Library of Congress Labs

Toward Piloting Computational Description

Final Presentation

Background

Introduction

LC Labs, a division of the Library of Congress (Library) Digital Strategy Directorate (DSD) in the Office of the Chief Information Officer (OCIO), leads a **program of experimentation** that includes user-centered research, prototyping and development of emerging methods, workflows and functionalities that **connect** Library collections, data, services and expertise **to users in new ways**.

This experiment, ***Toward Piloting Computational Description*** is a successor to the 2023 ***Exploring Computational Description*** experiment.

Where are the most effective combinations of automation and human intervention in generating high-quality catalog records that will be usable at the Library of Congress?

What are the benefits, risks and requirements for building a pilot application for ML-assisted cataloging workflows?

Background

Requirements

Test, and report on at least three (3) machine-learning-assisted, HITL workflows or methods to generate full-level bibliographic records (whenever possible) from the textual and/or visual elements of ebooks in epub, PDF, or other digital formats.

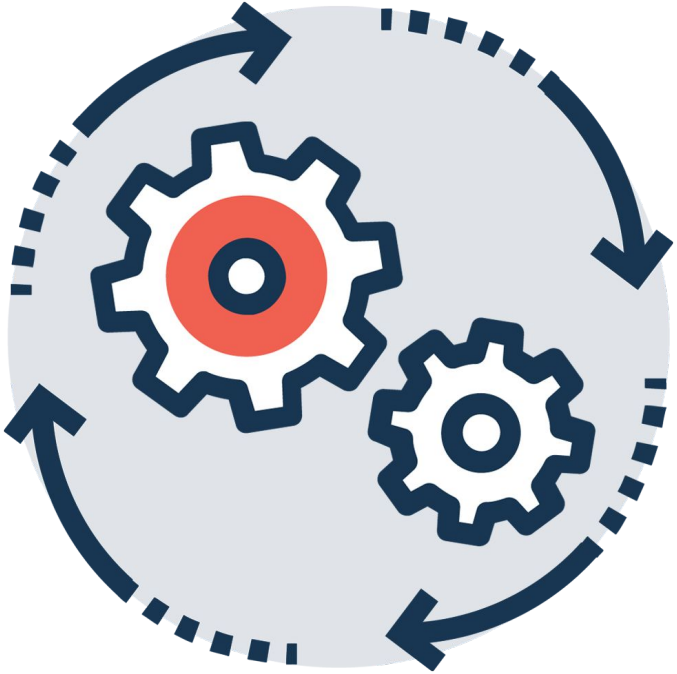
The minimum fields to be generated are: titles, author names, unique identifiers, date of issuance, date of creation, genre/form and subject terms.

Approximately 20,000 existing MARC records and ebooks made available by the Library for training data for the previous experiment supplemented with a larger set of additional CIP cataloging ebooks also added this year.

The models and methods tested must use open-source models.

Introduction

Process



Experiment Process

Overview

Understand

Needs Analysis

Define the problem
Understand needs & motivation

Explore

Explore Data

Identify & understand data sources

Define

Selection Criteria

Metrics for success
Standards for evaluation

Select

Make Selections

Identify and document candidates

Test

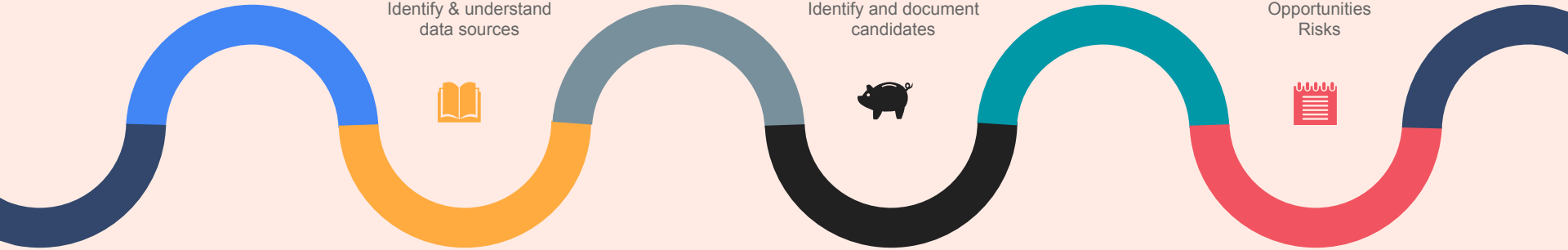
Prototype

Test and Refine

Review

Assess

Opportunities
Risks



Define the problem

- What are the priorities of the Library as an institution?
- Who are the users?
- What are their needs and motivations?
- What are the challenges for the Library?
- What are the challenges for users?
- **How can HITL prototypes address these problems?**

What data is available?

- Formats
- Statistical properties
- Relevant features
- Challenges with the data
- How balanced / unbalanced is the data?

Where are we?

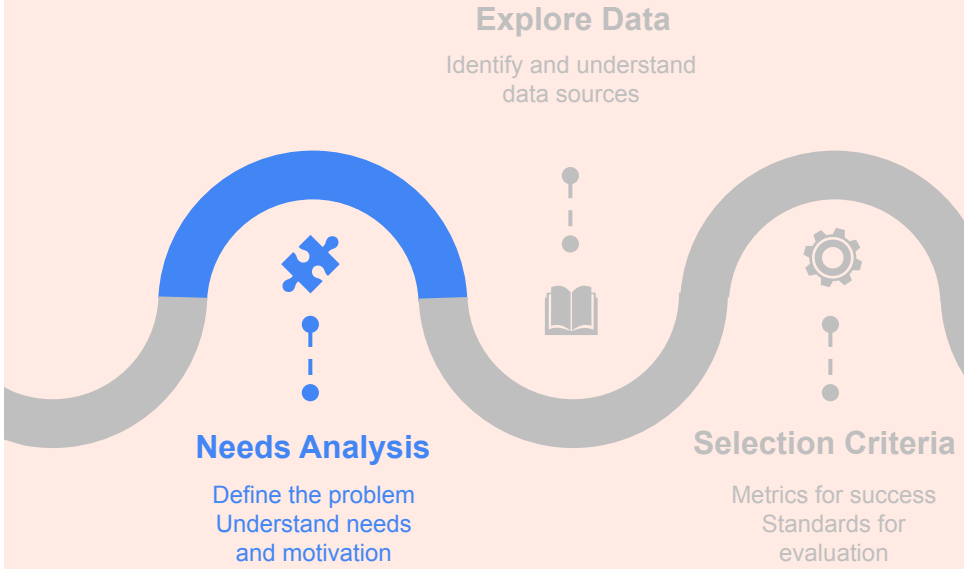
- Landscape analysis

- Assess candidates from landscape analysis:
  - **Suitability for data generation for HITL prototype ideas**
- **Select three (or more) for prototyping**

- Create:
  - Training data
  - Test and validation data
- Measure against selected metrics
- Train or fine-tune
- Repeat
- **Evaluate with end users**

- Evaluate against
  - Institutional requirements
  - User needs
- What are:
  - Benefits
  - Risks
  - Costs
  - Expected performance / benchmarks
- Which are the most promising approaches?

Needs Analysis



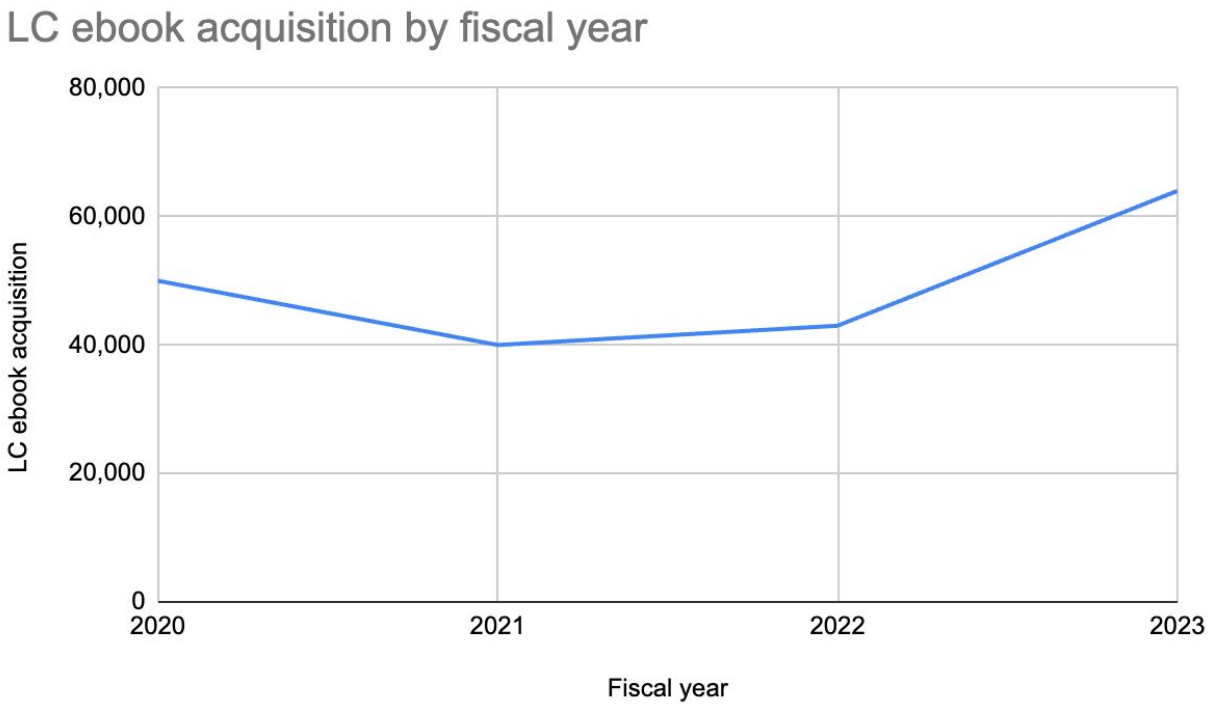
The previous experiment *Exploring Computational Description* was primarily focused on **evaluating and testing approaches to automated generation of catalog metadata**, rather than on a deeper exploration of the needs of users.

This experiment *Towards Piloting Computational Description* had a deeper focus on understanding and analyzing user needs and the priorities of the Library of Congress as an institution.

With a **core focus on testable HITL prototypes.**

Approach

1. Interviews with:
    a. Key senior stakeholders (Beacher Wiggins, Judith Cannan)
    b. Members of the cataloging team
2. Ebook cataloging workflow audit
3. Ideation, wireframing and review workshop(s)



EBook acquisitions over time

Simplified Workflow Outline

1. Study CIP request ebook briefly. Download or open the ebook file in CTS.
2. Search in Voyager for a record for this ebook. If any kind of record exists for the ebook version (not just the print that can be cloned), skip to step 25.
3. If no placeholder record exists, create a new ebook record from scratch by opening the ebook record template.
4. Assign a new bib LCCN to the record in the MARC 010 field.

Simplified Workflow Outline

5. Add the descriptive metadata for the ebook. Key in the title, authors, publication, series information, etc. Authority checking. The cataloger then looks at the authors and editors and checks Voyager to determine if the listed authors and editors already have name authorities in the search and discovery system.
    a. Names have to be differentiated by name authority record. The differentiation is aided by birth date, middle name, profession, etc.
6. Subject headings. Start the research for the LCSH – back to the ebook

Simplified Workflow Outline

7. Read and review the introduction and/or publisher's summary on Amazon or other website
    a. Translate the introduction or summary into LCSH
        i. Subject headings are recommended to be 10 or fewer
        ii. Generally attempt to have fewer than 5 subject headings
        iii. Minimum of 1 subject headings (non-fiction)
        iv. Fiction might not have an LCSH but it will have an LCGFT (Library of Congress Genre and Form Heading Terms). Novel is an acceptable LCGFT term if the introduction (or the summary provided by the publisher) is not terribly clear. Short Fiction if the form is relevant.
8. Search for similar works in Voyager – try to determine how other books have been cataloged in the past to encourage the collocation of books on similar topics.

Simplified Workflow Outline

8. Add the Class number. First two headings of the record need to reflect the class number of the text. This is the first part of the call number. The first part of the class number is the subject. Sometimes the class number will also have a cutter. Example: geographic regions, etc. Ebooks do not get shelflisted.
9. Verify that all fixed fields, especially MARC 006 and MARC 007, are accurate.
10. Review Record
    a. Established catalogers will review their own records
    b. Catalogers in training will have their records reviewed by their mentor/supervisor

Simplified Workflow Outline

11. Update specific MARC fields to indicate that the record is fully complete and ready to be distributed to OCLC.
12. Create holdings record with LCCN permalink
13. Notify DCMS that the record is ready and that the ebook can be moved permanently into Stacks
14. Assign Dewey Decimal System number – send the LCCN to the Dewey section where they will look up the record in Voyager and assign the Dewey number.The update to the Voyager record will overlay the original record.

