Horizon Europe Topic Matcher - Project Report

Abstract

This project develops an AI-powered system for matching research project proposals to relevant Horizon Europe funding calls (2021–2027). Using publicly available CORDIS project data and official call descriptions, the system leverages semantic embeddings, keyword features, and temporal information to recommend the most suitable calls for a given proposal. A hybrid approach combining user-driven filtering with multi-modal embeddings ensures accurate and efficient retrieval. Evaluation shows the system consistently identifies highly relevant topics among the top recommendations, demonstrating the potential of AI-assisted grant discovery for universities, SMEs, and consultants.

Introduction

Finding the right funding opportunity within the Horizon Europe program (2021–2027) is a complex and time-consuming task. Thousands of calls are published across multiple domains, each with detailed descriptions, objectives, and eligibility criteria. Potential applicants—universities, research groups, SMEs, and consultants—often face significant challenges in identifying the calls that are most relevant to their project ideas. Manual navigation of the program’s databases, such as CORDIS and the official Funding & Tenders portal, requires extensive expertise, considerable time, and careful attention to both thematic content and administrative requirements.

To address this challenge, the objective of this project is to develop an AI-powered recommendation system capable of automatically matching project proposals to the most relevant Horizon Europe funding calls. The system leverages publicly available data from CORDIS projects, which provide historical project abstracts, titles, keywords, and assigned call IDs, as well as detailed topic descriptions from official Funding & Tenders calls. By combining these sources, the system aims to predict the best funding opportunities for a given project idea, even when users do not have prior knowledge of EU funding mechanisms.

Target Users

The target users of such a system, once fully in production, could include:

- **Universities and research groups** preparing competitive proposals for EU funding.
- **Small and medium-sized enterprises (SMEs)** seeking innovation funding but lacking dedicated EU funding experts.
- **Consultants and project facilitators** aiming to streamline their grant application process.

By providing a tool that rapidly surfaces the top-matching calls for a given project idea, the system improves access to funding opportunities, reduces the time spent on research, and increases the likelihood of submitting relevant, high-quality proposals.

Technical considerations

From a technical perspective, several key considerations are important:

1. **Semantic Understanding of Texts**

   Project abstracts and call descriptions vary in length, detail, and terminology. Keyword-based matching alone may fail to capture the nuanced meaning of a proposal. Therefore, semantic embeddings are required to represent both projects and calls in a vector space that preserves contextual meaning.

2. **Handling Class Imbalance**

   Many funding topics are underrepresented, while a few dominate the dataset. A naïve retrieval approach may bias recommendations toward popular topics. This necessitates careful preprocessing, filtering of overly generic programs, and evaluation metrics that account for imbalance.

3. **Integration of Multi-Modal Features**

   Beyond textual descriptions, additional metadata such as keywords and target years provide important signals. Incorporating these features effectively into the model requires designing multi-modal representations and ensuring consistent handling across both projects and calls.

4. **Scalability and Efficiency**

   With thousands of projects and topics, the system must efficiently compute similarity scores. Precomputing embeddings and leveraging approximate nearest neighbor search techniques (e.g., FAISS) are critical to ensure low-latency retrieval.

5. **Evaluation without Traditional Supervision**

   The goal is retrieval rather than predictive modeling, so the project does not rely on train/test splits in the classical supervised sense. Instead, evaluation focuses on ranking-based metrics such as Top-K accuracy and mean reciprocal rank (MRR) to assess whether relevant topics appear among the top recommendations.

6. **Practical Application Considerations**

   Users may input project descriptions of varying length, include optional keywords, or specify a target year. The system must handle these variations gracefully, maintaining robustness in a real-world deployment context.

7. **Extensibility and Future-Proofing**

   Funding calls are updated regularly, and new calls are published continuously. The design of the system must allow for incremental updates, embedding recomputation, and potential integration with more advanced re-ranking models (e.g., cross-encoders or LLM-based scoring) in the future.

This report documents the project workflow, including data acquisition via web scraping, preprocessing and feature engineering, exploratory analysis, semantic embedding construction, and the deployment of a user-facing web application. By addressing these technical considerations, the system demonstrates a proof-of-concept for AI-assisted grant matching that is robust, scalable, and practically useful for end users.

