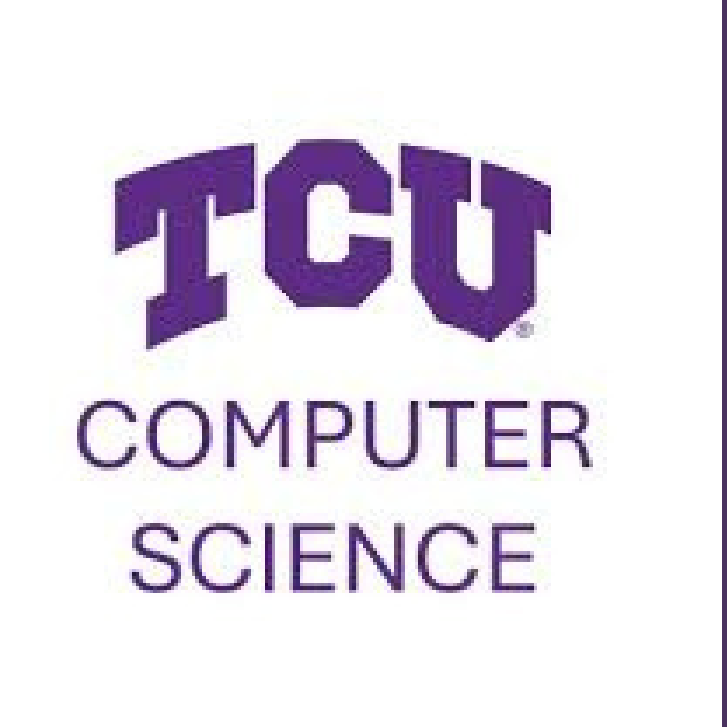




iPELiNT USPTO Forensics



Adessa Segura, Daltin Plitt, Dylan Caton, Eric Licea Tapia, Jane Allinger, Kasia Love

Abstract

How would one classify an apple fruit versus an apple phone? Typically as a fruit and a technology device. However some modern systems for classifying patents are insufficient and would be unable to differentiate between the two and cluster both based on their containing the word 'apple'. Our task with iPELiNT is to build upon solutions to better visualize how USPTO(United States Patent and Trademark Office) art unit's change over time. An art unit is a group of USPTO examiners specializing in a specific technology area. Our end product helped establish a data-driven system for conducting forensic analysis of USPTO patent examiner dockets using vector embeddings and internal data pipelines. We used mongoDB for our database, JavaScript and Python for our backend, and NuxtJS and Vue for our frontend. Our 5 phases of development are as follows. 1. Data Aggregation and Preparation. 2. Centroid Calculation and Art Unit Profiling. 3. Deviation Analysis and Scoring 4. Visualization and interpretation Framework.

Front End Methods and Results

Search and Dropdown UI: UI features a search function with a "render dropdown" that allows users to view an examiner and their corresponding art unit simultaneously. Users can click on an art unit to see all the examiners within it, or select a specific examiner to find out which art unit they belong to.

Ai-Generated Insights: A "loading" examiner function populates text in the screen including an AI-generated insight paragraph. If the user is unhappy with the provided insights, they have the ability to regenerate a new paragraph

Progress Bars: These display metrics for allowance, rejection, and deviation. To work around initial bugs, front end utilizes setting the bars to 0 and applying a 60-second delay using a setTimeout function for proper display.

Data Visualization: this is designed to read data from a Mongo database to display visual plots to user. Backend info such as 2D coordinates and patent topic clusters can now be mapped out visually to display groupings of similar patents

Back End Methods and Results

- **Data Processing:** We ingest raw patent data and filter out common legal jargon, isolating the unique keywords that define each patent's core technology.
- **Semantic AI Translation:** We use an AI encoder to convert the cleaned text into 768-dimensional mathematical vectors, capturing the deep semantic meaning of the patent.
- **Clustering & Mapping:** The system calculates a yearly "average" for each tech category and groups related patents into sub-specialties (e.g., distinguishing "bullets" from "guns").
- **Outlier Detection:** Using multi-dimensional distance metrics (Mahalanobis Distance), we precisely measure how far individual patents deviate from their expected technological baseline.
- **Interactive Visualization:** We reduce the massive 768-dimensional dataset down to a 2D plane, storing the results in a database to power a fast, interactive visual map for end-users.



Goals

- **Catching Lost Patents:** We automatically flag patent applications that were sent to the wrong review department.
- **Matching Experts to Ideas:** We track what reviewers are working on to ensure they are assigned to inventions that actually fit their technical background.
- **Spotting New Trends:** We group similar patents together to discover brand-new, emerging technology fields as they form.
- **Planning for the Future:** We use this data to predict where the patent office needs to reorganize or create entirely new departments to handle future innovations.

