AI-Enabled Multilingual Semantic Search for Accurate NCO Classification in NSS Surveys

Problem Statement

The National Sample Survey (NSS) requires enumerators to assign the correct National Classification of Occupations (NCO) codes based on respondents' job descriptions. Many respondents provide vague, ambiguous, or regional-language answers, making it difficult to assign the right NCO code. This often leads to inconsistent classification, lower data quality, and increased post-survey corrections. There is a need for a tool that can understand multilingual responses, find the closest matching occupation, and guide enumerators to accurate classification.

Proposed Solution

We propose an AI-powered semantic search tool that processes respondents' job descriptions in Tamil, English, or other Indian languages, through text or speech input. The system uses speech-to-text and neural machine translation to normalize responses into a common language for processing. Both the input and the NCO job database are converted into vector embeddings using multilingual transformer models such as XLM-R or Sentence-BERT. These embeddings are stored in a FAISS index for instant similarity search across thousands of entries.

The system returns the best matching NCO code, job title, and description in both English and the local language. If multiple matches have similar confidence scores, it generates context-specific clarifying questions in the respondent's language to guide the enumerator toward the most accurate classification. The tool is designed to work offline using a lightweight SQLite database and AI models deployed via TensorFlow Lite or ONNX Runtime, making it suitable for field surveys with low or no internet connectivity

Expected Outcomes

- Improved accuracy of NCO coding in NSS surveys.
- Faster and more efficient survey operations.
- Reduced training requirements for enumerators.
- Consistent and high-quality occupational data for policy-making.
- Inclusion of non-English-speaking respondents without losing accuracy.
- Offline functionality ensures usability in diverse field environments.

Tools & Technologies

- Programming Language: Python
- Libraries: FAISS, Sentence-BERT, XLM-R, TensorFlow Lite, ONNX Runtime
- Database: SQLite for offline storage
- Speech & Translation: Vosk (ASR), MarianMT / IndicTrans2
- Deployment: Mobile or web application with offline capability