This report also outlines a roadmap for potential future enhancements, including user-driven filtering, cross-encoder re-ranking, and additional feature integrations, to further improve the system’s precision and usability.

Notebooks

00 Web Scraping Process (cf. Notebook 00_Scraping.ipynb)

Intro

1. project.csv (https://data.europa.eu/data/datasets/cordis-eu-research-projects-under-horizon-europe-2021-2027?locale=en)

   This csv contains all projects that had been confirmed for funding for the 2021-2027 period so far.

   1. The data used for this project was **downloaded on July 28th, 2025**. Since then, there might have been updates, as new projects might have been approved.
   2. This data forms the basis for our project along with the topic descriptions that were accessed via web scraping.

2. **Other data**

   The 'cordis-HORIZONprojects-csv' file also contains other data, like for example the topics.csv. But this csv only contains the project id, the related topic id like "HORIZON-MSCA-2024-SE-01-01" and the title of the topic like "MSCA Staff Exchanges 2024". **Unfortunately, there is no other topic data available in this file** that would be relevant for our matching project.

3. **RSS feed**

   In a second attempt to get more data on the topics, especially the topics descriptions, we found a topic-related RSS feed that would be an ideal source as it is constantly up to date. But, unfortunately, this feed only contains more administrative information about the topics and not the topic descriptions.

   1. Link to RSS-feed: https://ec.europa.eu/info/funding-tenders/opportunities/data/referenceData/callupdates-rss.xml

The web scraping process

Since neither the official topics.csv file nor the RSS feed provided sufficient descriptive information about the funding topics, we decided to extract the missing details directly from the European Commission's Funding & Tenders Portal.

For this task, we built a reproducible web scraping workflow in Python, documented in the Jupyter Notebook 00_Scraping.ipynb. The main steps were:

1. **Tools and Libraries**
    - **requests** was used to programmatically access the topic web pages and download their HTML content.
    - **BeautifulSoup (bs4)** was employed for parsing and navigating the HTML structure. This allowed us to reliably locate and extract specific elements of interest such as the *expected outcome* and *scope* sections.
    - **pandas** provided a convenient way to store and organize the scraped data in tabular form, aligning it with the project metadata available in the CSV files.
    - **tqdm progress bars** were added to monitor scraping progress across multiple topic pages.
2. **Scraping logic**
    - **Each project record contains a topic identifier (e.g., HORIZON-CL3-2024-CS-01-02).**
    - Based on this ID, the corresponding topic webpage on the Funding & Tenders Portal was constructed and requested.
    - Within the HTML, the scraper specifically targeted sections containing the “Expected outcome” and “Scope” paragraphs, which are key to understanding the content and objectives of each topic.
    - Text cleaning routines were applied to remove HTML artifacts (e.g. line breaks, tags, buttons such as “Show less”), leaving only the relevant descriptive text.
3. **Output**
    - The collected descriptions were matched back to their respective topic IDs and merged with the existing project-level dataset.
    - This enriched dataset now contained not only project metadata but also comprehensive textual information about the associated funding topics.
    - Finally, the data was stored in CSV format for subsequent preprocessing and exploratory analysis.
4. **Robustness considerations**

  - Since websites can change their structure, the scraper was designed with flexible selectors and error handling to skip or log pages that could not be parsed.
  - A delay between requests was also introduced to avoid overloading the server and to comply with responsible scraping practices.

Reflection on Limitations and Assumptions

1. **Topic Descriptions:**

   The web scraping approach for collecting topic descriptions was based on the assumption that these pages consistently follow the same structure. While error handling and text cleaning techniques were applied, the scraper's reliance on specific HTML elements means that any future structural changes to the website could break the scraping process, requiring adjustments. Additionally, scraping might miss some nuance or context that would be available through more direct data sources.

   Future Consideration:

   The web scraping process currently only covers **historical data**, and the format of future data could change. Therefore, for the continued use of the **project-topic (topic=tender) matching tool**, a **stable solution and scraper logic** is needed. This would ensure consistent data collection over time and avoid disruptions due to changes in the source website's structure.

01 Data Cleaning Rationale (cf. Notebook 01_Data_Cleaning.ipynb)

The data cleaning process was designed to prepare two different outputs from the same raw dataset, each serving a specific purpose. The work was carried out in Python within Jupyter, using the following tools and techniques:

- **pandas** for efficient data manipulation and handling.
- **regex** for text pattern matching and cleaning.

  ast for safely evaluating stringified lists into Python lists.

