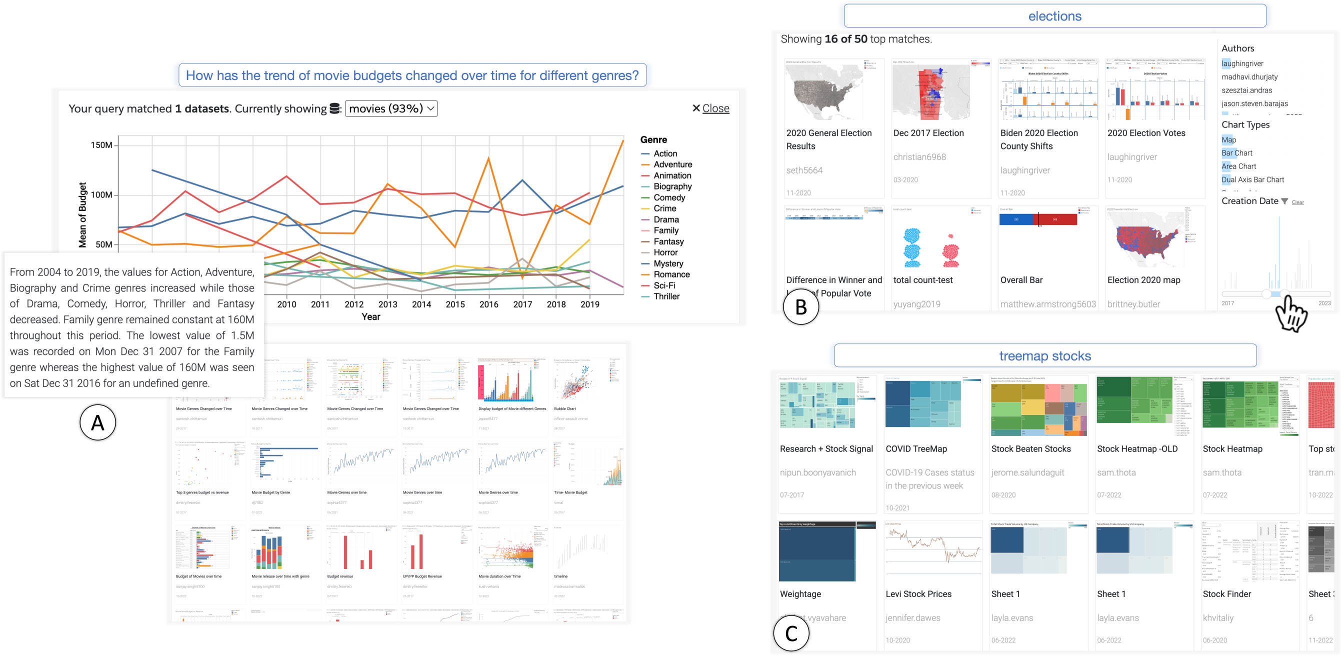
Olio: A Semantic Search Interface for Data Repositories

Vidya Setlur
Tableau Research
Palo Alto, USA
vsetlur@tableau.com

Andriy Kanyuka
Tableau Software
Vancouver, Canada
akanyuka@tableau.com

Arjun Srinivasan
Tableau Research
Seattle, USA
arjunsrinivasan@tableau.com


How has the trend of movie budgets changed over time for different genres?

Your query matched **1 datasets**. Currently showing: movies (93%)

From 2004 to 2019, the values for Action, Adventure, Biography and Crime genres increased while those of Drama, Comedy, Horror, Thriller and Fantasy decreased. Family genre remained constant at 160M throughout this period. The lowest value of 1.5M was recorded on Mon Dec 31 2007 for the Family genre whereas the highest value of 160M was seen on Sat Dec 31 2016 for an undefined genre.

elections

Showing **16 of 50** top matches.

treemap stocks

Figure 1: Examples of various semantic search scenarios supported in Olio. (A) *Q&A*. For an input query, *"How has the trend of movie budgets changed over time for different genres?,"* Olio detects that it is a Q&A search with an analytical intent, 'trend.' A curated data source, 'movies,' is the top-scored match to the query, and the system generates a multivariate line chart response. A generated text summary describes the visualization as shown. Pre-authored visualization content is also displayed as thumbnails below the generated response as additional information. (B) *Exploratory Search*. Olio identifies the input query, *"elections,"* as a keyword search query and shows pre-authored visualizations with text content pertaining to 'elections.' (C) *Design Search*. The query, *"treemap stocks"* is identified as a search of all content containing treemap visualizations pertaining to 'stocks.' Olio returns a set of relevant pre-authored visualizations for the query and displays them as thumbnails. The thumbnails are linked to the actual visualizations if the user desires to continue with their analytical workflow.

