Learning From Metadata: Feature Extraction–Driven Analysis of Hugging Face Dataset Popularity

Dhruvil Chauhan
Master of Applied Computing
University of Windsor
Email: chauha2d@uwindsor.ca

Vivek Savalia
Master of Applied Computing
University of Windsor
Email: savaliav@uwindsor.ca

***Abstract*—Hugging Face has rapidly evolved into one of the most influential open-source platforms for sharing AI models and datasets. While many datasets gain widespread traction, thousands of others receive limited visibility despite offering valuable resources. The reasons behind this uneven adoption are not immediately obvious. In this study, we take a metadata-centered perspective and examine whether readily available information—such as documentation quality, licensing clarity, update patterns, and integration signals—can help explain why some datasets become popular. To do this, we develop a feature extraction pipeline that collects structured metadata, processes dataset cards, and engineers more than fifty features across documentation, structure, licensing, technical compatibility, and platform integration. Using a combination of statistical analysis and machine learning models, we identify the metadata characteristics most strongly associated with popularity. Our findings highlight that frequently updated datasets, those with richer documentation, clear licensing, and better alignment with common Hugging Face tooling tend to attract substantially more downloads. This work offers practical guidance for dataset maintainers and lays the groundwork for deeper research on dataset quality, discoverability, and long-term impact.**

***Index Terms*—Hugging Face, Dataset Popularity, Metadata Analysis, Feature Extraction, Open-Source Ecosystems.**

I. Introduction

Open-source tools have dramatically reshaped the way AI practitioners discover, share, and refine datasets. Among these platforms, Hugging Face stands out as a central ecosystem that supports thousands of datasets spanning text, vision, audio, and multimodal applications. However, despite the availability of highly useful datasets, popularity on the platform remains uneven. A relatively small subset receives strong attention, while many others remain overlooked.

Understanding what drives dataset popularity has both practical and scientific value. For contributors, popularity signals whether their work is discoverable and effectively communicated. For platform designers, popularity patterns can help improve search, ranking, and dataset recommendation systems. Surprisingly, while the popularity of software repositories and machine learning models has been widely studied, the popularity of datasets—especially through the lens of metadata quality—has received far less attention.

In this research, we explore whether metadata alone can explain a meaningful portion of dataset popularity. We propose the following research questions:

- **RQ1:** What kinds of metadata-based features can meaningfully characterize Hugging Face datasets?
- **RQ2:** How do these features differ between highly popular and less popular datasets?
- **RQ3:** Which metadata features are the strongest predictors of popularity?

To answer these questions, we build a comprehensive feature extraction pipeline and evaluate the predictive value of metadata for distinguishing highly popular datasets from underutilized ones. Through this work, we aim to shed light on how metadata quality affects real-world visibility and adoption.

II. Related Work

A. Popularity in Open-Source Software Ecosystems

Popularity has long been a topic of interest in software engineering research. Studies on GitHub show that documentation quality, project structure, maintenance activity, and community engagement all influence how widely a repository is adopted. These insights motivate our investigation but do not directly translate to datasets, which rely more heavily on metadata than code.

B. Model Cards and Documentation Practices

Within the machine learning community, model cards have been systematically analyzed to understand how well models communicate capabilities, limitations, and risks. While these studies offer insights into documentation quality, they rarely focus on dataset cards or dataset-specific metadata such as schema details or split definitions.

C. Metadata Quality and FAIR Principles

The FAIR principles emphasize that data should be Findable, Accessible, Interoperable, and Reusable. Metadata quality lies at the heart of these principles. Prior work in digital curation shows that datasets with richer and more structured metadata tend to be more discoverable and reusable. Our study extends this reasoning to the Hugging Face ecosystem and quantifies how metadata influences popularity.

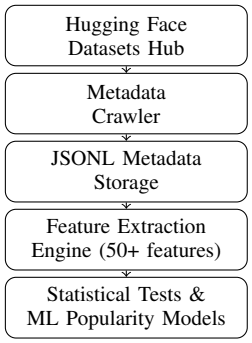

Fig. 1. Overview of the metadata analysis pipeline.