ML-ready dataset (cleaned_project_data.csv)

- This version is the strict, filtered dataset intended as the foundation for downstream machine learning models.

Filtering logic: Very generic funding schemes (e.g., *ERC*, *HORIZON-MSCA*, *HORIZON-WIDERA*, *HORIZON-EIC*, *HORIZON-EIE*) are removed to reduce noise and avoid models overfitting to broad, uninformative categories.

  - The reasoning for this filtering is detailed further in the EDA Notebook and was added retrospectively based on insights from exploratory analysis and the initial matching results.

- **Column restriction:** Only the essential columns required for modeling are preserved — identifiers, enriched text fields, normalized keywords, and one-hot encoded year indicators.
- **Text processing techniques:**
  - Titles and objectives are standardized via regex (removing special characters, normalizing whitespace, converting to lowercase).
  - Topic descriptions are cleaned to remove boilerplate headers (“Expected outcome:”, “Scope:”) and UI artifacts such as *“Show less”*.
  - Keywords are transformed into normalized lists and stored alongside a stringified representation for flexibility.
- **Feature engineering:**
  - Composite text fields were created (e.g., project_text_simple, project_text_keywords, topic_text) to provide different levels of granularity for later NLP tasks.
  - Year information is extracted from topic identifiers and encoded into binary one-hot features.
- This stricter scope increases data consistency, reduces variance introduced by generic categories, and ensures that the training signals are more reliable.

EDA dataset (cleaned_project_data_eda.csv)

- This version retains **all rows and all original columns** from the raw input to support exploratory data analysis.
- **No filtering** is applied, so it provides the fullest possible context for descriptive statistics, profiling, and visualizations.
- It contains the same cleaned and enriched fields as the ML-ready version, but also preserves all original raw columns for transparency.

This design allows stakeholders to explore the dataset broadly, identify patterns, and better understand the original data structure before narrowing down to the modeling subset.

Design Decision

By producing both outputs, the pipeline addresses two competing requirements:

- A **noise-reduced, consistent dataset** for machine learning.
- A **comprehensive, transparent dataset** for exploratory data analysis.

This dual-path strategy ensures robustness for downstream modeling and flexibility for data exploration, while keeping the implementation in a single coherent notebook.

Reflection on Limitations and Assumptions

1. **Filtering Assumptions:**

   The decision to filter out broad funding schemes (e.g., *ERC*, *HORIZON-MSCA*) was made based on the assumption that these categories represent overly generic project types, which could introduce noise into the models. This exclusion was made **in retrospect** after reviewing the results from the exploratory data analysis (EDA) notebook. As the EDA was intended to provide an unfiltered overview of the data, **this filtering was not included in the EDA step or in the ML preparation pipeline**, allowing for a more comprehensive exploration of the topics before narrowing down the dataset. However, while the filtering step reduces noise for ML modeling, it may also lead to the exclusion of potentially valuable insights within these generic categories.

   Future Consideration:

   There could be a more sophisticated way of handling these generic topics in future work. For example, **generic topics** might benefit from a **question-based filtering system** where users answer questions such as "Are you a single researcher or a team of researchers?" or "Are you building on an existing product or is this a new idea?" These types of questions could help break down larger, more generic topics into more specific use cases, allowing them to be matched alongside smaller, more semantically relevant projects. This idea will require further exploration and could be integrated into future versions of the matching tool.

02 Exploratory Data Analysis and Baseline Comparison (cf. Notebook 02_EDA&Baseline.ipynb)

Introduction

The exploratory data analysis (EDA) was conducted on the full cleaned dataset (cleaned_project_data_eda.csv) to provide a detailed overview of the projects and topics, assess structural patterns, identify potential issues, and inform downstream semantic matching. This dataset contains all original fields plus enriched text features (project_text_simple, project_text_keywords, topic_text, keywords_clean) and one-hot year encodings.