ABSTRACT

Search and information retrieval systems are becoming more expressive in interpreting user queries beyond the traditional weighted bag-of-words model of document retrieval. For example, searching for a flight status or a game score returns a dynamically generated response along with supporting, pre-authored documents contextually relevant to the query. In this paper, we extend this hybrid search paradigm to data repositories that contain curated data sources and visualization content. We introduce a semantic search interface, Olio, that provides a hybrid set of results comprising both auto-generated visualization responses and pre-authored charts to blend analytical question-answering with content discovery search goals. We specifically explore three search scenarios - question-and-answering, exploratory search, and design search over data repositories. The interface also provides faceted search support for users to refine and filter the conventional best-first search results based on parameters such as author name, time, and chart type. A preliminary user evaluation of the system demonstrates that Olio's interface and the hybrid search paradigm collectively afford greater expressivity in how users discover insights and visualization content in data repositories.

UIST '23, October 29–November 01, 2023, San Francisco, CA, USA
© 2023 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0132-0/23/10.
https://doi.org/10.1145/3586183.3606806

CCS CONCEPTS

• **Human-centered computing** → **Visualization**; **Natural language interfaces**; • **Information systems** → *Specialized information retrieval.*

KEYWORDS

Hybrid search, question and answering, exploratory search, design search, federated querying, dynamic and static content, visualizations, curated data sources.

ACM Reference Format:
Vidya Setlur, Andriy Kanyuka, and Arjun Srinivasan. 2023. Olio: A Semantic Search Interface for Data Repositories. In *The 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23), October 29–November 01, 2023, San Francisco, CA, USA.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3586183.3606806

1 INTRODUCTION

User expectations of search interfaces are evolving. Search engines are increasingly expected to answer questions along with providing contextually relevant content that help address a searcher's goal [30]. Existing keyword-based search methods are mostly designed for content retrieval. Their main underlying drawback is limited support for structured query types that generally expect focused and specific responses. Natural language (NL) question & answering (Q&A) interfaces, on the other hand, support more fact-finding inquiry but do not support content or document discovery and retrieval. To bridge the gap between these two contrasting search paradigms, a hybrid approach called *semantic search* [46] applies user intent and the meaning (i.e., semantics) of words and phrases to determine the right content that might not be present immediately in the text (the keywords themselves) but is closely tied to what the searcher wants [18]. The information retrieval technique goes beyond simple keyword matching by using information such as entity recognition, word disambiguation, and relationship extraction to interpret the searcher's intent in the queries. For example, keyword search can find documents with the query, "French press," while queries such as "How do I make quickly make strong coffee?" or "manual coffee brewing methods" are better served by semantic search to produce targeted responses.

With an increase in the number of data repositories on the web, including structured data in the form of relational databases, files, and knowledge graphs, there is a plethora of information that supports the blend of generating responses to fact-finding questions with document retrieval [43]. Along similar lines, data repositories and visualization tools such as Observable [77], Tableau Public [100], and Microsoft Power BI Partner Showcase [72] host hundreds or thousands of visualizations representing a wide range of datasets, making them rich platforms for knowledge sharing and consumption. Search plays a pivotal role in these repositories, providing people the ability to winnow in on content they are interested in (e.g., charts on a specific topic, charts showing data trends and bespoke visualizations such as Sankey diagrams, or charts authored by a particular person). Current search systems tend to rely on document-retrieval techniques to provide relevant search results for a given query. However, the challenge with data repositories lies in the sparseness of searchable text within them; data sources and charts often have limited text information in the form of titles, captions, and textual data values, for example. There is a need to explore alternative ways to index and search for content based on this limited availability of textual information.