Institutional Priorities

What should HITL (Human-in-the-Loop) prototypes driven by ML data be focused upon?

- Supporting the Library of Congress's e-preferred policy
- Reducing the backlog of ebooks to be cataloged
- Increasing access to ebooks through the Library of Congress catalog
- Increasing productivity of time for catalogers
- Relieving catalogers of routine steps
- Supporting the human catalogers and streamline their work

Catalogers' Priorities

Agreement with senior management that:

- Strong discovery is a key goal of cataloging
- Prototypes should aim towards:
  - Reduction or alleviation of tedious work
  - Reduction of human time required to catalog each ebook
  - Reduction of backlog of uncataloged ebooks

What we learned

- Subject cataloging is time consuming
  - Caution about whether ML workflows can produce accurate and comprehensive subject cataloging without additional expert review
- Hope that ML methods could suggest subjects that could then be reviewed by experts
- Access to similar or related works might facilitate review of subject suggestions by showing how similar works had been cataloged
- Class numbers / call numbers were of interest

What we learned

- Use of authority records was another potential pain point and/or opportunity for ML assistance
- “Self-evaluating AI” was of interest:
  - Transparent scoring of model confidence or other measures of potential quality
  - Information to guide catalogers in their review process

What we learned (last year)

- Evaluation of the user facing assisted cataloging prototypes suggested that:
  - Use of authority data was valuable
  - Review of data *in-context* was valuable
- More work is needed to iterate towards producing *full* bibliographic records as valid MARC (or BIBFRAME) via automated methods

Landscape (last year)

- The landscape of AI/ML is completely in flux at the moment.
- New frameworks and models are coming online all the time.
- These models and frameworks often have very basic, and immature, tooling for:
  - Integration with other workflows
  - Processing data at scale
  - Handling longer texts
  - Evaluation

Landscape (last year)

- It is not clear what the overall costs and performance of generative AI models might be
  - Compute cost for self-hosted solutions can be prohibitive
  - Per use costs for commercial APIs can be opaque
- Generative AI models may contain the data we are evaluating against *in their training dataset*
- Tooling or approaches that work for one generative AI model may not work for another, or even for a different version of the same AI model
- There are good prima facie reasons to believe there is a lot of potential for generating high quality bibliographic metadata using LLMs but reasons to be wary

Landscape (what has changed)

- The landscape of AI/ML is still completely in flux at the moment, but a number of big central players have emerged
- These models and frameworks have more mature tooling for:
  - Integration with other workflows
  - Processing data at scale
  - Handling longer texts
  - Evaluation
- It is not clear what the overall costs and performance of generative AI models might be
  - Compute cost for self-hosted solutions continue to be prohibitive
  - Per use costs for commercial APIs can be opaque but are much more transparent than previously, and are generally lower
- Generative AI models may contain the data we are evaluating against *in their training dataset* (and this continues to be a possibility)

What we recommended (last year)

- Generative AI approaches such as ChatGPT and others have much less mature ecosystems
  - Integration with existing workflows is via toolchains that are relatively new, lightly documented, or in flux
  - Quantitative evaluation against a wider range of data potentially needs exploration and additional testing and development
- Continue a program of regular engagement with the state of the art specifically as it interfaces with opportunities in automated description
- Deepen understanding of the ways in which automated description can be of immediate practical use in the workflows of expert catalogers and other library users and stakeholders

Several key differences from the previous experiment:

- Aim to produce full valid MARC records (rather than simple text extraction)
- Source data used in model training and evaluation was MARC on a subfield level rather than unstructured text.
- Take advantage of the changes in the AI/ML landscape to leverage generative AI and large language models
- Make fuller use of authority data
- Target specific pain points or opportunities in the ebook cataloging workflow

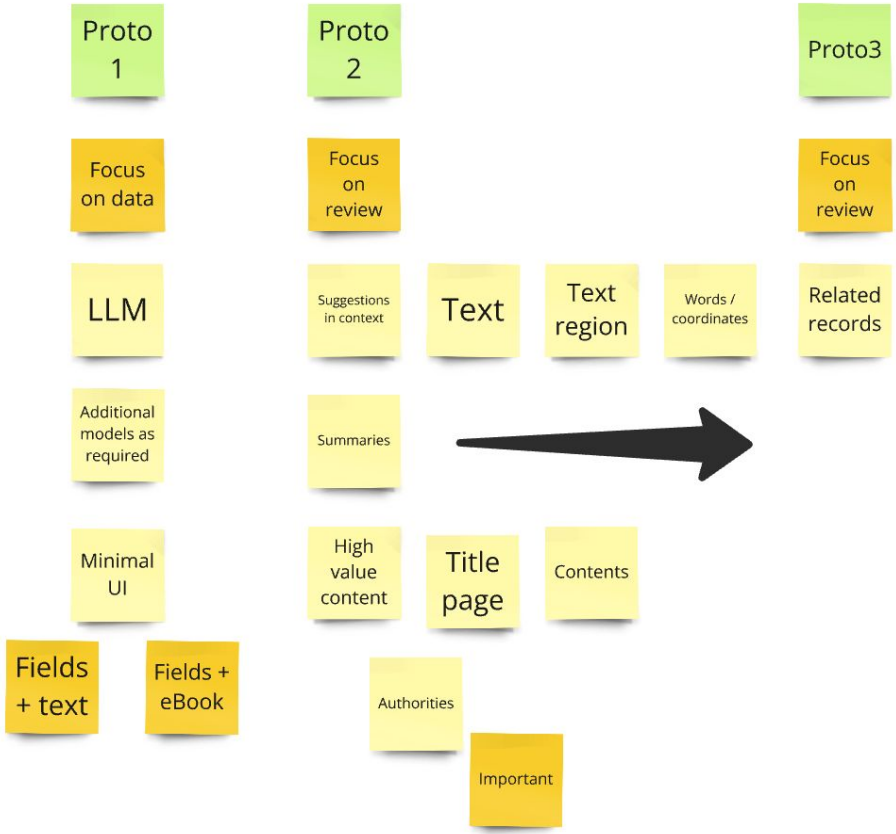
Prototype Ideas

Experiment was to produce:

- Three initial prototypes, plus
- One higher fidelity “clickable” prototype

However, we planned from the start to make the initial low-fidelity prototypes data driven with clickable user interfaces that use machine generated predictions.

The aim was to proceed by progressive enhancement building new features into each prototype and refining the prototype as we went incorporating feedback from testers.


Proto 1
Proto 2
Proto3
Focus on data
Focus on review
Focus on review
LLM
Suggestions in context
Text
Text region
Words / coordinates
Related records
Additional models as required
Summaries
Minimal UI
High value content
Title page
Contents
Fields + text
Fields + eBook
Authorities
Important

Prototype 1: Concept

- Display record form
- Display e-book PDF
- View ebook metadata
- View subject and summaries

Our goal was to create an intuitive and efficient user interface that could handle the display and review of large amounts of data that we could then gather cataloger feedback on to improve the UI as well as the generated data.

Prototype 2: Concept

- View Marc for each form field
- MARC XML tabs
- MARC 21 tabs

Our goal was to add more detailed data into the form page, such as MARC tabs and MARC records for each form field.

Record: 2021697918: From leadership theory to practice : a game plan for success as a leader /

Record | Subjects | Summaries

Title
From leadership theory to practice : a game plan for success as a leader /

Author
Robert Palestini

Date of publication
c2009

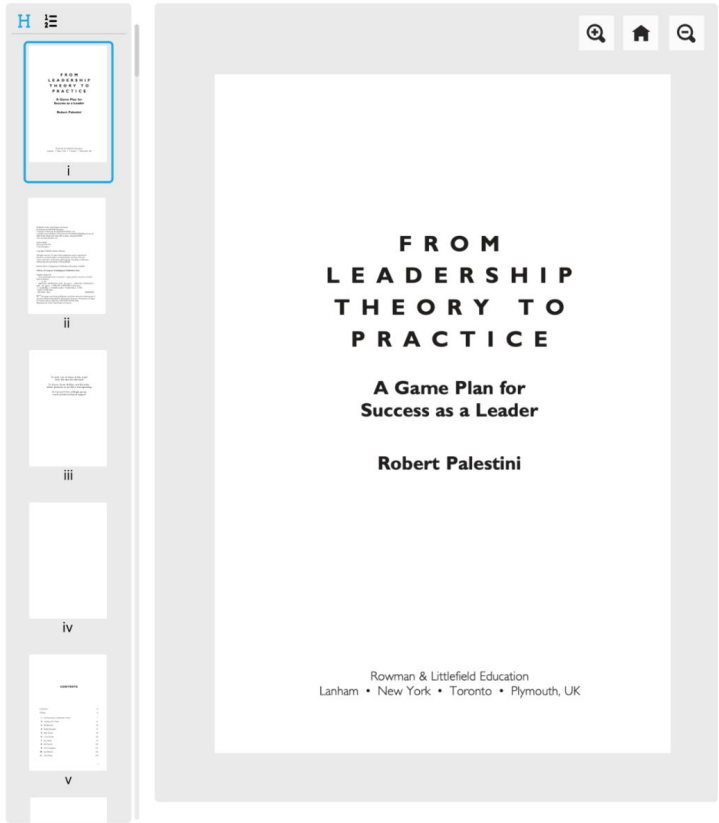
Publisher name
Rowman & Littlefield Education

ISBN
9781607090243

LCCN
HD57.7

Image | Metadata

page i of 200



Prototype 3: Concept

- Subject suggestions
- Name suggestions
- Suggestion confidence levels
- Pick and submit or reject suggestions
- View related items

Our goal was to add authority data, “self-evaluating AI” information, and related records to directly address the feedback gathered from catalogers in the initial interviews.

Record: 2021697918: From leadership theory to practice : a game plan for success as a leader /

Record | Subjects | Summaries

Title
From leadership theory to practice : a game plan for success as a leader /

Author
Robert Palestini

Date of publication
c2009

Publisher name
Rowman & Littlefield Education

ISBN
9781607090243

LCCN
HD57.7

Image | Metadata | Related records

< page i of 200 >

search for record

Aligning mind and heart : leadership and organization dynamics for advancing K-12 education / Chris Heasley, Robert Palestini.

Heasley, Chris, 1977-

Leadership | Educational leadership

Book LB2806 .H4155 2022

Aligning mind and heart : leadership and organization dynamics for advancing K-12 education / Chris Heasley, Robert Palestini.

Heasley, Chris, 1977-

Leadership | Educational leadership

Book LB2806 .H4155 2022

Aligning mind and heart : leadership and organization dynamics for advancing K-12 education / Chris Heasley, Robert Palestini.