Dataset Overview

- **Total projects:** 17,797
- **Unique topics:** 2,073
- **After filtering generic topics** (ERC, HORIZON-MSCA, HORIZON-WIDERA, HORIZON-EIC, HORIZON-EIE):
    - Projects: 4,429
    - Unique topics: 1,873

This demonstrates that a **substantial fraction of the dataset is dominated by generic programs**, justifying the need for filtering before any semantic matching.

Topic Distribution and Imbalance

- Distribution of projects per topic is highly skewed: the **median topic has only 2 projects**, while the top generic schemes contain hundreds to over a thousand projects.
- Grouping topics by their prefix (first 12 characters) highlights funding schemes that dominate the dataset, such as HORIZON-MSCA and ERC, and reveals patterns in the allocation of projects across program types.
- Binning projects by count per topic confirms the **extreme imbalance**, which is a critical consideration for model design: supervised models need enough examples per class for reliable training.

Missing Values

- The keywords_str field is empty for 1,543 projects (mostly ERC and POC projects), while other fields, including the main text fields and year indicators, are complete.
- Understanding the distribution of missing keywords is important because they contribute to the composite text features used in semantic matching.

Text Length Analysis

- project_text_simple averages ~1,916 characters, while topic_text averages ~5,200 characters, indicating **topic descriptions are substantially longer**.
- The difference in text length can influence similarity computations, especially when using TF-IDF, as longer topic descriptions contain richer contextual information.

Project Duration and Financial Characteristics

- Project durations range from 182 days to 3,652 days (approximately 6 months to 10 years), with an average of ~1,199 days.
- totalCost has many zero values (8,636 projects), while ecMaxContribution shows wide variance.
- These characteristics may inform future filtering or weighting decisions, though for text-based semantic matching they are currently secondary.

Baseline Comparison

To assess the impact of filtering and the informativeness of different text representations, baseline matching experiments were conducted:

- **Random baseline (Top-10):** selects topics at random
- **TF-IDF + Cosine similarity:** computed for both project_text_simple and project_text_keywords

Dataset	Text Variant	Top-1 Accuracy	Top-10 Accuracy
Filtered	project_text_simple	0.561	0.855
Filtered	project_text_keywords	0.579	0.871
Unfiltered	project_text_simple	0.156	0.258
Unfiltered	project_text_keywords	0.161	0.261

Observations and Nuances

1. **Impact of filtering:** Removing generic funding schemes dramatically increases both Top-1 and Top-10 accuracy. Without filtering, the correct topic is rarely the top prediction and often does not appear within the top 10.
2. **Text representation matters:** Including keywords (project_text_keywords) provides only a small improvement over project_text_simple in this baseline. Most semantic signal is already captured in the project title and objective fields.
3. **Top-1 vs. Top-10:** While Top-10 accuracy is high (~85–87%) in the filtered dataset, Top-1 accuracy (~56–58%) indicates substantial room for improvement. The correct topic is often among the top recommendations but not consistently the top choice.
4. **Residual generic topics:** Some broad topics may remain even after filtering, which could slightly influence similarity metrics, but the major sources of noise have been removed.
5. **Dataset imbalance:** The high variance in projects per topic means rare topics may be underrepresented in similarity computations. Future models may require weighting or sampling strategies to ensure equitable coverage.

Note on Keywords and Future Use

Although adding keywords to the project text produced only a small improvement in this baseline, they will be leveraged as a **separate vector** in the semantic matching pipeline. Using keywords independently may enhance Top-1 ranking performance beyond what is currently observed.

Practical Considerations and Next Steps

- Filtering generic topics is **essential** to reduce noise and improve baseline semantic matching performance.
- The Top-10 metric demonstrates that the correct topic is usually included among the top recommendations, but the **main focus of the next step will be increasing Top-1 accuracy**, ensuring the correct topic appears as the top prediction.

03 Semantic Matching with Multi-Modal Embeddings (cf. Notebook 03_Semantic_Matching.ipynb)

Overview

The goal of this step is to match projects to their corresponding funding topics using **semantic embeddings**. Each project and topic is represented as a **multi-modal vector**, combining:

- Project/Topic text descriptions
- Keywords
- One-hot encoded year indicators

This approach allows us to perform **efficient large-scale retrieval** with FAISS and retrieve the top-K candidate topics for each project, balancing accuracy and computational feasibility.