Another challenge is that current search features for data repositories offer limited expressivity in specifying search queries, restricting users to predominantly perform keyword search for content based on the visualizations' titles and authors. In contrast, other contemporary search interfaces such as general web search, image and video search, and social networking sites enable users to find and discover content through a rich combination of textual content (e.g., keywords or topics covered in a website), visual features within the content (e.g., looking for images with a specific background color), dates (e.g., only viewing videos from the recent week), geographic locations (e.g., limiting search to certain zip codes or cities), and even different types of media (e.g., searching for similar images through features like reverse image search).

Designing expressive search interfaces for data repositories requires gaining a deeper empirical understanding of people's search requirements, given the current limitations of these systems. For instance, what goals do people have in mind when using search in the context of data repositories? How do people formulate their search queries? Is text alone a sufficient modality for search? If not, what are complementary/alternative modalities to consider? What supporting metadata do people want to query for or use to filter the search results?

Contributions. To explore these research questions, we first conducted a set of formative user elicitation interviews with 14 participants who regularly search for visualizations or are involved in the design of search interfaces within mainstream visualization tools and data repositories. Findings from the interviews identified search scenarios specific to content exploration for data repositories and motivated the design and implementation of Olio[1], an interface that supports semantic search behavior by dynamically generating visualization responses and pre-authored visualizations for data repositories. Specifically, the interface implements three search scenarios on a semantic search framework: *Q&A search* by interpreting analytical intent over a set of curated data sources, *exploratory search* using document-based information retrieval methods on existing indexed visualization content, and *design search* by leveraging visualization metadata for the content (Figure 1). The interface also supports facet-driven browsing to prune the search results by author name, time range, and visualization type. Employing Olio as a design probe, we conducted a qualitative study with 11 participants to gain feedback on the implemented metadata and querying features, identify system design and implementation challenges, and better understand user behavior.

The study confirmed that the semantic search paradigm supports the different data repository search goals. We observed that the ability to perform both Q&A and search for pre-authored content facilitated a fluid analytic search experience but raised new questions about user expectations from search systems and the style of user interaction. Lastly, from our observational data and participant feedback, we highlight promising directions for future

[1] The word *Olio* is defined as 'a miscellaneous collection', reflecting the hybrid mix of search content displayed in the interface [69]

work on visualization search interfaces, including better support for creating curated data sources, the need for scaffolding (i.e., support to help with the discoverability of features or functionality in a user interface [88]) and building trust in the system behavior and exploring additional search paradigms and modalities.

2 RELATED WORK

Prior research relating to search systems in the context of visual analysis generally falls into three main categories: (1) semantic web search systems, (2) natural language interfaces (NLIs) for visual analysis, and (3) search interfaces for visualizations.

2.1 Semantic Web Search Systems

Semantic search was initiated as a document search technique to improve searching precision by understanding the purpose of the search (i.e., intent) and the contextual significance of words as they appear in the searchable data space to generate more relevant results [46]. Approaches to semantic web search can roughly be divided into those systems based on structured query languages [29, 39, 51, 58, 80, 101], keyword-based approaches [22, 49, 63, 103, 111], where queries consist of lists of keywords, and natural-language-based approaches [26, 32, 38, 64, 65]. Our work is inspired by the body of semantic search techniques where we explore how we can support various search scenarios (i.e., Q&A, exploratory, and design search) for exploring data repositories.

Early research focused on the problem of augmenting traditional text search with additional metadata using ontological techniques to increase recall and precision [14, 21, 45, 75]. Our work explores semantic augmentation in the context of data repository search by considering additional metadata pertaining to attributes in curated data sources, such as synonyms and related concepts, as well as metadata that describes the pre-authored content, such as the visualization type, data attributes, and author name.