Heasley, Chris, 1977-

Leadership | Educational leadership

Book LB2806 .H4155 2022

Record: 2021697918: From leadership theory to practice : a game plan for success as a leader /

Record | Subjects | Summaries

Title
From leadership theory to practice : a game plan for success as a leader /

Author
Robert Palestini

Date of publication
c2009

Publisher name
Rowman & Littlefield Education

ISBN
9781607090243

LCCN
HD57.7

Image | Metadata | Related records

page i of 200

back to related

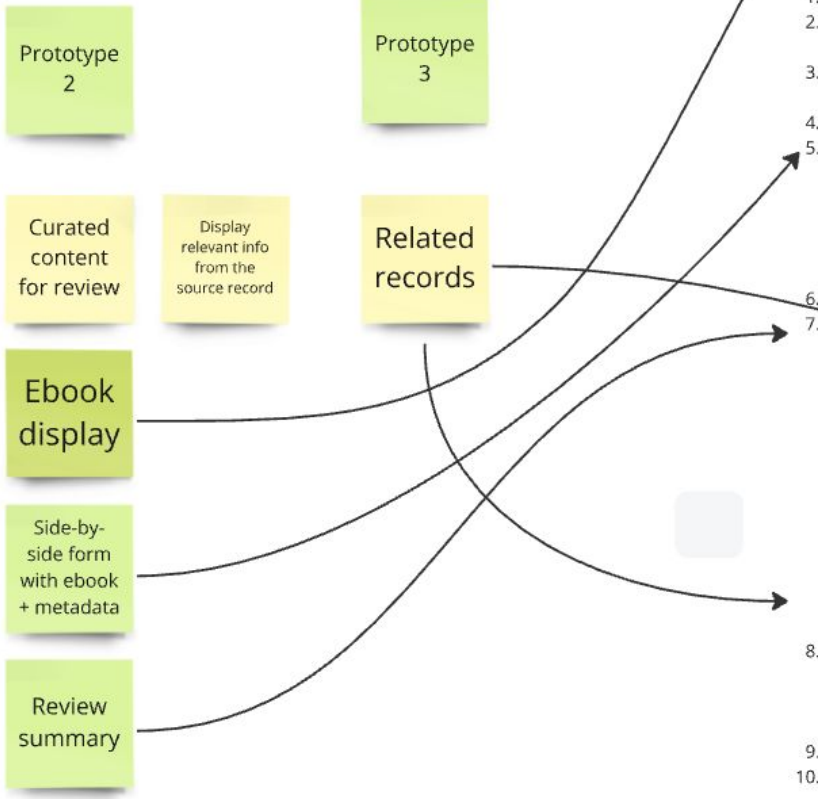
ligning mind and heart : leadership and organization dynamics for advancing K-12 education / Chris Heasley, Robert Palestini

Title page | Table of contents | Metadata | Summaries | Subjects

Keywords

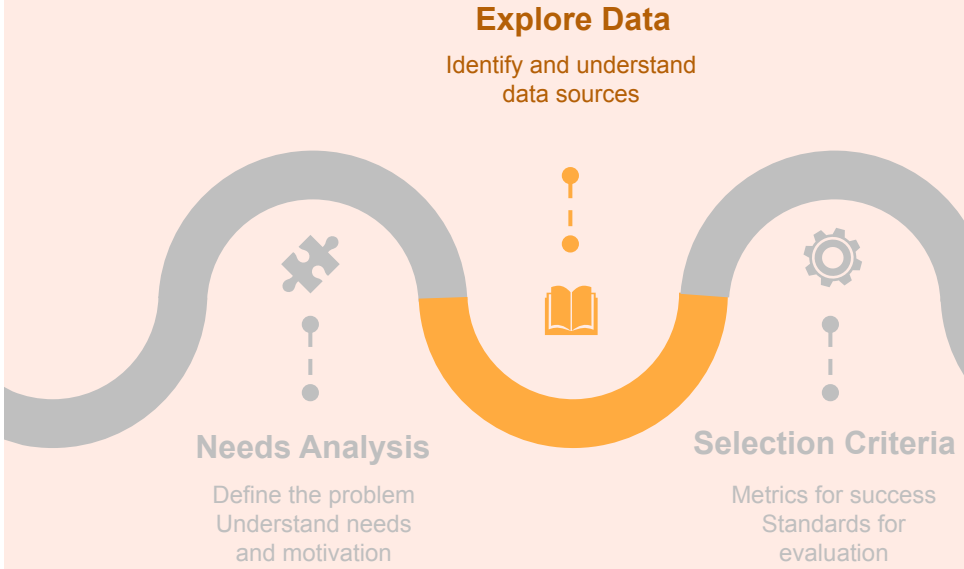
copy	text	score
☐	leadership behavior	6.12%
☑	human resource leadership behavior	5.97%
☐	structural leadership behavior	5.92%
☐	symbolic leadership behavior	5.85%
☐	political leadership behavior	5.84%
☑	symbolic frame leadership behavior	5.83%
☑	human resource behavior	5.83%
☑	coach	4.31%
☑	appropriate behavior	4.25%
☐	Players	4.19%

copy 5 subjects to 2021697918 Save discard



1. Study CIP request ebook briefly. Download or open the ebook file in CTS.
2. Search in Voyager for a record for this ebook. If any kind of record exists for the ebook version (not just the print that can be cloned), skip to step 25.
3. If no placeholder record exists, create a new ebook record from scratch by opening the ebook record template.
4. Assign a new bib LCCN to the record in the MARC 010 field.
5. Add the descriptive metadata for the ebook. Key in the title, authors, publication, series information, etc. Authority checking. The cataloger then looks at the authors and editors and checks Voyager to determine if the listed authors and editors already have name authorities in the search and discovery system.
    a. Names have to be differentiated by name authority record. The differentiation is aided by birth date, middle name, profession, etc.
6. Subject headings. Start the research for the LCSH – back to the ebook
7. Read and review the introduction and/or publisher's summary on Amazon or other website
    a. Translate the introduction or summary into LCSH
        i. Subject headings are recommended to be 10 or fewer
        ii. Generally attempt to have fewer than 5 subject headings
        iii. Minimum of 1 subject headings (non-fiction)
        iv. Fiction might not have an LCSH but it will have an LCGFT (Library of Congress Genre and Form Heading Terms). Novel is an acceptable LCGFT term if the introduction (or the summary provided by the publisher) is not terribly clear. Short Fiction if the form is relevant.
    b. Search for similar works in Voyager – try to determine how other books have been cataloged in the past to encourage the collocation of books on similar topics.
8. Add the Class number. First two headings of the record need to reflect the class number of the text. This is the first part of the call number. The first part of the class number is the subject. Sometimes the class number will also have a cutter. Example: geographic regions, etc. Ebooks do not get shelflisted.
9. Verify that all fixed fields, especially MARC 006 and MARC 007, are accurate.
10. Review Record

Explore Data



Exploring the data

The data provided for this experiment consisted of:

- Ebooks in PDF and EPUB format
- MARCXML records for each ebook

The e-books were from four collections:

- Open Access E-books
- Legal Reports
- E-Deposit Registration E-books
- Cataloging-In-Publication E-books

Many more additional Cataloging-in-Publication e-books were added to the dataset for this year's experiment.

What we learned (last year)

We also identified that Subject Classification was likely to be challenging

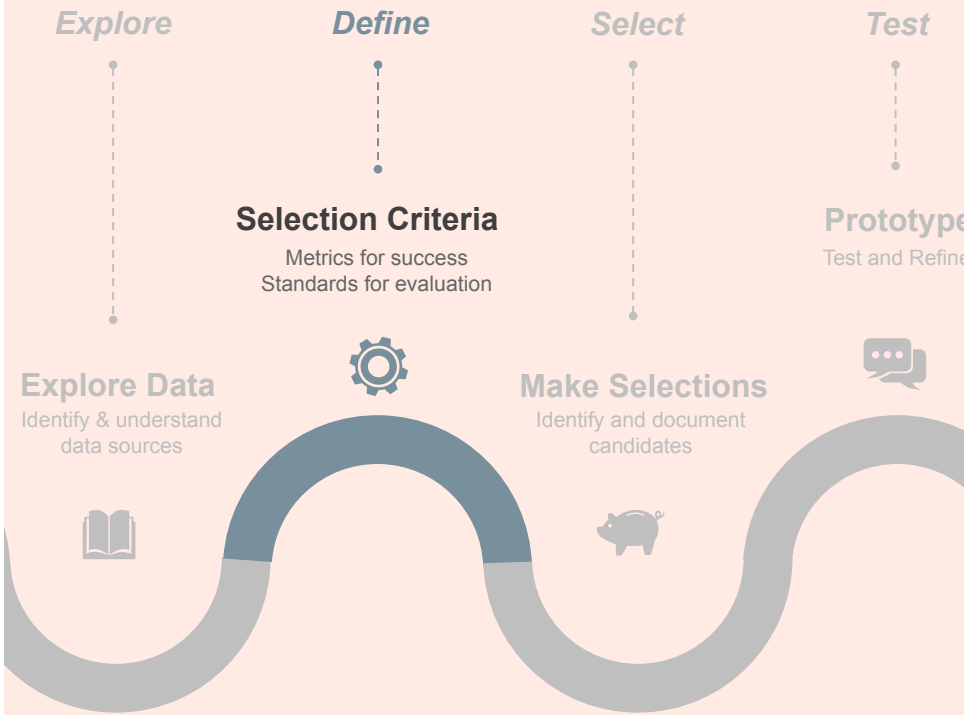
- The number of **subjects** to number of documents is high
- The number of **instances** of each individual subject are very low, with most subjects only appearing once in the entire corpus
- A very small number of **subjects** appear many times
- Subjects, as a whole, are very unbalanced across the entire dataset

Fields for identification

The following list includes a wider range of fields than were tested in the 2023 prototyping:

- 010: Library of Congress Control Number (LCCN)
- 020: International Standard Book Number (ISBN)
- 050: Call Number
- 082: Dewey Decimal Classification Number
- 100: Main Entry - Personal Name
- 245: Title Statement
- 264: Production, Publication, Distribution, Manufacture, and Copyright Notice
- 600: Subject: Personal Name
- 650: Subject Added Entry - Topical Term
- 651: Subject: Geographic Name
- 655: Genre
- 700: Added Entry - Personal Name

Selection Criteria



The potential array of machine learning tools, models, and workflows that can be applied to e-book texts is vast. Hundreds of tools are launched every year and **the academic literature is full of promising approaches for generating useful information from text**.

For the previous experiment in 2023, we were primarily interested in testing a broad spread of representative approaches.

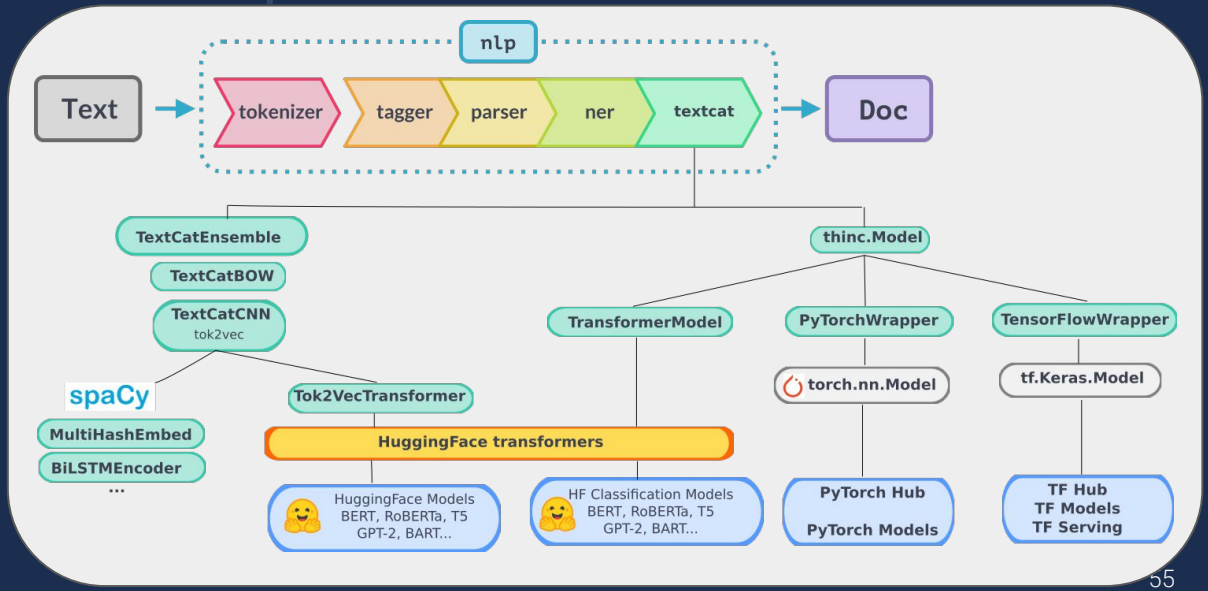
For this experiment, we were primarily focused on **identifying models that could deliver the data we need for the user facing HITL prototypes**.

What are we selecting?

Model

Often, when talking about models, we are referring to a **pre-trained model which has been trained on existing data** and which can then be used to make predictions on new data.

Many libraries support the download and reuse of existing models from hubs such as *Hugging Face*, or provide a suite of pretrained models which can be used as is or fine-tuned on specific data.



Architecture

Models are built on top of an architecture which defines how the model **accepts input**, how the model **is trained**, and how the model **produces output data**.

The architecture alone is **not** a *model* and the same architecture, such as the *Transformer* architecture may be the basis for hundreds of different models and/or libraries.