Data and Preprocessing

- **Input:** cleaned_project_data.csv (ML-ready dataset)
- Each project contains:
    - Cleaned title and objective (project_text_simple)
    - Keywords (keywords_clean)
    - Topic description (topic_text)
- Keywords are safely parsed with ast.literal_eval to avoid unsafe eval usage.
- Year information is included as a binary one-hot vector to incorporate temporal relevance.
- The topic dataset is deduplicated to provide unique topic vectors for retrieval.

This preprocessing ensures **consistent multi-modal feature vectors** for downstream semantic matching.

Model Choice and Rationale

- **Bi-Encoder:** all-MiniLM-L6-v2
    - Chosen for **speed, scalability, and robustness** to longer concatenated inputs (text + keywords + years).
    - Produces embeddings that can be efficiently searched with FAISS.
- **Alternative Tested:** multi-qa-mpnet-base-dot-v1

  - Showed similar performance on small-scale tests.
  - Could not handle **longer multi-modal inputs** reliably in the application context.
- **Cross-Encoder:** cross-encoder/ms-marco-MiniLM-L-6-v2
  - Used here only for a **small batch (50 projects)** as a demonstration due to computational expense.
  - Performance is lower than the bi-encoder for Top-10 retrieval in this small sample.

Embedding Construction

1. Project text embeddings
2. Topic text embeddings
3. Aggregated keyword embeddings for projects and topics
4. Year vectors
5. Concatenation into a single **multi-modal vector**
6. L2-normalization for FAISS inner-product similarity search

FAISS allows **efficient retrieval of top-K candidates**, enabling practical use for thousands of projects and topics.

Evaluation Metrics

FAISS Bi-Encoder (Full Dataset):

Metric	Value
Top-10 Accuracy	0.9564
Top-1 Accuracy	0.8905
MRR@10	0.9148

Cross-Encoder (latest sample with only 50 projects for demo purposes):

Metric	Value
Top-10 Accuracy	0.6400
Top-1 Accuracy	0.3200
MRR@10	0.4186

- The cross-encoder metrics are based on a **small illustrative batch only**.
- Larger batches improve reliability but are computationally expensive. In other runs with larger batches of 1000 projects in total, the results were not as low as the current demo batch, but still not as good as the bi-encoder results.
- Despite prior tests with larger batches, the bi-encoder still provides **better Top-10 retrieval efficiency** and is suitable for production use.

Analysis

1. **High Top-10 Accuracy (0.9553):** The bi-encoder retrieves highly relevant topics consistently.
2. **Top-1 Accuracy (0.8923):** Strong performance in ranking the correct topic first.
3. **Keyword and Year Integration:** Combining keywords and temporal information improves semantic alignment.
4. **Cross-Encoder Limitations:** Slow pairwise scoring and small-batch demonstration show that it is impractical for full dataset evaluation.
5. **Scalability:** The bi-encoder + FAISS approach scales efficiently to the full dataset (~18k projects × ~1.8k topics).

Limitations and Considerations

- **No train/test split:** The embeddings are pre-trained, so the focus is **proof-of-concept retrieval**, not predictive modeling.
- **Cross-Encoder:** Could be applied to **re-rank top candidates** in future iterations for marginal Top-1 improvements.
- **Model Alternatives:** Larger embeddings or domain-specific models could improve accuracy at the cost of computation.
- **Keyword Handling:** Different aggregation or weighting strategies could refine multi-modal similarity.
- **Dynamic Topic Updates:** New topics would require embedding updates for consistent retrieval performance.

Conclusion

- The pipeline demonstrates a **robust, scalable semantic matching system** for projects and funding topics.
- Multi-modal embeddings allow efficient retrieval and maintain high Top-10 and Top-1 accuracy.
- **Bi-encoder is the practical choice** for a proof-of-concept due to efficiency, scalability, and multi-modal support.
- Cross-encoder results highlight computational trade-offs and provide a benchmark for future improvements.
- Overall, this approach is sufficient to **demo a live application** while leaving room for future refinement, including keyword weighting, cross-encoder re-ranking, and alternative embedding models.

4. Application & User Interface (UI) Implementation

Overview

The application provides a user-friendly interface for project proposers to retrieve relevant Horizon Europe topics based on multi-modal semantic embeddings. The system combines **project text, keywords, and target year information** to generate a vector representation for each project and perform efficient similarity search against precomputed topic embeddings using **FAISS**.