To support targeted Q&A in semantic search, systems have explored ways for accurately detecting NL patterns and phrases that represent temporal intents such as "in the 20th century" or spatial intents such as "in Europe" [62]. Typical approaches in this direction involve a combination of statistical techniques (syntactic parsing) and semantic operations to identify ontology concepts in the user's input. For instance, QUERIX [59] combines the Stanford CoreNLP parser with WordNet to recognize salient phrases from NL user queries [61]. Other Q&A systems apply linguistic processing to the question, identifying named entities and other query-relevant phrases [25, 95, 106]. Olio identifies a set of analytical intents (e.g., trends, location, groupings, aggregations, filters) in the queries for supporting Q&A in a data-oriented semantic search context.

More recently, web search engines blend complementary search experiences of machine-generated results with pre-authored documents and web pages [35]. Search platforms [2, 6] have made updates to their search algorithms that place greater emphasis on search queries, considering overall context and meaning over individual keywords. The algorithm employs form-based or 'template' queries to answer questions at scale in real-time such as the weather, flight status, or the current score of a basketball game. The premise of our research is to explore a similar search paradigm, specifically in the context of data repositories, where we explore the interpretation of queries containing bespoke analytical intents in addition to keyword search.

Traditional information retrieval methods rely on large amounts of searchable text content. However, multimedia repositories that include videos and images, have limited searchable text content. To this end, research in multimedia retrieval has explored metadata extraction techniques to improve the precision and recall of the search algorithms. Techniques include constructing bag-of-word image descriptors from the associated text in documents referring to other similar images [104], analyzing visual features in images [54], parsing XML descriptors in MPEG video files [47], and object and scene retrieval in videos [93], to name a few. Our work addresses an analogous problem when searching data repositories, given the sparseness of searchable text content. We include additional semantics for both data sources and visualizations using ontological enrichment from external corpora, along with properties extracted from the XML properties in the visualizations.

2.2 NLIs for Visual Analysis

NLIs for visual analysis specifically support dynamic Q&A in the larger context of semantic search experience. Systems like DataTone [44] support analytical Q&A, producing a chart according to that inference and then providing ambiguity widgets through which the user could adjust the system's default choice. Eviza [87] and Analyza [34] extend that premise through contextual inferencing. Evizeon [53] and Orko [99] explore the notion of pragmatics in analytical conversation by using the knowledge of data attributes, values, and data-related expressions.

Commercial visualization Q&A systems [7, 9, 72] have evolved over the years to better understand a user's analytical intent expressed in NL and provide reasonable visualization responses. The forms of inferring intent typically rely on explicitly named data attributes, values, and chart types in the user's input queries. Ask Data [89] handles various analytical expressions in NL form, such as grouping of attributes, aggregations, filters, and sorts. The system also handles impreciseness around vague numerical concepts such as 'cheap' and 'high' by inferring a range based on the underlying statistical properties of the data.

However, these systems assume that the data source or dashboard is already preselected before interpreting the queries. Further, they tend to focus on a subset of semantic search (primarily Q&A). Our work explores how analytical search intent can be interpreted to support the various flavors of search across multiple repositories of data sources and visualizations.

2.3 Search interfaces for Visualizations

Large-scale search platforms for visualizations have focused on experiences to help users reason and analyze data sets of interest. ManyEyes, a web-based service, combined public data sharing with interactive visualizations [105]. Users could upload and visualize data on the web, facilitating the sharing and discussion of visualizations. Morton et al. [76] used Tableau Public as a platform to analyze the use of online visual analysis systems and point out that there is a need for improvement of web-based visualization analytics systems to better support both search and content diversity of visualization designs.

Past research also highlights a need for search tools and interfaces to be better integrated into users' authoring workflows. Battle et al. suggest new user experiences, such as design search where the visualization community could easily find D3 content based on chart types, visual style, and structure to help translate their ideas into often complex and bespoke visualizations [17]. Hoque and Agrawala [52] present a search engine for D3 visualizations collected from the web that allows queries based on their visual style and underlying structure. Their search engine indexes the marks and encoding, along with visual style and layout, to support the exploration of D3 charts with specific design characteristics. SightLine is a web portal that passively collects and organizes visualizations to explore the design space of visualizations on the web [86]. By preserving the context of each visualization visit, the tool enables personal provenance through the discovery and exploration of trending visualizations, as well as a more targeted search by querying the metadata collected for each visualization. Along these lines, Observable has the provision for specifying search tags to restrict and combine search terms [78]. The tags stem from metadata properties for these notebooks, including author, title, collection name, etc. [70].