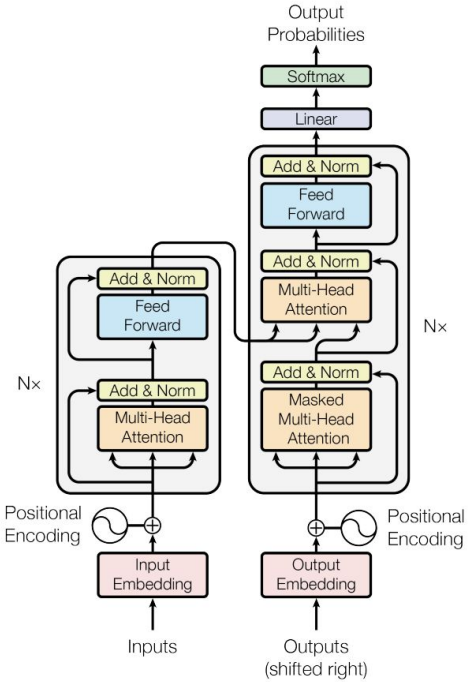
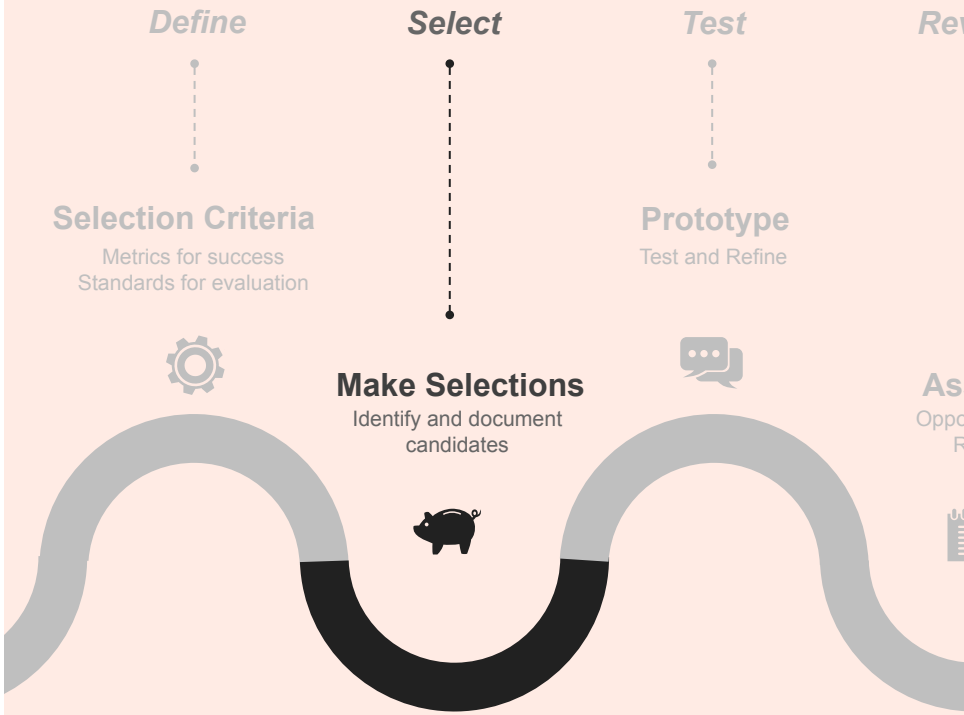


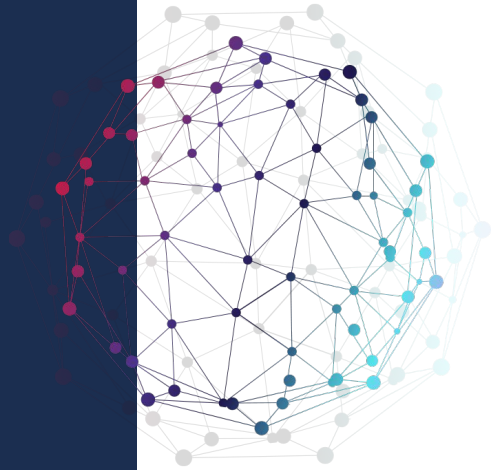
Figure 1: The Transformer - model architecture.

Selection


Define
Select
Test
Selection Criteria
Metrics for success
Standards for evaluation
Prototype
Test and Refine
Make Selections
Identify and document
candidates

Selection

Computational Description as a Machine Learning Problem



The tasks for *Exploring Computational Description* and for *Toward Piloting Computational Description* can be understood as instances of two common problems in natural language processing (NLP):

- **Token classification**. Also known as sequence classification, or sometimes text extraction or entity recognition.
- **Text classification**.

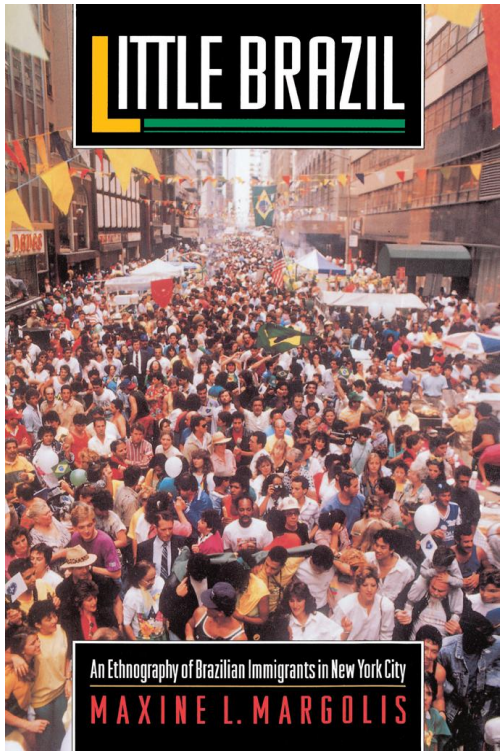
Both of these are instances of *supervised learning* in which existing labeled data is used to train or fine-tune machine learning workflows.

Token Classification

Token classification is the process of identifying groups of tokens—usually words, or parts of words—in a text and assigning them to particular classes or categories.

Or, for a given category or class, returning all of the groups of tokens that fall under that category or class.

For example, we want our machine learning model to be able to identify when a group of words (or tokens) is the name of the author of a work, or a title statement, or the date of publication.



Published by Princeton University Press, 41 William Street,
Princeton, New Jersey 08540
In the United Kingdom: Princeton University Press,
Chichester, West Sussex

Library of Congress Cataloging-in-Publication Data

Margolis, Maxine L., 1942-
Little Brazil : an ethnography of Brazilian immigrants in New York City /
Maxine L. Margolis.
p. cm.
Includes bibliographical references and index.
ISBN 0-691-03348-X (cl.)
ISBN 0-691-00056-5 (pbk.)
eISBN 978-1-40085-175-1 (ebook)
1. Brazilian Americans—New York (N.Y.)—Social life and
customs. 2. New York (N.Y.)—Social life and customs.
I. Title.
F128.9.B68M37 1993
974.7'1004698—dc20 93-13699 CIP

R0

Copyright © 1994 **264$c** by Princeton University Press **264$b**

Published by Princeton University Press **264$b** , 41 William Street,

Princeton, New Jersey 08540

In the United Kingdom: Princeton University Press,

Chichester, West Sussex

All Rights Reserved

Library of Congress Cataloging-in-Publication Data

Margolis, Maxine L., 1942- **100**

Little Brazil : an ethnography of Brazilian immigrants in New York City / Maxine L. Margolis. **245**

Given a group of tokens (or words):

```
0: 'Little', 1: 'Brazil', 2: ':', 3: 'an', 4: 'ethnography', 5: 'of', 6:
'Brazilian', 7: 'immigrants', 8: 'in', 9: 'New', 10: 'York', 11: 'City',
12: '/', 13: 'Maxine', 14: 'L.', 15: 'Margolis.'
```

We want our machine learning model to successfully identify that tokens 0 through 12 correspond to the Title of the work,

Little Brazil : an ethnography of Brazilian immigrants in New York City **Title** / Maxine L. Margolis.

and ideally also that tokens 13 through 15 correspond to the author of the work, and the entire sequence 0 through 15 corresponds to the MARC 245 Title Statement for the work.

Text Classification

Text classification, on the other hand, is about characterizing the sentiment, subject, topic or theme of an entire text.

A book can have a particular subject, or be about a particular theme, or be an instance of a specific genre classification, without any of the words used to describe that subject heading or genre classification appearing anywhere in the book at all.

For example, we want our machine learning model to be able to identify this book as concerning **New York (N.Y.)—Social life and customs** whether or not those exact words appear in the book in that form.

Structured Data

Unlike the previous experiment, we were focused on producing full valid MARC records, with subfield level information.

This imposes an additional requirement on the models, frameworks, libraries or tools that we choose.

They must be able to produce:

- reliable
- structured
- machine readable data

that can be passed to downstream tooling for transformation into MARC

The basic prompting of LLMs in this context consists of creating an LLM prompt that includes the text of the ebook along with some kind of question eliciting bibliographic information about the ebook. To take a simplified example:

```
Base your answer on the following text: <EBOOK_TEXT> . Who is the author of this ebook?
```

In practice this will elicit a response like:

```
The author of the text you provided is Chinua Achebe, whose magnum opus "Things Fall Apart" was published in 1958.
```

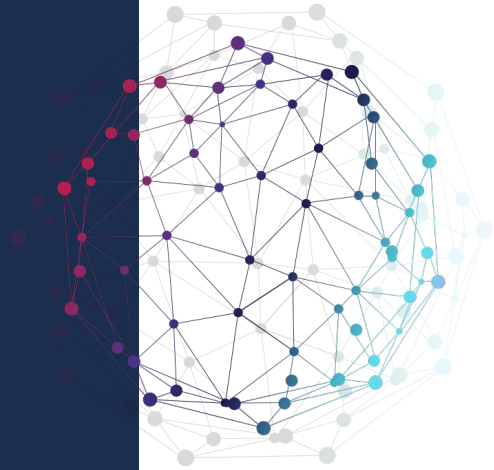
However, if we want to be able to produce a MARC record, we need our tools to be able to produce outputs like:

```
<datafield ind1="1" ind2=" " tag="100">
 <subfield code="a">Achebe, Chinua.</subfield>
</datafield>
<datafield ind1="1" ind2="0" tag="245">
 <subfield code="a">Things fall apart.</subfield>
</datafield>
```

Or a similar equivalent in a different serialization format like JSON.

Selection

Machine learning landscape



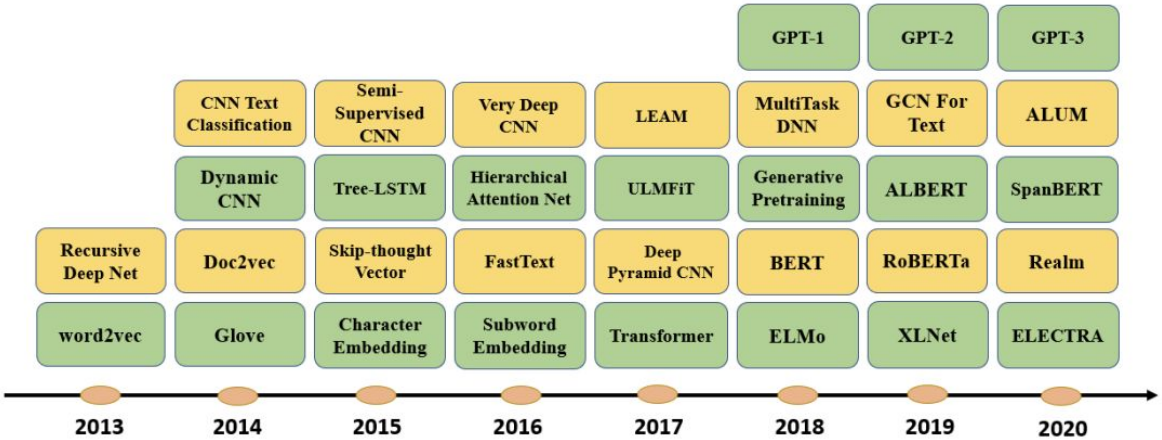


Fig. 22. Some of the most prominent deep learning models for text embedding and classification published from 2013 to 2020.