The application is designed to be **scalable, responsive, and intuitive**, enabling both web-based interaction through HTML forms and programmatic access via a JSON API.

Backend Architecture

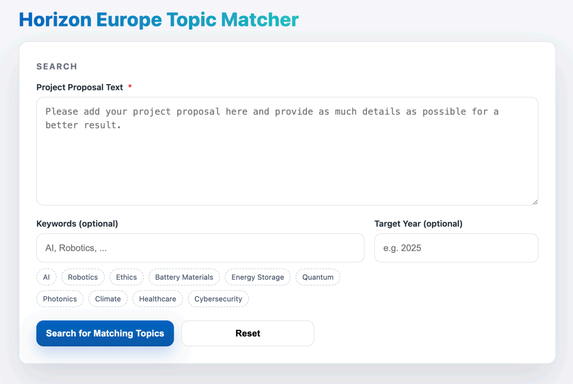
1. **Model and Embedding Loading**
    - **Sentence Transformer Model**: all-MiniLM-L6-v2 is used to encode project text and keywords into dense vector embeddings.
        - Originally, multi-qa-mpnet-base-dot-v1 was tested and produced comparable results in offline evaluation. However, when handling longer project texts with additional multi-modal features (keywords + year vectors), it caused runtime issues. MiniLM was chosen for **robustness and scalability**.

- **FAISS Index**: Precomputed topic embeddings (text + keywords + year vectors) are combined and normalized for **cosine similarity search** using an inner-product index.
- **Resource Loading**: All embeddings and metadata are loaded **once at application startup** to minimize latency during user requests. The topic embeddings are cast to float32 to prevent FAISS errors and ensure numeric consistency.

2. **Multi-Modal Search Logic**
    - **Text Embeddings**: The project proposal is encoded as a dense vector.
    - **Keyword Embeddings**: Keywords entered by the user are separately encoded and concatenated with the text embeddings.
    - **Year Encoding**: Optional project year is represented as a one-hot vector to provide temporal relevance.
    - **Vector Combination**: Text, keyword, and year vectors are concatenated in a consistent order matching the topic embeddings in the FAISS index.
    - **Normalization**: The combined vector is L2-normalized to align with the FAISS inner-product similarity metric.
    - **FAISS Retrieval**: Top-K most similar topics are retrieved efficiently using the prebuilt FAISS index.
3. **API Endpoints**
    - **JSON API (/search-topics)**: Provides programmatic access for scripts or external tools.
    - **HTML Form (/search)**: Offers a web interface where users input project text, keywords, and year.
    - **Index Page (/)**: Serves a clean and responsive form for new searches.
4. **Memory Management**
    - Garbage collection (gc.collect()) is invoked after loading embeddings and building the index to reduce memory footprint in long-running sessions.

Frontend Design

Screenshots:



Horizon Europe Topic Matcher

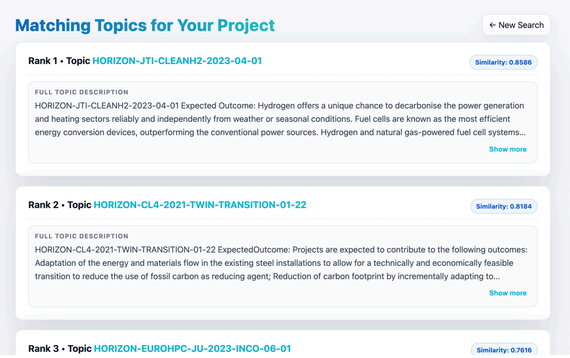
SEARCH

Project Proposal Text *

Please add your project proposal here and provide as much details as possible for a better result.

Keywords (optional): AI, Robotics, ...

Target Year (optional): e.g. 2025



Matching Topics for Your Project

Rank 1 • Topic HORIZON-JTI-CLEANH2-2023-04-01 Similarity: 0.8586

FULL TOPIC DESCRIPTION

HORIZON-JTI-CLEANH2-2023-04-01 Expected Outcome: Hydrogen offers a unique chance to decarbonise the power generation and heating sectors reliably and independently from weather or seasonal conditions. Fuel cells are known as the most efficient energy conversion devices, outperforming the conventional power sources. Hydrogen and natural gas-powered fuel cell systems...