Building on prior research, our work recognizes the various scenarios for search in the context of data repositories and explores a semantic search user experience for supporting these scenarios within a *unified* interface. Olio serves as a research probe to explore the interpretation of search intent against data sources and visualizations by utilizing their underlying metadata.

3 IDENTIFYING SEARCH SCENARIOS FOR DATA REPOSITORIES

To better understand the types of search tasks people would find useful when searching over data repositories, we conducted a series of interviews. We sought to collect a broad perspective from users spanning different backgrounds (e.g., programmers vs. non-programmers) and roles (e.g., visualization designers, consultants, casual viewers, or consumers). We recruited 14 participants (7 females, 7 males), including seven visualization designers or consultants, three product managers involved in the design of visualization repositories, and four software engineers and designers. Participants had working experience with visualization repositories for tools like Tableau (e.g., Tableau Public), Microsoft Power BI (e.g., Power BI Partner Showcase), D3 (e.g., D3's Observable Example Gallery), and general experience searching for visualizations on Google.

Interviews were conducted remotely and lasted 30-45 minutes. We asked participants about their backgrounds (e.g., their job descriptions, visualization repositories they use actively) and then asked them to share their experience, including the scenarios in which they search data or visualization repositories, current limitations, and areas for improvement in terms of the search experience, and metadata they find most relevant during visualization search. We qualitatively analyzed the session transcripts and used an affinity diagramming approach to iteratively group similar comments (e.g., comments referring to searching for visualizations with a specific title or by an author, comments referring to using chart type as part of the search query). We combined these groups under broader clusters of different scenarios search is used in as well as the most relevant search querying features. Below, we summarize the key findings from our formative interviews in terms of the user goals and metadata features most relevant to search in the context of data repositories containing both datasets and pre-authored charts.

3.1 Search Scenarios

We identified three key user goals or scenarios for search in the context of data repositories.

- **Question & Answering (Q&A).** One common goal echoed by participants, particularly those who worked with organization-specific repositories hosting several data sources, was to leverage search to answer analytic questions. This goal is similar to information lookup [50] in the broader web search context where user queries map to brief and discrete pieces of information (e.g., entities, dates, computed values). However, with data repositories, participants wanted to issue analytic questions (e.g., *"What are sales trends across regions?," "highest covid cases by country"*) and get an appropriate response containing visualization and/or text generated from the available data sources.
- **Exploratory Search.** In line with the notion of exploratory search in web search [68], participants wanted to leverage data repositories to learn about a topic through available charts and data. Examples of exploratory search queries include *"NFL drafts," "USA covid trends,"* or *"Fifa world cup."* Such queries are typically open-ended and do not provide refined filtering criteria beyond the topic itself. For instance, one participant (a visualization consultant) referred to exploratory search as one of his prominent goals during the initial stages of customer interactions. He highlighted the example of searching for visualizations on *"private equity dashboards"* on Tableau Public during his recent interaction with a client at an investment firm. Describing her use cases for search, another participant (a visualization designer) alluded to exploratory search as one of her frequent search goals, stating *"I often use search to see a few examples of what people create and to hunt for data sources about a topic."*
- **Design Search.** The ability to find visualizations based on design features (e.g., chart type, color) was another popular use case for search, especially among the seven participants who were designers/consultants or novice visualization authors. Design search query examples include *"sunburst chart," "bar and line combination chart,"* or *"map with icons."* Based on anecdotes shared by the participants, this type of search is typically performed when users are looking for learning resources (e.g., a novice D3 developer looking for examples of force-directed layouts created with D3, a Tableau user trying to create a bespoke visualization like a Sankey diagram) or trying to understand design practices and find inspiration for their own work (e.g., using searches like *"maps with a dark background"* to find examples of charts with specific color constraints).