Figure above from [2004.03705] Deep Learning Based Text Classification: A Comprehensive Review

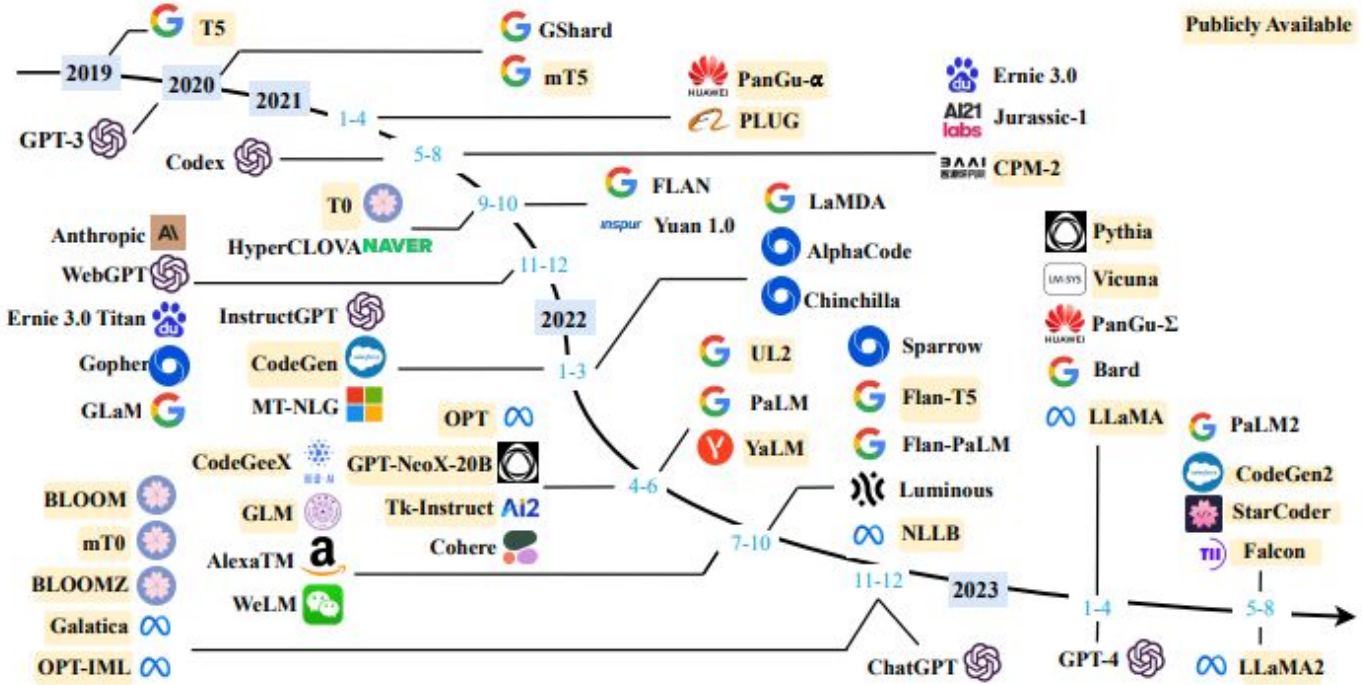


Figure above from [2303.18223] A Survey of Large Language Models

Landscape: 2024

- Large language models are commodity products available as Software as a Service cloud services, or as downloadable open source models that can be deployed on premise or on own cloud infrastructure
- Many new LLMs exist that were not available when the last experiment completed, and many new versions of existing models have been released:
  - GPT-4 / 4o
  - Anthropic Claude
  - Google Gemini
  - Mistral / Mixtral
  - etc

Landscape: 2024

- Frameworks for training and fine-tuning LLMs are more widely available
- Some commercial ML providers offer fine-tuning as a service
- Frameworks for working with multiple ML models such as LangChain are much more widely used
- Frameworks for forcing ML models to output structured data are more widely available:
  - LangChain
  - DSpy
  - Guidance, etc.

What did we select and why?

Model 1

LLM Prompting

Model 1: LLM Prompting

The aim of Model 1 is to evaluate the use of Large Language Models (LLMs) to produce catalog records for ebooks as MARC by testing:

- Basic prompting of LLMs
- In-context learning
- Constraining or structuring outputs to specific schemas or data models

Model 1 was selected to allow us to more thoroughly explore the use of LLMs and generative AI to extract catalog data, which had only been very superficially explored in the previous task order.

Model 1: LLM Prompting

Two different libraries were chosen to manage the creation of prompts, interaction with the LLM and formatting of output, Langchain and Stanford University's DSPy. We chose two different libraries to allow us to compare different approaches to managing:

- Prompting LLM models
- Constraining the outputs of LLM models to schemas, and
- Formatting output data

Both libraries are relatively agnostic about which LLMs are in use, and support a number of different hosting solutions, including self-hosted open-source LLMs, commercial hosting of open-source LLMS (via software as a service cloud solutions), and commercial hosting of closed source LLMs (such as OpenAI's GPT, or Anthropic's Claude).

Model 1: LLM Prompting

In addition, Stanford University's DSPy contains tools to optimize the prompts provided to the language model so that, in theory, the quality of the outputs can improve over time.

Both Langchain and DSpy can be provided with schemas which form part of the process for:

- Prompting the language models to provide output data
- Structuring and constraining the output data to a specific model

Model 1: Schemas

Three different approaches were taken with schemas:

1. ***Basic MARC-based schema***: Schemas that follow the general structure and field/subfield naming of MARC. For this approach, the field/subfield descriptions were taken directly from the MARC 21 documentation.
2. ***LLM description MARC-based schema***: schemas that followed the same structure as 1, but with field/subfield descriptions derived from LLM queries about the MARC 21 definitions.
3. ***PseudoMARC***: A schema that structured the data by field/subfield with human-readable names, with new descriptions created describing the role of the information being sought from the ebook text.

Model 1: Schema Variants

These schema types were in turn provided to the models in multiple variants which tested whether it was better to:

1. Request an entire MARC record with all fields at once, where the concerns were:
    a. Would the data/prompts provides be too large for some models which have a fixed *context window* or input size?
    b. Would asking for more diverse types of data result in lower quality outputs?
2. Request just a single field, with the hypothesis being that:
    a. With a single field, more examples can be provided to the model for *in context learning* or *few-shot optimization*
    b. Potentially there may be less confusion around the desired output
3. Request a smaller set of fields but larger than one, to the test if an intermediate approach, between the two extremes above, might provide better results

Experiment “Runs”

With:

- Two different libraries/frameworks
- Three different approaches to generating schemas
- Three different variant schemas varying by “size” or comprehensiveness

There are potentially a lot of different combinations to test.

The total number of experiment runs for Model 1 comprised around a little over **30** different variants.

Metrics

We gathered two primary metrics in evaluating the outputs for Model 1:

- **Accuracy**: a simple measure of the number of exact matches versus the total number of predictions.
- **Cosine similarity**: because we want to measure how close the model gets to the right answer, as this gives better actionable information than straight match or not match. This works by comparing the predicted output to the label on a character by character basis so we have a better understanding of whether the prediction was slightly wrong, such as producing incorrect punctuation or formatting, as opposed to just standard accuracy. This allowed us to better understand which models were generating better predictions overall and was a recommendation from last year's report..

LLMs used

Although tested with multiple open LLMs, the results in this document are primarily taken from MistralAI models.

This is largely due to their consistency in returning parsable JSON in response to prompts.

Experiments using Meta Llama models either produced no parsable JSON output (Llama-2-13b) or far fewer than MistralAI models (Llama-3-8b).

Model 2

LLM Fine-tuning

Model 2: LLM Fine-tuning

The aim of Model 2 is to evaluate the use of Large Language Models (LLMs) to produce catalog records for ebooks as MARC by testing:

- Fine-tuning of LLM outputs using structured training data

Note, this is different from merely providing examples as part of the prompt for few-shot optimization or in-context learning.

We used MistralAI's API for fine-tuning the Mixtral 7b model.

Note, Mistral's model is an open-source model, so similar fine-tuning could be carried out on hardware controlled by the Library, if required.

Model 2: Fine-tuning schemas

As with model 1, we tested Mixtral 7B fine-tuning with a mixture of the same three core schema approaches:

1. ***Basic MARC-based schema***: Schemas that follow the general structure and field/subfield naming of MARC. For this approach, the field/subfield descriptions were taken directly from the MARC 21 documentation.
2. ***LLM description MARC-based schema***: schemas that followed the same structure as 1, but with field/subfield descriptions derived from LLM queries about the MARC 21 definitions.
3. ***PseudoMARC***: A schema that structured the data by field/subfield with human-readable names, with new descriptions created describing the role of the information being sought from the ebook text.

Model 2: Schema Variants

As with model 1, we had variant data provided to Mixtral for fine-tuning:

1. Request an entire MARC record with all fields at once
2. Request a smaller set of fields but larger than one, to the test if an intermediate approach, between the two extremes above, might provide better results
3. A variant of approach 2, but using a slightly different schema type

We randomly selected 1600 example records and used 1570 for training the model for between 200 and 400 fine-tuning steps, and then used another 30 records for validation.

A separate set of 400 records were used for evaluation of the model.

N.B. The same 400 records were also used to evaluate Model 1 so results can be compared.

Model 3

Vector “Search”

Model 3: Vector “Search”

The aim of Model 3 is to use the outputs from the predicted MARC records to:

- Match field values to authority records
- Identify related e-book or other catalog records for reference and to aid in subject cataloging

The broad approach was to index linked data from id.loc.gov for LCSH and LCNAF authority records, and MARC records taken from the MDS datasets into a vector database to allow semantic search and identification of candidate matches and related records.

Model 3: Vector “Search”

The primary aim for this model is not to generate data for use in the MARC record directly, but instead, the aim is to present this data to expert users in the Human in the Loop (HITL) interface.

The focus was on providing potential authority record matches for creators and subjects that can be used by the reviewers to:

- Review the prediction made by the LLM
- Identify the matching authority record in LCNAF or LCSH and select the correct record

Model 3: Vector “Search”

While the LLM(s) are generally quite good identifying the name of an author or editor, there may not be enough information in the e-book text to definitively associate that name with a specific authority record in LCNAF.

Similarly, an LCSH subject heading predicted by the LLM may be a close match for the correct subject, but not be the exact subject heading selected by an expert cataloger.

Using the predicted values to query against a database that contains authority records allowed us to return a list of potential candidate matches from LCNAF and LCSH that could be reviewed by users.

Vectors are a mathematical representation of an object, a list of numerical values, where each element in the vector represents some feature of the object.

Natural Language Processing and Large Language Models use a particular type of vector, known as an embedding to convert words in natural language into a numerical representation. Importantly, these vector embeddings have the property that the vectors for semantically similar words or phrases are closer together than vectors for semantically distinct words or phrases. This can then be used to find related or similar content.

In the case of this prototype, we converted *both* the predicted values from the experiment models and the reference values from LCNAF and LCSH into vectors which could be used to find matches.

Note, that a vector search of this type is searching for semantically similar, rather than orthographically similar values.

Additionally, vector search of this type allows us to score and rank the predictions to provide expert users with matches from LCNAF or LCSH ordered by how good a match they are for the predicted value.

Model 3: Vector “Search”

One suggestion that had emerged from the initial user research with catalogers was the idea that by returning potentially similar or related works from the catalog, expert users could review how similar works had been cataloged in the past and this would aid in good subject cataloging practice.

So, in addition to LCNAF and LCSH we also indexed all of the 2016 MDSConnect catalog records.

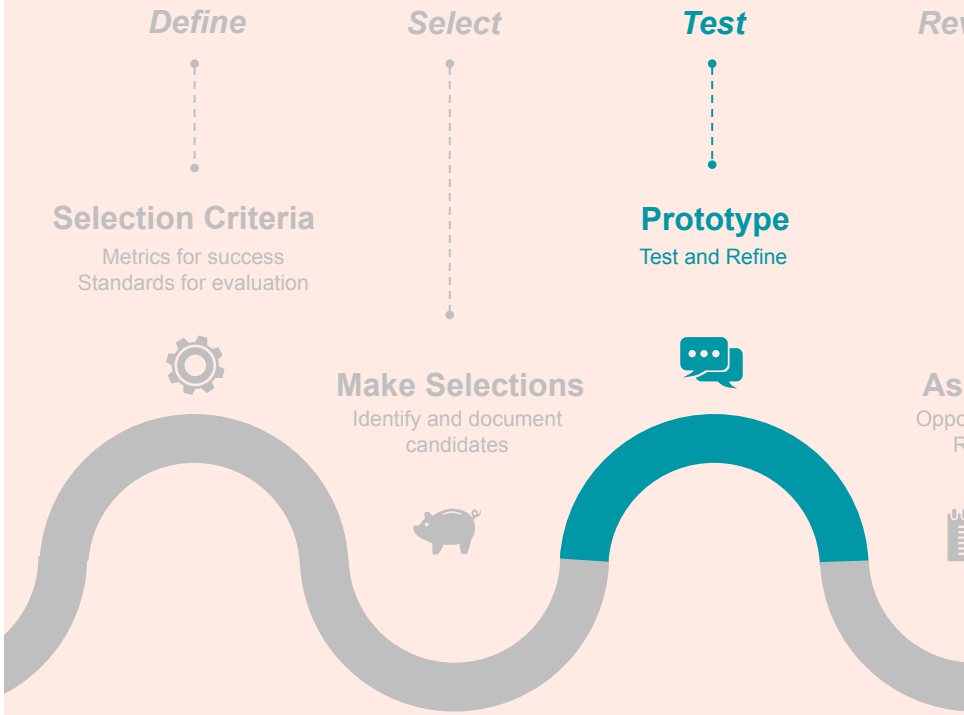
Model 3: Vector “Search”

In each case the approach taken was:

- Load the source dataset into a vector database.
  - Extract payload and source term.
  - Vectorise the source term.
  - Load the vector and payload into the vector database.
- Query a predicted MARC field against the vector database.
  - Extract a search term from MARC subfields.
  - Vectorise the search term.
  - Query the vector database for the most similar entries.
- Format the results for use in the Cataloging UI.