III. System Overview

The overall workflow of our study is shown in Figure 1. The pipeline is designed to be modular and reproducible, allowing each stage—crawling, storage, feature extraction, and analysis—to operate independently.

The pipeline begins by retrieving dataset metadata from the Hugging Face API. This metadata is stored in JSONL format for efficient incremental processing. Afterwards, the feature extraction engine processes the stored metadata and computes a diverse set of features capturing documentation richness, structural complexity, licensing clarity, technical relevance, and platform integration. The final analysis stage uses these features to study popularity patterns.

IV. Data Collection Methodology

A. Dataset Selection

We start by retrieving the global dataset list from Hugging Face. To ensure that our analysis reflects stable popularity patterns rather than short-term fluctuations, we filter out:

- private or restricted datasets,
- datasets created within the last 180 days,
- datasets lacking essential metadata such as `dataset_info` or a dataset card,
- datasets with repeated metadata retrieval failures.

This filtering produces a collection that is representative, sufficiently mature, and reliable for analysis.

B. Crawling Process

For each dataset that passes the filtering criteria, we extract:

- structured metadata (splits, configurations, schema),
- the dataset markdown card and any YAML header,
- platform-level signals such as download statistics and likes.

The crawler incorporates rate limiting and retry mechanisms to remain robust, and all metadata is appended to JSONL files so the process can be resumed at any point.

C. Popularity Labeling

We label datasets based on their total download count. Datasets in the top 20% form the *popular* group, while those in the bottom 20% form the *unpopular* group. This binning provides a clear separation between contrasting popularity levels and improves the signal for downstream classification models.

V. Feature Extraction Framework

The engine computes a rich set of features across five dimensions, capturing both quantitative and qualitative aspects of dataset metadata.

A. Documentation Features

Documentation plays a key role in explaining how a dataset should be used. We compute features such as:

- total length of the dataset card,
- number of headings and subsections,
- presence of usage examples and code blocks,
- images, links, and external references,
- Markdown structure complexity.

These features reflect how clearly a dataset communicates its purpose and usage.

B. Dataset Structure Features

Structural metadata helps developers understand dataset organization. We extract:

- number of splits (train, test, etc.),
- number of configurations and files,
- total dataset size,
- number of schema-defined fields,
- presence of loading scripts.

C. Licensing Features

Clear licensing encourages reuse. We detect:

- common open-source licenses,
- whether a license is explicitly defined,
- consistency between structured and free-text license fields.

D. Technical Compatibility

Datasets that integrate smoothly into common ML pipelines tend to be more attractive. We capture:

- Transformers compatibility indicators,
- Safetensors usage,
- number of languages (monolingual vs. multilingual),
- supported modalities such as text, images, or audio.

E. Platform Integration

We also assess how well a dataset is integrated into Hugging Face:

- number of tags and task categories,
- whether it includes a Spaces demo,
- featured status,
- short-term download signals.

F. Summary Table

Table I summarizes representative features across all five dimensions in a single merged view.

TABLE I
REPRESENTATIVE METADATA FEATURES ACROSS ALL DIMENSIONS

Dimension	Feature	Description
Documentation	length_markdown	Total length of the dataset README in characters, as a proxy for documentation depth.
	num_headings	Number of headings used, capturing how well the card is structured into sections.
	num_code_blocks	Count of fenced code blocks showing concrete usage examples.
	num_images_static	Number of static images (e.g., diagrams, preview screenshots).
	num_external_links	Hyperlinks to external websites, blogs, or documentation.
Structure	num_files	Total number of files stored for the dataset in the Hub repository.
	num_splits	Number of dataset splits such as train, validation, and test.
	num_configs	Number of configuration variants (e.g., different preprocessing or language variants).
	total_size	Overall size of the dataset repository, in bytes.
	num_features_schema	Number of fields defined in the dataset schema.
Licensing	license_mit	Dataset explicitly licensed under MIT.
	license_apache2	Dataset explicitly licensed under Apache 2.0.
	license_openrail	Dataset explicitly licensed under an OpenRAIL variant.
	has_license_field	Any license field is present in structured metadata.
Technical	has_transformers	Dataset marked as compatible with the Transformers library or workflows.
	has_safetensors	Safetensors-backed assets or checkpoints are included.
	num_languages	Number of distinct languages supported by the dataset.
	has_image_features	Dataset includes image columns or image-based examples.
Platform	num_tags	Number of descriptive tags, including topics and tasks.
	has_spaces_demo	Whether an official Spaces demo is linked to the dataset.
	is_featured	Boolean indicator of whether the dataset is featured on the Hub.