Rank 2 • Topic HORIZON-CL4-2021-TWIN-TRANSITION-01-22 Similarity: 0.8184

FULL TOPIC DESCRIPTION

HORIZON-CL4-2021-TWIN-TRANSITION-01-22 ExpectedOutcome: Projects are expected to contribute to the following outcomes: Adaptation of the energy and materials flow in the existing steel installations to allow for a technically and economically feasible transition to reduce the use of fossil carbon as reducing agent; Reduction of carbon footprint by incrementally adapting to...

Rank 3 • Topic HORIZON-EUROHPC-JU-2023-INCO-06-01 Similarity: 0.7616

1. **Form Layout**
   - The input form includes:
     - **Project Proposal Text** (mandatory)
     - **Keywords** (optional, with suggested chips for quick selection)
     - **Target Year** (optional)
   - Responsive design ensures usability on both mobile and desktop devices.
2. **Interactive Keyword Chips**
   - Clicking a chip automatically adds the keyword to the input field, preventing duplicates and improving UX for users who may not know all relevant keywords in advance.
3. **Submission Workflow**
   - On submission:
     - Form validation ensures the project text is provided.
     - The submit button shows a **loading spinner** to indicate progress.
     - Results are fetched asynchronously via POST requests, allowing smooth updates without full page reloads.
4. **Results Display**
   - Each result card shows:
     - Topic ID (linked to the official Horizon Europe page)
     - Similarity score

    - Full topic description with a **3-line preview**
    - “Show more/less” toggle for expanding long descriptions
  - For the **top result**, optional LLM-generated summary or match analysis can be displayed if enabled in the future.

5. **Responsive & Accessible Styling**
   - CSS variables define brand colors, shadows, and typography for a cohesive visual identity.
   - Cards and buttons have **rounded corners and soft shadows** for better readability.
   - Mobile-first grid layout adapts to smaller screens, stacking fields vertically.

Technical Considerations

1. **Performance**
   - By precomputing and caching embeddings, the application avoids repeated encoding overhead.
   - MiniLM embeddings are **memory-efficient (384 dimensions)** while retaining semantic quality.
   - FAISS allows **millisecond-scale retrieval** even with thousands of topics.
2. **Scalability**
   - New topics can be added by embedding them and appending to the FAISS index.
   - The separation of project text, keyword embeddings, and year vectors allows incremental updates without recomputing the entire index.
3. **Extensibility**
   - Placeholder LLM explanation blocks indicate potential future integration for automatically generating human-readable summaries and match analyses.
   - The API design enables additional endpoints (e.g., batch search) without major code changes.
4. **Limitations**

- Cross-encoder approaches were considered for re-ranking but are computationally expensive. Mini-batch tests show worse or inconsistent results on small samples, so they are not applied in production.
- No user authentication or rate-limiting is implemented yet; the application is currently **proof-of-concept**.

Conclusion

The application provides a **robust, efficient, and user-friendly interface** for matching project proposals to Horizon Europe topics. By leveraging **multi-modal embeddings**, precomputed FAISS indices, and a responsive frontend, the system delivers:

- High-quality semantic matches in real time
- Clear and interactive UI for project proposers
- Extensibility for additional features such as LLM-based summaries or enhanced keyword weighting

Overall, this implementation demonstrates a **proof-of-concept for live deployment**, balancing computational efficiency with usability, and forms a solid foundation for future refinement and scaling.

5. Future Roadmap and Enhancements

While the current implementation demonstrates a robust, scalable semantic matching system for Horizon Europe project proposals, several avenues exist to enhance accuracy, usability, and practical utility. These enhancements form a roadmap for future work:

1. Hybrid User-Driven Filtering

Currently, all recommendations rely purely on multi-modal embeddings. A hybrid approach could significantly improve precision by allowing users to pre-select high-level filters, such as funding type (e.g., MSCA, ERC, Cluster projects) or thematic area. The system would then perform semantic retrieval within this smaller, relevant subset, reducing noise from unrelated programs and improving Top-1 accuracy.

2. Lightweight Re-Ranker

A Cross-Encoder model could be applied to the top-N candidates (e.g., Top-20) from the Bi-Encoder retrieval. This lightweight re-ranking would fine-tune the final ordering of topics, enhancing Top-1 recommendations without incurring the prohibitive computational costs of full pairwise scoring across all topics.