Note that these scenarios are neither exhaustive nor mutually exclusive. For instance, three participants mentioned "targeted search" as another scenario, where the intent was to retrieve a specific chart or dataset that the users knew existed in the repository. However, we do not explicitly call this scenario out as it would inherently be supported by any search system that supports exploratory search (e.g., users can include specific and precise terms during exploratory search to retrieve the desired content). Furthermore, queries like *"sales by state and segment as a heatmap"* or *"maps showing covid trends,"* combine Q&A and design search, and design and exploratory search, respectively. We also asked participants to rank the scenarios in terms of frequency/importance. The responses, however, were fairly mixed and there was no single primary goal or a specific ordering of scenarios that stood out across participants.

Thus, rather than being a definitive and ordered set, the three scenarios listed above are primarily intended to serve as guidance for broad categories of user tasks to keep in mind when designing search systems for data repositories.

Besides understanding *when* and *why* people use search in data repositories (i.e., the above scenarios), to design and implement an effective search system, we also wanted to identify *what* information people find most relevant while searching and browsing visualizations. To this end, combining the participants' comments and search documentation for platforms like Observable [70] and Tableau Server [28], we curated a list of the most prominent metadata fields that we focus on in our prototype. These fields include the visualization title and description, the chart type (e.g., 'bar chart,' 'map,' 'heatmap'), graphical encodings such as mark type, the visualization author, and the chart's creation date.

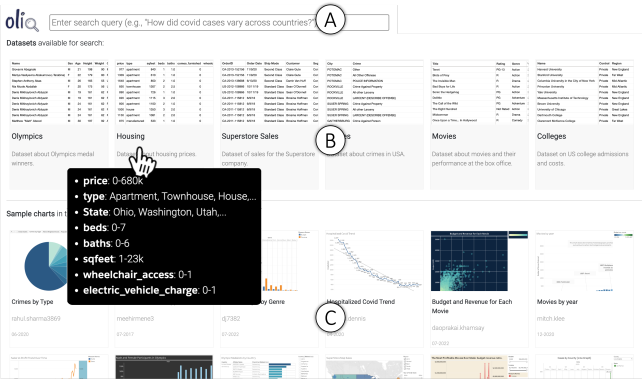
3.2 Design Considerations

Combining the feedback from the formative interviews with guidelines and findings from prior work on visualization search (e.g., [52, 86, 105]), web and image search interfaces (e.g., [50, 68, 92]), and NLIs for visual analysis (e.g., [90, 97, 102]).

DC1. Support a unified experience that supports all three search scenarios. As we discussed the different search scenarios during the formative study, participants noted that they would ideally want the same interface and modality to perform the different tasks. Thus, one consideration for us while building Olio was to design a seamless experience that supported a common input modality (NL) and blended Q&A (a task commonly performed on data source collections) with exploratory and design search (tasks commonly performed with pre-authored visualization repositories).

DC2. Support linguistic variations in queries. Both prior work on NLIs for visualization (e.g., [89, 97, 102]) and web search (e.g., [16, 92]) has shown that people use a variety of phrasings in search queries to accomplish the same goal. Even during our interviews, participants used linguistically varied examples while discussing the same goal (e.g., "*What are sales trends across regions?*" vs. "*sales by region over time*"). Accommodating such user behavior, a second design consideration for Olio was that the system should support a variety of query formats - terse keywords as well as queries phrased as questions or sentence fragments, with an understanding of analytical intent relevant to data repositories in either case.

DC3. Show textual responses and provide guidance for Q&A queries. When discussing Q&A scenarios, we asked participants about the types of visualizations they would expect for different queries. During these conversations, in line with prior research on information lookup on the web [68], participants noted that besides charts, it may be valuable to "*provide a text response to a text query*," suggesting the inclusion of complementary text along with a generated chart. To this end, we noted that given a Q&A query, Olio should not only select an appropriate data source and generate a chart but also text content that leverages the chart to help answer the input query. Furthermore, since Q&A queries can map to multiple data sources and users may not be aware of the available data source and fields, the system should guide users to ask questions (e.g., via query suggestions) and provide metadata information on the relevant data sources (e.g., available data fields and values to query).


