alignment / confidence → toggle evidence → optional Graph tab → **Download JSON/CSV**.
- **Hosting:** GitHub Pages/Netlify (no backend). If live DB queries are later required, add a minimal FastAPI service.

4.4 Preference-Learning Re-Ranker (RL-light, optional)

- **Goal:** Improve the ordering of KSA→Standard alignments.
- **Data:** ~150–200 SME binary/pairwise judgments (A vs B).
- **Model:** Pairwise logistic (Bradley-Terry) or LambdaMART (XGBoost).
- **Features:** cosine sim, lexical overlap, doc frequency, cue-phrase flags, evidence heuristics.
- **Metrics:** NDCG@10, P@5 uplift vs baseline; SME "useful" rate.

4.5 Methods & Techniques — Prioritized (Top 10)

- **1) Ingest & Provenance:** Structure-aware parsing of AFSC documents with page/section capture (pdfminer.six / unstructured.io).
- **2) Pattern-Based KSA Spotting:** Cue-phrase mining + POS/NER to extract candidate K/S/A spans.
- **3) Semantic Retrieval for Candidates:** Dense embeddings (sentence-transformers) + FAISS to surface KSA-salient sentences.
- **4) LLM Typing & Canonicalization:** Few-shot prompts at low temperature to assign K/S/A labels and paraphrase to a clean form.
- **5) De-duplication & Clustering:** Embedding-similarity thresholding and light clustering to merge near-duplicates.
- **6) Hybrid Alignment Scoring:** BM25 (lexical) + cosine (dense) fusion to rank ESCO/O*NET matches (retain top-k).
- **7) Evidence & QA:** Snippet extraction with page refs plus guardrail heuristics to reduce K/S/A conflation.
- **8) Graph Build & Exports:** networkx → GraphML / Neo4j CSV; emit per-AFSC JSON bundles for the site.
- **9) Static Web Search UX:** React + Fuse.js with simple filters; optional Cytoscape.js graph tab.
- **10) Preference Re-ranking (RL-light):** Pairwise logistic or LambdaMART trained on quick SME preferences; report NDCG@10 uplift.

> *Estimated order reflects pipeline flow; items 9–10 can proceed in parallel once data artifacts exist.*

---

5 Evaluation:

- **Gold set:** label ~100–150 KSA items across AFSCs with validity, type, and alignment.
- **Extraction:** Precision/Recall/F1 (target **F1 ≥ 0.70** on the labeled set).
- **Alignment:** Accuracy@k (target **@5 ≥ 70%**).
- **Web usability:** Task success (find 3 KSAs) and time-to-answer; brief SUS survey.
- **Re-ranker (if used): NDCG@10 uplift ≥ +5%** vs baseline on held-out AFSCs.

---

6 Deliverables:

- **Data:** normalized KSA table; alignment table with scores; evidence snippets.

- **Graph: GraphML**, **Neo4j CSV**, **per-AFSC JSON** bundles.
- **Web Demo (static):** AFSC search + filters; evidence toggle; optional graph tab; download.
- **Re-Ranker (optional):** model file + short results note.
- **Final Write-Up:** methods, evaluation, limitations, and ethics considerations.

---

7 Timeline (indicative, Fall 2025):

W1–2: finalize AFSC list; gather documents; repo + schema.
W3–4: extraction (pattern + embeddings + prompts); normalization.
W5: ESCO/O*NET alignment; evidence pipeline; first labeled set.
W6: graph build + exports (GraphML/Neo4j CSV/JSON).
W7: web MVP (search + K/S/A tabs + evidence toggle).
W8: optional graph tab; polish; accessibility pass.
W9: preference collection (SME).
W10: train re-ranker; measure uplift; finalize demo & report.

---

8 Risks & Descoping:

- **Doc access/quality:** expand to adjacent curriculum docs; if needed, omit quotes but keep page refs.
- **Time/compute:** cache embeddings; CPU-friendly models; narrow AFSCs (4–5) or ship web MVP without graph tab.
- **Data variance:** rules to reduce false positives; maintain error log and spot checks.

---

9 Ethics & Responsible Use:

- Use only unclassified sources; store derived metadata + page refs, not restricted documents.
- Document limitations (LLM hallucinations; K/S/A boundary ambiguity).
- Anonymize SME judgments and make preference data opt-in.

Appendix — Mini-Glossary

KSA — Knowledge, Skills, and Abilities (competency structure used in DoD)
AFSC — Air Force Specialty Code (job family)
ESCO / O*NET — Civilian skill/occupation frameworks used for alignment
Provenance — Doc title/date, section/page, and a short quote showing the evidence for each item
CFETP / AFOCD / AFECD — Unclassified Air Force classification/training documents

Contact

- Author: Amir Jafari
- Email: ajafari@gwu.edu
- GitHub: amir-jafari/Capstone