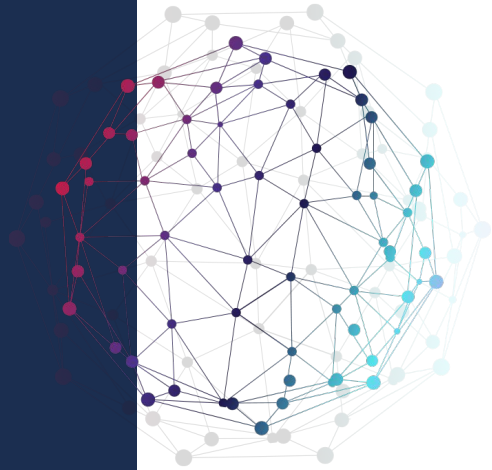
No machine generated metrics were created as the primary purpose of this data was to provide user facing information for the Cataloging UI prototype(s).

Prototyping



Prototypes

Demo



Data for the user facing prototype was composed from:

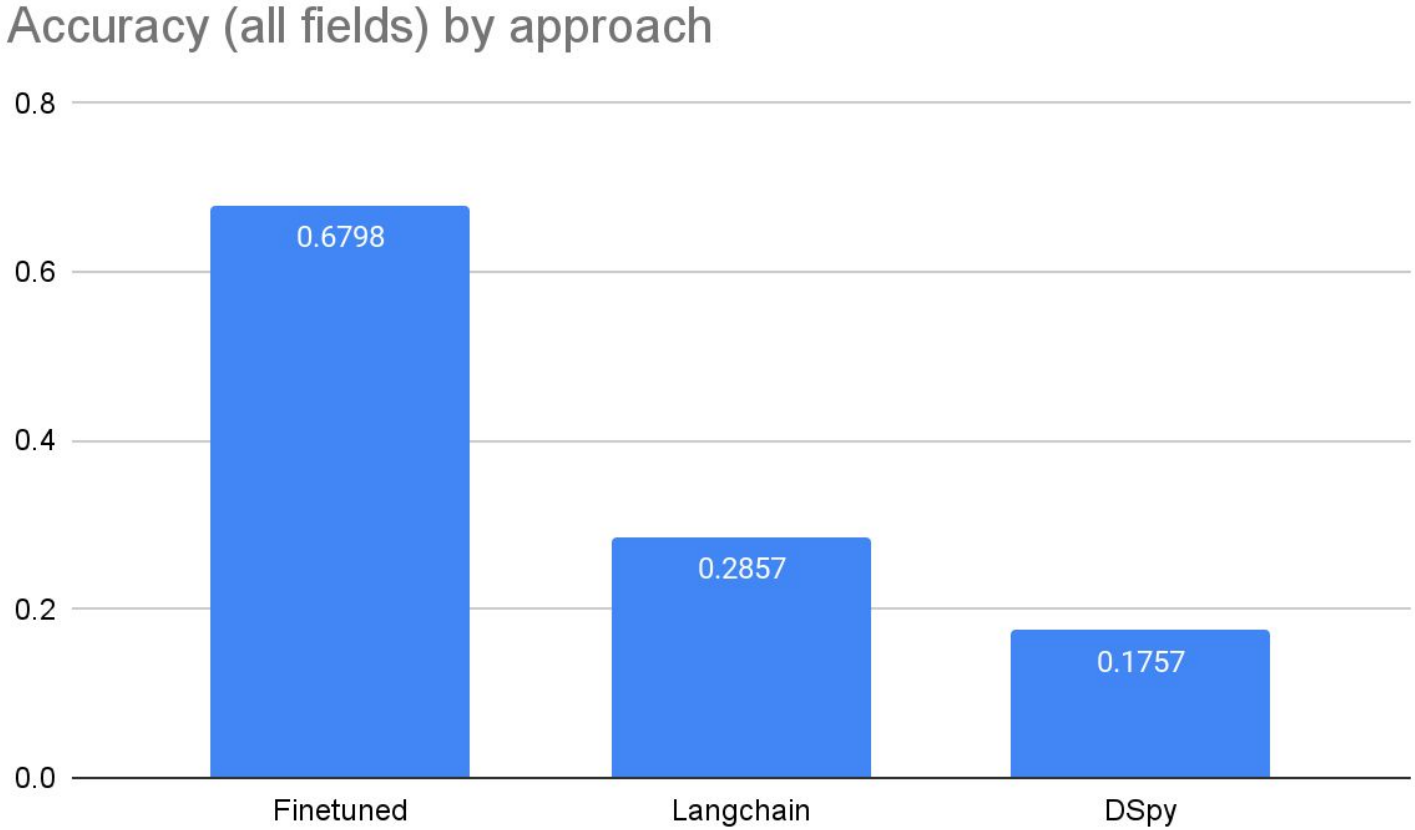
- The best performing (overall) fine-tuned Mistral AI model
- Additional rule-based data for MARC 020, 050 and 082
    - ISBNs, for example, have a regular format and can be found using regular expressions or other methods
- Additional NLP based data for MARC 020, 050, and 082
    - We trained a small NLP model for these fields using Spacy
- Vector search results for LCNAF, LCSH and related catalog records as additional suggestions on 100, 700 and 6xx fields

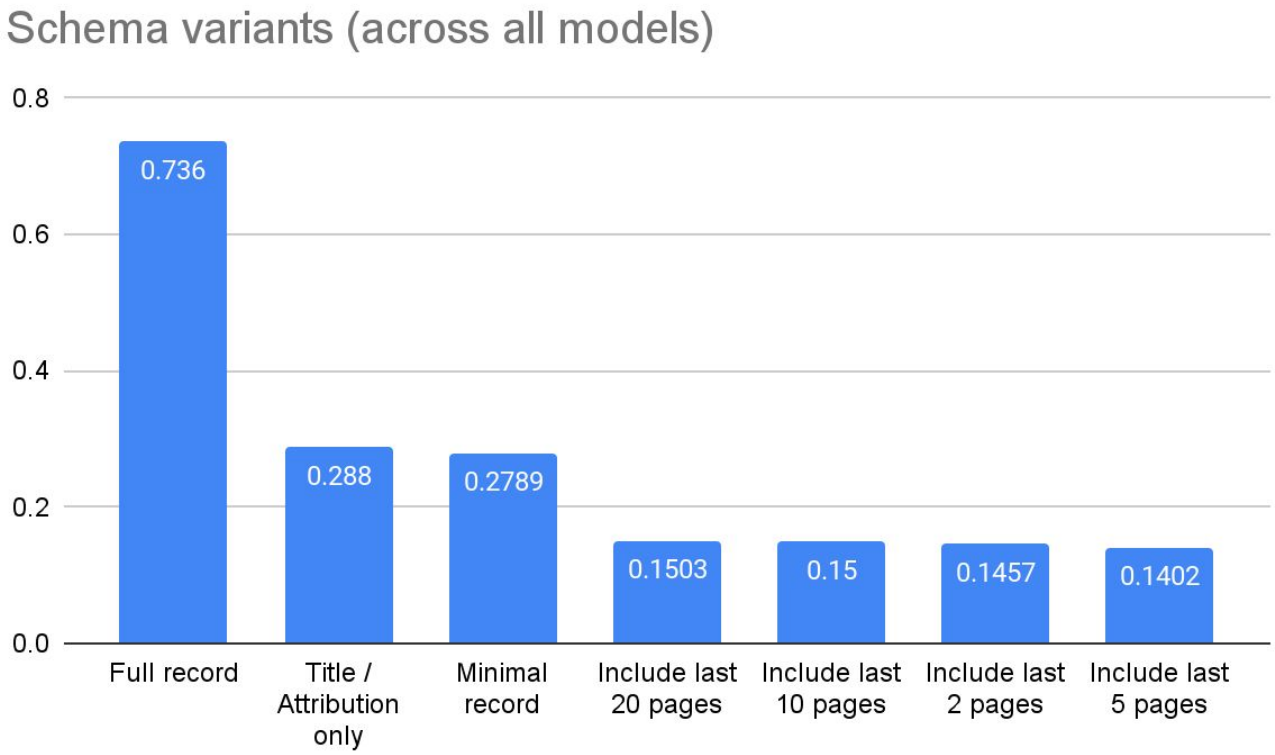
```
=LDR 22 4500
=010 $a2016008076
=020 \\$a9780822360858
=020 \\$a9780822360995
=020 \\$a9780822374466
=050 00$aPN1992.3.E8
=082 00$a791.450947$222
=100 1$aImre, Aniko.
=245 10$aTV socialism /$c Aniko Imre.
=264 \\$aDurham :$bDuke University Press,$c2016.
=490 \\$aConsole-ing passions : television and cultural
power
=650 \\$aTelevision broadcasting$xHistory$y20th
century.$zEurope, Eastern
=650 \\$aSocialism$xHistory$y20th century.$zEurope,
Eastern
=650 \\$aTelevision broadcasting$xSocial aspects$zEurope,
Eastern.
=650 \\$aTelevision programs$zEurope, Eastern.
=650 \\$aTelevision and politics$zEurope, Eastern.
```