oliq
Enter search query (e.g., "How did covid cases vary across countries?")
A
Datasets available for search:
Olympics
Dataset about Olympics medal winners.
Housing
Dataset about housing prices.
Superstore Sales
Dataset of sales for the Superstore company.
B
Dataset about crimes in USA.
Movies
Dataset about movies and their performance at the box office.
Colleges
Dataset on US college admissions and costs.
Sample charts in t
Crimes by Type
rahul.sharma3869
06-2020
meehirmene3
07-2017
by Genre
dj7382
07-2022
Hospitalized Covid Trend
C
dennis
04-2020
Budget and Revenue for Each Movie
daoprakai.khamsay
07-2022
Movies by year
mitch.klee
12-2020

- **price**: 0-680k
- **type**: Apartment, Townhouse, House,...
- **State**: Ohio, Washington, Utah,...
- **beds**: 0-7
- **baths**: 0-6
- **sqfeet**: 1-23k
- **wheelchair_access**: 0-1
- **electric_vehicle_charge**: 0-1

Figure 2: Olio's landing screen. (A) A search input box with a placeholder query suggestion generated based on one of the available data sources. (B) Thumbnail previews of some available data sources. Here, hovering over the 'Housing' data source shows a tooltip displaying metadata about the data source's attributes and values. (C) A sampling of pre-authored visualizations available for search.

DC4. Provide visual summaries and filtering options for search results. One struggle that was echoed by several participants was that current visualization search systems do not provide an easy way to comprehend and sift through results beyond manual inspection. To overcome this limitation with current systems, we noted that the system should provide visual summaries and support dynamic filtering [13] to help people get an overview, organize, and create meaningful facets of the visualization search results.

4 OLIO

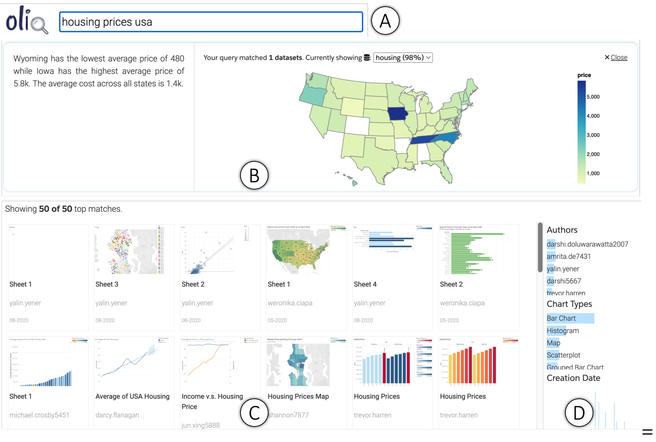
Olio is designed as an interface that supports semantic search behavior by dynamically generating visualization responses and pre-authored visualizations from data repositories. Below, we describe Olio's interface through a brief usage scenario and subsequently detail the key system components and implementation.

4.1 Interface

The interface initially shows a landing screen that displays a sampling of data sources available for Q&A search (Figure 2). A user can hover over a data source thumbnail and view its corresponding metadata information (**DC3**). The user then types a search query, *"housing prices usa"* in the input text box (Figure 3A). The system detects that token 'usa' is a geographic location and searches for a relevant data source in its data repository. Olio finds the housing data source to be a match, and a map is dynamically generated as a Q&A response to the query (Figure 3B). In addition, as part of exploratory search, the query tokens are used as keywords to match any pre-authored visualizations (**DC1**). A grid of thumbnails is displayed to serve as a preview to the user for browsing and exploration (Figure 3C). Each thumbnail is hyperlinked to its corresponding visualization file that the user can choose to peruse in more detail or download to their local machine. The title, author name, and creation date of the visualization are displayed below each thumbnail to provide additional context. Scented widgets [107] appear on the right side of the exploratory search panel to support faceted browsing of the pre-authored visualizations (Figure 3D). The user can narrow down the search results by simultaneously applying one or more filters, namely, author name, visualization type, and the creation date (**DC4**).