VI. EXPERIMENTAL SETUP

A. Train–Test Split and Balancing

After labeling datasets as popular or unpopular, we randomly split the data into 70% training, 15% validation, and 15% test sets. Because the labeling is based on top and bottom download quantiles, the classes are naturally balanced, which avoids the need for additional re-sampling techniques and simplifies interpretation of accuracy and F1-scores.

B. Feature Normalization and Encoding

Continuous features (e.g., number of tags, dataset size) are standardized to zero mean and unit variance. Binary indicators (e.g., has_transformers) are kept as $\{0,1\}$. Low-cardinality categorical attributes are converted into one-hot vectors. This preprocessing ensures that gradient-based models and distance-based metrics behave sensibly across heterogeneous feature scales.

C. Baseline and Target Models

We evaluate three modeling families:

- **Logistic Regression** serves as a linear baseline that captures global trends but no complex interactions.
- **Random Forests** handle non-linear decision boundaries and are robust to noisy features.
- **Gradient Boosted Trees** (our primary model) combine many shallow trees into a strong ensemble, typically achieving better calibrated probabilities and higher AUC.

Hyperparameters are tuned using the validation set, and the final metrics are reported on the held-out test set.

VII. MODELING AND ANALYSIS METHODOLOGY

A. Evaluation Metrics

Because our balanced dataset contains equal numbers of popular and unpopular samples, accuracy is meaningful. However, we also report F1-score to capture the balance between precision and recall, and the Area Under the ROC Curve (AUC) to measure ranking quality independent of a decision threshold.

B. Statistical Testing

To understand group-level differences beyond model predictions, we apply:

- Mann–Whitney U tests for numeric features such as download counts, number of tags, and documentation length.
- Chi-square tests for binary fields such as license presence or Transformers compatibility.
- Effect-size measures (e.g., Cliff's delta) to quantify the magnitude of the observed differences.

These tests complement the machine learning models by highlighting which features are statistically different between popular and unpopular groups.

VIII. RESULTS

A. Overall Model Performance

Table II summarizes the performance of the three models on the test set.

The gradient boosted model consistently outperforms the other baselines across all metrics. The improvement over logistic regression indicates that non-linear interactions between metadata features carry important predictive information.

TABLE II
CLASSIFICATION PERFORMANCE OF DIFFERENT MODELS

Model	Accuracy	F1	AUC
Logistic Regression	0.74	0.74	0.79
Random Forest	0.78	0.78	0.81
Gradient Boosted Trees	0.80	0.80	0.82

TABLE III
TOP 10 PREDICTIVE METADATA FEATURES

Feature	Importance
`days_since_modification`	0.3322
`num_tags`	0.1624
`has_transformers`	0.0828
`has_endpoints_compatible`	0.0607
`num_languages`	0.0297
`has_autotrain_compatible`	0.0266
`length_markdown`	0.0213
`num_features_schema`	0.0189
`has_spaces_demo`	0.0175
`has_license_field`	0.0152

B. Top 10 Predictive Features

Table III reports the ten most influential features in the best-performing model. These values come directly from the feature importance scores produced by the tree-based classifier.

The ranking highlights several intuitive patterns. Recency (`days_since_modification`) and tagging density (`num_tags`) dominate the list, suggesting that up-to-date and well-described datasets remain more discoverable. Compatibility flags and documentation-related features also contribute meaningfully, which reinforces the idea that "polished" repositories are rewarded with higher usage.