bf-a405-04d7a0f10e89

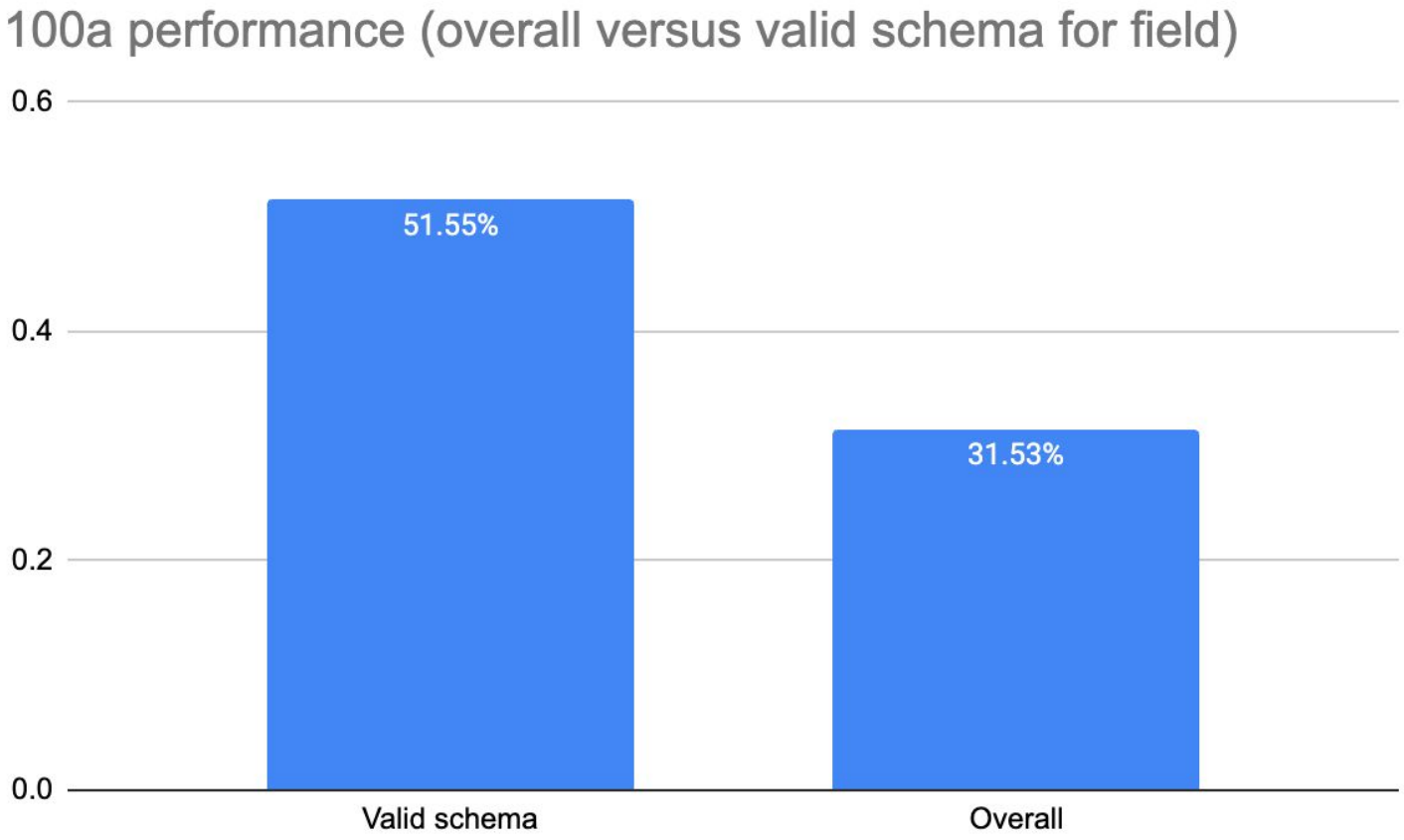
Prototypes

Findings: Performance

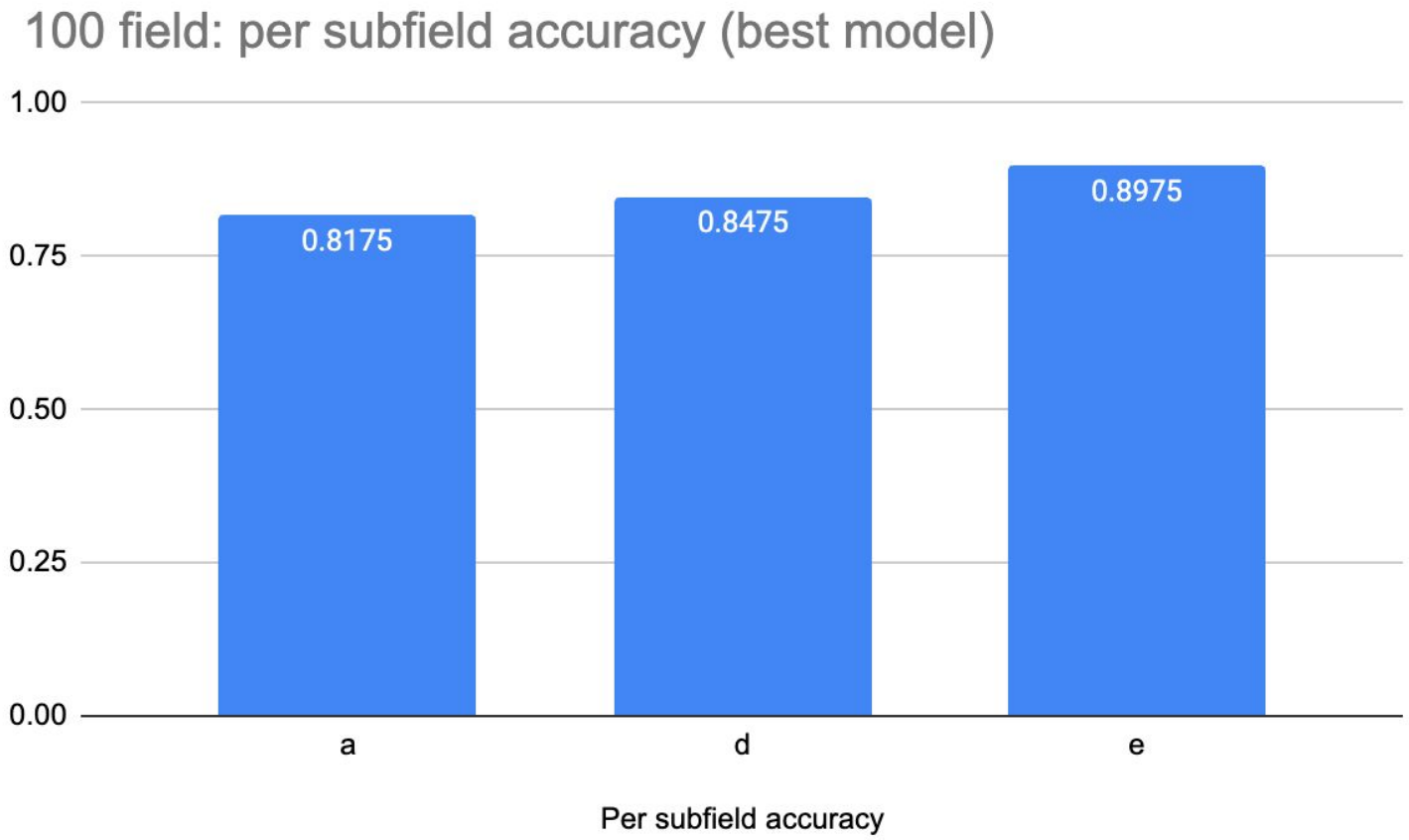

Accuracy (all fields) by approach
0.8
0.6
0.4
0.2
0.0
0.6798
0.2857
0.1757
Finetuned
Langchain
DSpy



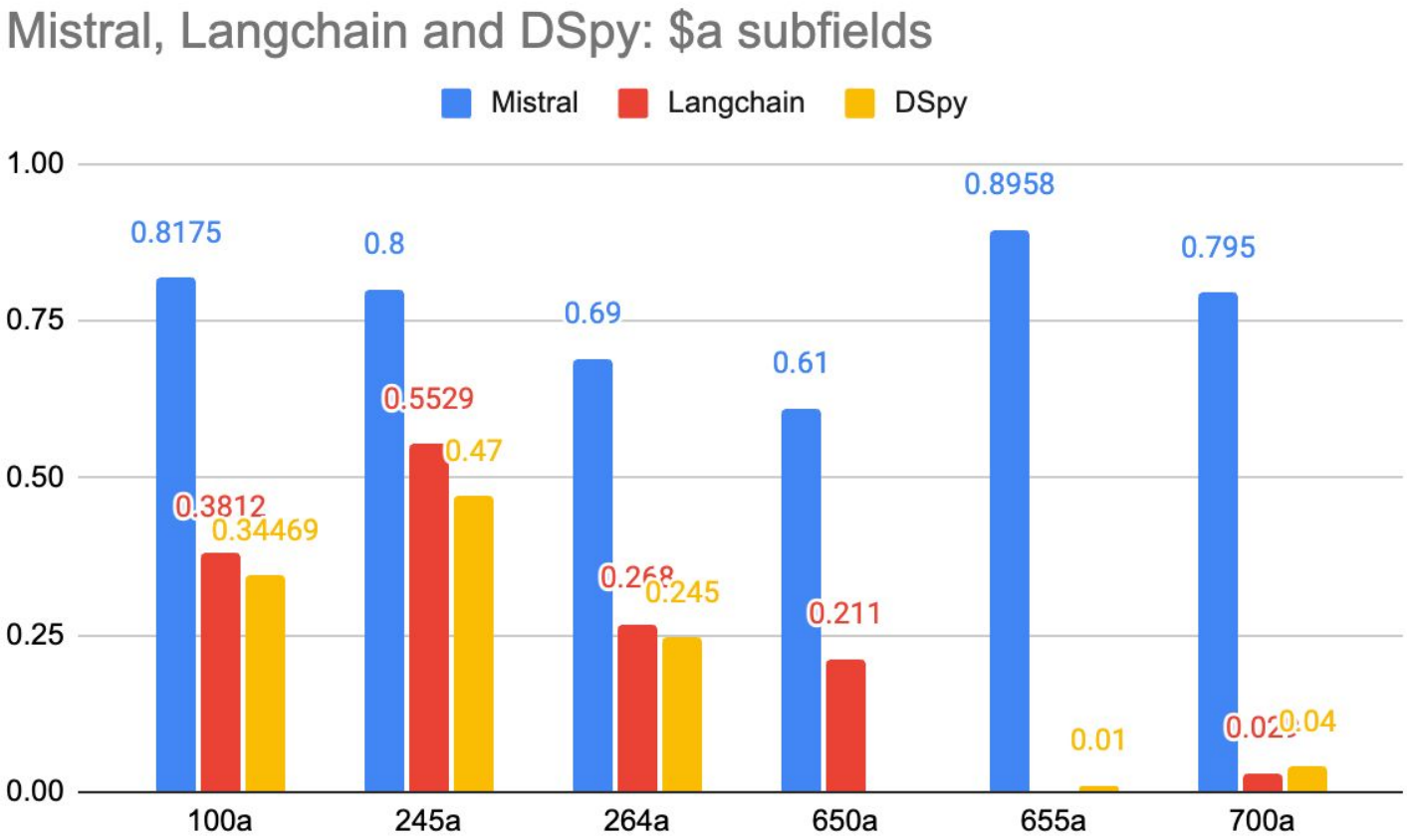
Performance (by schema variant)



Langchain and DSpy: Illustration of accuracy overall versus where the model predicted a valid 100$a field



Granular metrics: measure down to subfield level, per model, per run



$a subfields for core fields, across three approaches

One important caveat versus the metrics presented for last year's experiment.

These numbers are measuring against:

- Full valid MARC records (rather than simple value lists)
- Extraction from an entire ebook (rather than just an annotated ebook fragment)

So the numbers presented are quite representative of expected performance in real world scenarios.

For every field and every subfield, the best performing model was one of the fine tuned MistralAI variants.

In some cases, the best performance for a single field was:

- A schema that included *all* of the MARC fields and subfields

In other cases, it was:

- A schema that only included core fields: 100, 245, 264 and 700

Subject review (manual)

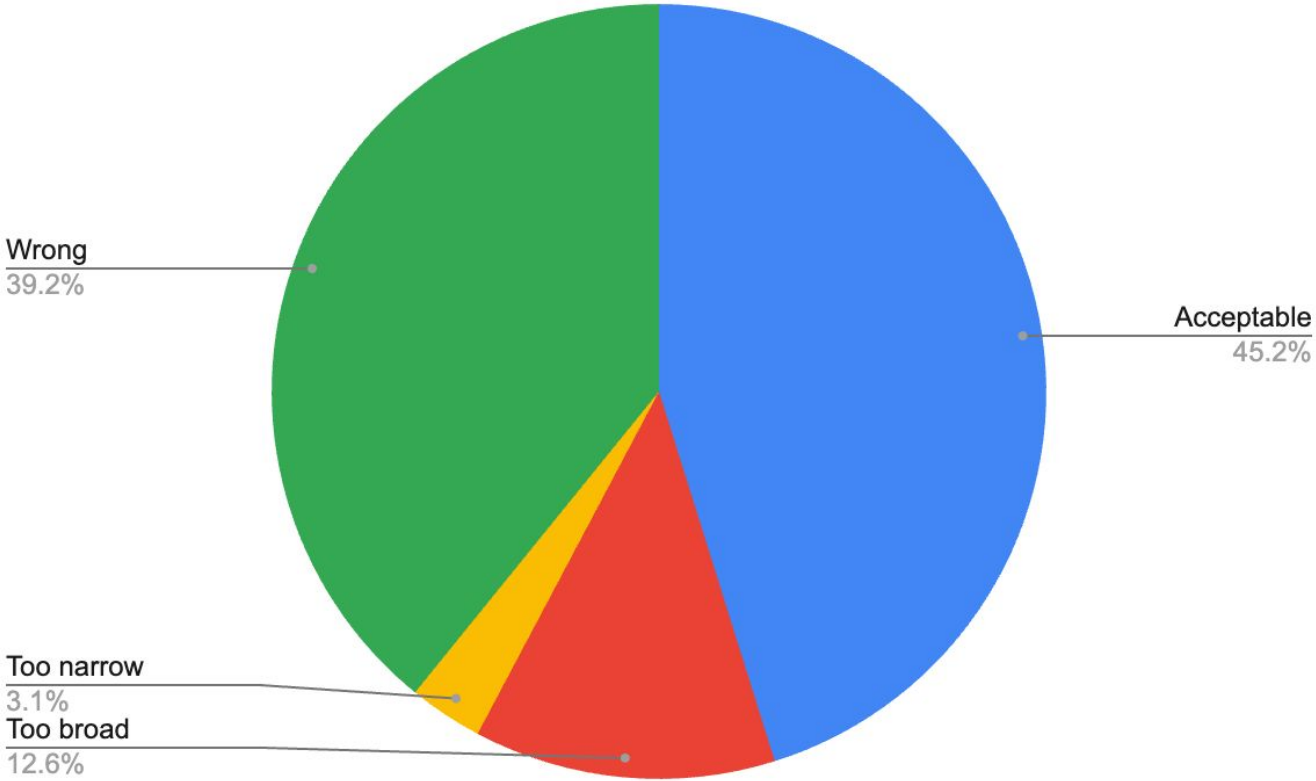
Expert reviewers were asked to review the subject predictions from the LLM model and the additional suggestions provided by the vector search against LCNAF.