3. UI Improvements (Demo Stage)

While the current UI demonstrates the core functionality, several improvements could enhance usability:

- Separating HTML structure, CSS styling, and JavaScript logic for maintainability and scalability.
- Interactive filtering panels for funding type, domain, or country.
- Improved keyword selection and suggestion chips with auto-complete.
- Progressive loading of results with a smoother experience on long topic descriptions.
- Mobile-first enhancements, ensuring responsive layout and accessibility compliance.

4. LLM-Based Explanations with Cost Control

Integrating an LLM could provide concise, human-readable reasoning for each recommended topic, explaining why a project matches a specific call. Cost control mechanisms (e.g., query caching, batched requests, or small token-limited prompts) ensure that these explanations remain feasible in production environments.

5. Leveraging Additional Data for Enhanced Matching

Beyond semantic similarity, other historical and administrative data could inform more sophisticated matching:

- Historical funding patterns: amount funded per topic, type of projects funded, acceptance rates.
- Temporal trends: likelihood of funding based on previous project characteristics or year-specific priorities.
- Supplementary administrative information from RSS feeds or other EU tender portals to enrich scoring logic.

  By incorporating these signals, the system could evolve toward predictive recommendations, estimating not only the relevance of a call but also the likelihood of successful funding.

6. Dynamic and Continuous Updates

Horizon Europe calls are continuously updated. Future enhancements include:

- Automated ingestion of new calls via the RSS feed or other official sources.
- Incremental embedding updates for both projects and topics to ensure the model remains current.
- Integration with a monitoring pipeline to detect shifts in topic structure or content that may affect retrieval quality.

7. Roadmap Summary

These planned improvements, collectively, aim to transform the prototype into a more practical, accurate, and user-friendly grant matching system. They balance computational efficiency, interpretability, and real-world applicability, while maintaining the modularity required for future extensions such as cross-encoder re-ranking, LLM reasoning, and predictive scoring.

6. Conclusion

This project demonstrates the feasibility and effectiveness of an AI-powered system for matching research project proposals to relevant Horizon Europe funding calls. By leveraging multi-modal embeddings that combine textual descriptions, keywords, and temporal information, the system achieves high Top-1 and Top-10 accuracy, consistently retrieving the most relevant funding topics for a given proposal. The integration of FAISS-based retrieval ensures scalability and low-latency performance even when handling thousands of projects and topics.

Key findings include:

1. **Semantic Embeddings are Effective:** Using a Bi-Encoder with multi-modal features significantly outperforms baseline TF-IDF or random retrieval approaches. Incorporating keyword and year information further improves ranking performance, particularly for Top-1 accuracy.
2. **Cross-Encoder Re-ranking Trade-offs:** While a Cross-Encoder can potentially refine rankings, applying it on small batches or top candidates demonstrates that the computational cost is high relative to the marginal improvement, justifying a lightweight, targeted approach.

3. **Importance of Data Preprocessing:** Filtering overly generic funding schemes and carefully cleaning project and topic texts are crucial for reliable performance. Historical insights from CORDIS projects enhance the model’s ability to capture nuanced semantic relationships.
4. **Robust, Scalable Pipeline:** The combination of precomputed embeddings, FAISS indexing, and a modular pipeline ensures that the system can handle incremental updates and new projects efficiently.

Beyond Horizon Europe, the methodologies developed in this project are applicable to other grant programs, innovation funding platforms, and research proposal ecosystems. Any context where applicants need to navigate large, heterogeneous, and frequently updated funding opportunities can benefit from a similar AI-assisted matching approach. Furthermore, by extending the system to integrate additional historical or administrative data, predictive scoring, and LLM-generated explanations, this framework can evolve into a decision support tool that not only identifies relevant opportunities but also estimates success likelihood and guides strategic planning for research and innovation projects.

Overall, this work provides a proof-of-concept for AI-assisted grant discovery, highlighting both the technical and practical potential of semantic search and multi-modal embeddings in complex real-world retrieval tasks. With continued development along the roadmap outlined, the system could become a powerful, widely applicable tool for universities, SMEs, consultants, and other organizations navigating competitive funding landscapes.