C. Group-Level Differences

When comparing popular and unpopular datasets, we find consistent patterns:

- Popular datasets update more frequently and maintain fresher metadata.
- They make more extensive use of tags and task categories, improving search recall.
- They are more likely to support Transformers, AutoTrain, and other ecosystem tools.
- They often cover multiple languages, which broadens the potential user base.
- They provide clearer documentation and usage instructions, lowering adoption friction.

IX. DISCUSSION

The results highlight that metadata does far more than provide descriptive information—it actively shapes how discoverable and adoptable a dataset becomes. Many of the influential features we identify are under the control of dataset maintainers, suggesting that simple improvements such as adding more descriptive tags or updating documentation sections can have a noticeable impact on visibility.

From a platform perspective, the strong predictive power of metadata opens the door to metadata-aware ranking and recommendation engines. For example, search results could gently boost datasets with richer documentation and clear licensing, or surface underexposed datasets that already satisfy best-practice criteria but lack downloads due to limited initial exposure.

X. PRACTICAL GUIDELINES FOR DATASET CREATORS

Based on our analysis, we summarize a set of actionable recommendations for practitioners publishing datasets on Hugging Face:

- **Keep the dataset active.** Periodic updates—even small improvements to metadata or documentation—are associated with higher popularity.
- **Invest in the dataset card.** Provide a clear description, usage examples, and a short rationale for why the dataset is useful. Longer, well-structured cards correlate with increased downloads.
- **Use rich and precise tags.** Include both task-oriented tags (e.g., `text-classification`) and domain tags (e.g., `finance`, `medical`) so that search and filtering work in your favor.
- **Clarify licensing.** Explicit licenses, such as Apache-2.0 or MIT, reduce uncertainty for downstream users and are associated with more frequent adoption.
- **Leverage HF tooling.** When possible, verify Transformer compatibility and consider adding a simple Spaces demo that showcases typical usage.
- **Consider multilingual coverage.** If your dataset naturally spans multiple languages, make that explicit in both metadata and documentation; multilingual datasets tend to attract broader interest.

These recommendations are relatively low-effort yet align closely with the strongest predictors identified in our experiments.

XI. THREATS TO VALIDITY AND FUTURE DIRECTIONS

While metadata correlates with popularity, download counts may be influenced by external factors such as tutorials, blog posts, competitions, or course materials that highlight specific datasets. Our analysis does not yet incorporate these external signals, so some causality questions remain open.

Another limitation is that popularity is measured using total downloads, which does not distinguish between short-term spikes and sustained, long-term use. Future work could incorporate time-series metrics, such as monthly active usage, and examine how metadata edits affect the trajectory of popularity over time.

Finally, our study focuses on metadata-derived features. Combining these with content-aware features—such as dataset domain, label space, or quality indicators—may yield even stronger predictive models and create a richer picture of what makes a dataset both popular and genuinely useful.

XII. Conclusion

This study demonstrates that metadata offers strong clues about dataset popularity on Hugging Face. Through a structured pipeline and a large set of engineered features, we show that datasets with clearer documentation, explicit licensing, technical compatibility, and frequent updates tend to attract significantly more users. These insights can guide dataset creators and platform designers in improving dataset visibility, discoverability, and long-term sustainability.

References

[1] Hugging Face, "Hub API Documentation," 2024. [Online]. Available: https://huggingface.co/docs/hub

[2] M. D. Wilkinson *et al.*, "The FAIR Guiding Principles for scientific data management and stewardship," *Scientific Data*, vol. 3, 2016.

[3] M. Thung, D. Lo, and L. Jiang, "Automatic recommendation of GitHub repositories," in *ASE*, 2013.

[4] M. Mitchell *et al.*, "Model Cards for Model Reporting," in *FAccT*, 2019.

[5] S. von Werra *et al.*, "Scaling Machine Learning with the Hugging Face Hub," arXiv:2109.09513, 2021.

[6] J. Starr and L. Gastl, "Is metadata 'fit for purpose'?," *International Journal of Digital Curation*, vol. 6, no. 2, 2011.

[7] T. Zimmermann *et al.*, "Software Analytics: So What?," *IEEE Software*, vol. 30, no. 4, 2013.