The reviewers were asked to indicate whether the suggestion(s) were:

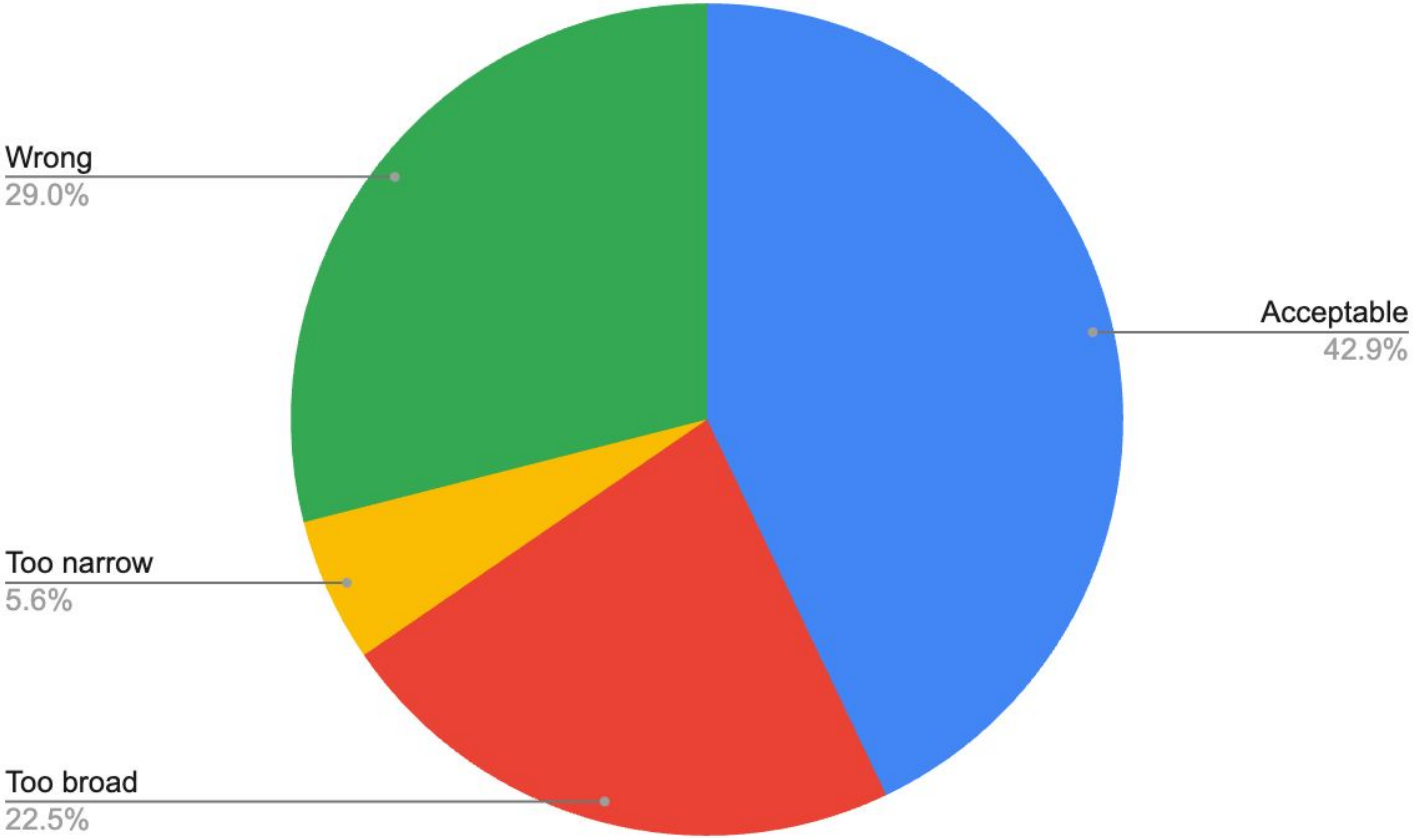
- Acceptable
- Too broad
- Too narrow
- Wrong

And to make comments where applicable.

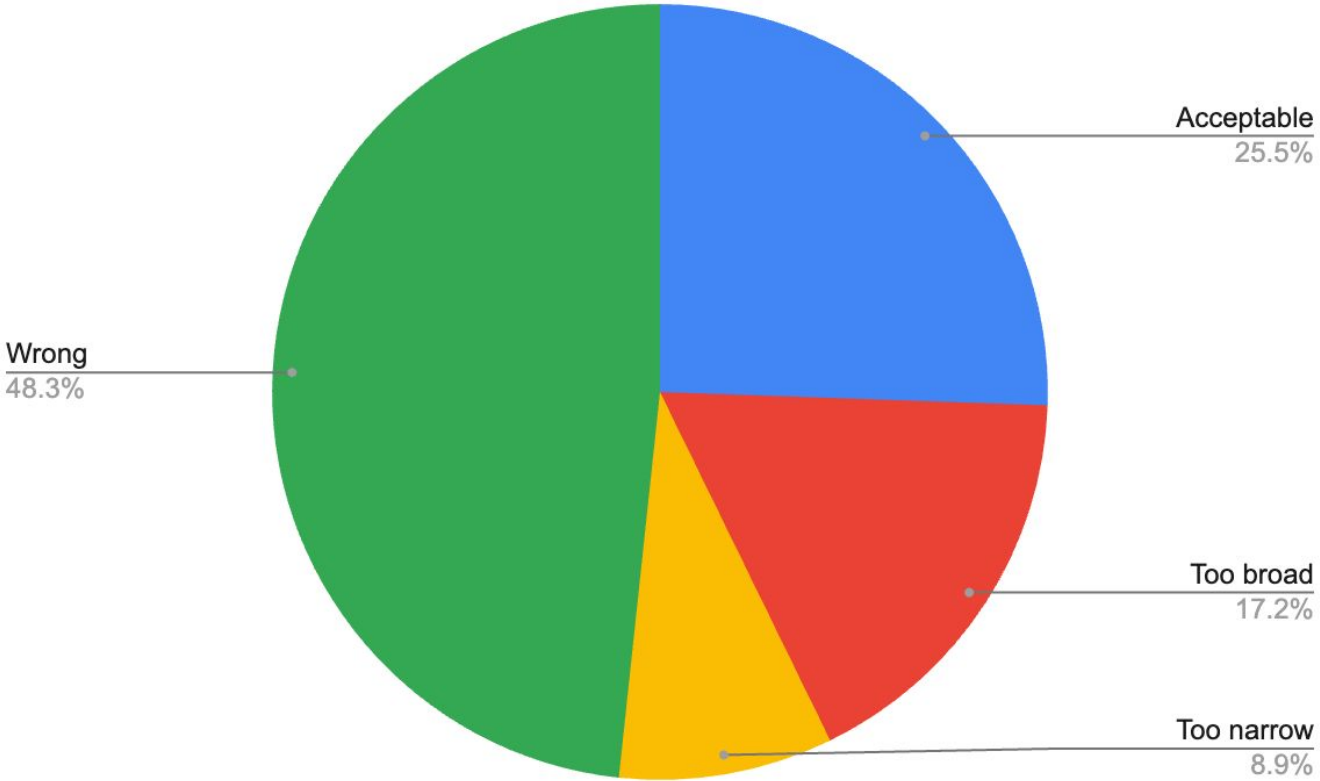
Top ranked suggestion



Original prediction



Top ranked suggestion from the vector search



All suggestions

Subject review (comments)

There were approximately 250 comments on subject predictions assessed to be “wrong” by reviewers. A full analysis to follow, however, regular comments included:

- Wrong subdivision order
  - For the original suggestions from the LLM (not the LCSH matches)
- Subjects being too broad, as, for example, there needed to be a geographic subdivision
- Subjects being too narrow, as, for example, when the geographic subdivision didn’t include all of the places covered by the work
- Incorrect MARC field, e.g. when a term that should be 610 was predicted for 650, etc.
- Subdivisions being provided alone rather than the entire subject

Subject review (manual)

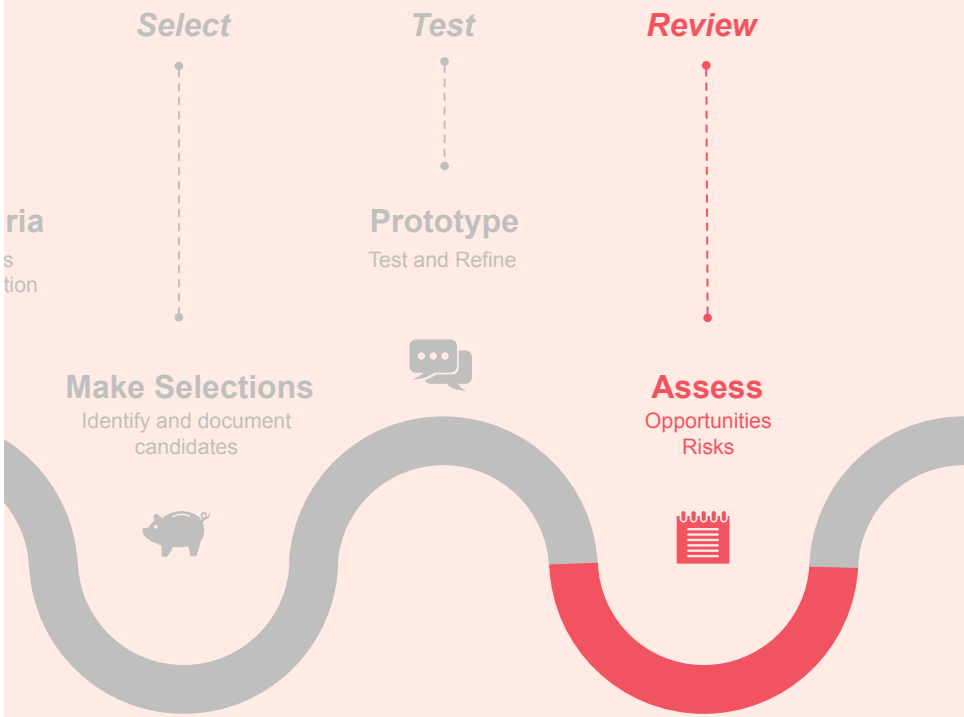
Expert reviewers were asked to review the subject predictions from the LLM model and the additional suggestions provided by the vector search against LCNAF.

The reviewers were asked to indicate whether the suggestion(s) were:

- Acceptable
- Too broad
- Too narrow
- Wrong

And to make comments where applicable.

Review


Select
Test
Review
ria
s
tion
Prototype
Test and Refine
Make Selections
Identify and document candidates
Assess
Opportunities
Risks

Practicality and Performance

- Producing valid MARC records using machine methods is possible
- LLMs can be constrained to produce:
  - Structure data
  - Subfield level data
- Overall accuracy approaches 80+% for most fields and subfields
- Subject fields (6xx) tended to score lower than other descriptive fields
- However, we know from academic research that inter-cataloger agreement for subject cataloging is often quite low (under 50%)
- Fine-tuned LLMs generally perform better than other options
- However, hosting and fine-tuning LLMs “on premise” is more difficult and potentially more costly than using a SaaS commercial service

Next Steps

- Can the same process be repeated with BIBFRAME rather than MARC?
- To what extent are the LLMs relying on:
  - Prior knowledge?
  - CIP cataloging blocks within the e-book text?
- Are commercial providers able to produce better results?
- Do larger models perform better when fine-tuned than smaller models?
- Are there any advantages to using self-hosted LLMs?
- Disadvantages?
- How do different types of material perform? (This may not require new data, but rather re-analysis of the data we already have)

Thank you!