4.2 System Overview

Olio is implemented as a web-based application using Python and a Flask backend connected to a Node.js frontend. We leverage Elasticsearch [5], an open-source Java search engine that is designed to be distributive, scalable, and with near real-time query execution performance. As a result, Olio can scale to a large number of data repositories for indexing and search. Figure 4 illustrates a high-level depiction of the system's architecture, with the following main components: query classifier, parser, semantic search framework, Q&A module, and the general search module.



housing prices usa

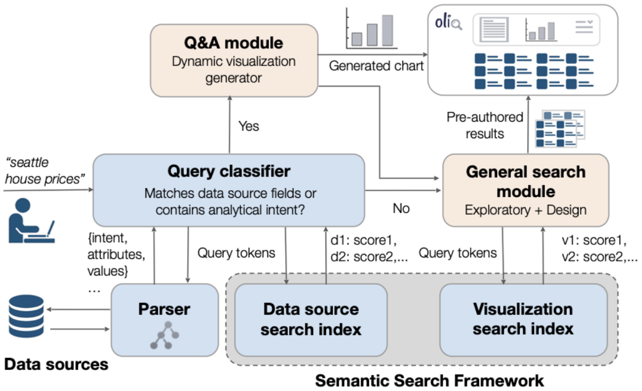
Wyoming has the lowest average price of 480 while Iowa has the highest average price of 5.8k. The average cost across all states is 1.4k.

Your query matched **1 datasets**. Currently showing: housing (98%)

Showing **50 of 50** top matches.

Sheet 1 yalin.yener 08-2020	Sheet 3 yalin.yener 08-2020	Sheet 2 yalin.yener 08-2020	Sheet 1 weronika.ciapa 05-2020	Sheet 4 yalin.yener 08-2020	Sheet 2 weronika.ciapa 05-2020
Sheet 1 michael.crosby5451	Average of USA Housing darcy.flanagan	Income v.s. Housing Price jun.xing5888	Housing Prices Map shannon7877	Housing Prices trevor.harren	Housing Prices trevor.harren

Figure 3: The Olio interface. (A) Search input box. (B) Dynamically generated content, including a chart on the right and text highlighting the key takeaway messages from the chart on the left. Users can hover the mouse cursor over the 🛢 icon to display a dataset summary tooltip similar to Figure 2B. (C) Top 50 pre-authored visualizations that map to the input query. (D) Scented widgets that support dynamic filtering of the pre-authored content results.



Q&A module
Dynamic visualization generator
Generated chart
Yes
Pre-authored results
"seattle house prices"
Query classifier
Matches data source fields or contains analytical intent?
No
General search module
Exploratory + Design
{intent, attributes, values}
...
Query tokens
d1: score1, d2: score2,...
Query tokens
v1: score1, v2: score2,...
Parser
Data source search index
Visualization search index
Data sources
Semantic Search Framework

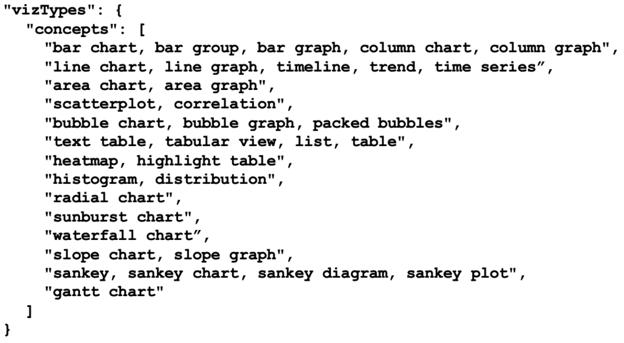
Figure 4: System architecture overview showing the various components: query classifier, parser, semantic search framework, Q&A and general search modules. The query classifier first checks for the presence of tokens that refer to fields from the data sources and analytical intents from the parsed query. If present, dynamically generated visualizations from Q&A search component are rendered, along with general search components (exploratory and design), returning pre-authored visualization results.

A repository of curated data sources is included in the system for Q&A search. The data sources could be any tabular CSV file, but for the purpose of this prototype, we include eight data sources across a variety of familiar topics such as sales [8, 19], sports [1], world events [4], entertainment [15], and civic issues [10, 11]. The datasources varied in the number of attributes as well as their cardinality, including 4-20 columns and ~300-28,000 rows.

4.3 Data Repositories and Metadata

Unlike traditional document search, data sources and visualizations tend to be text-sparse, with limited searchable text content. Hence, Olio augments the data repositories with additional metadata and semantics that helps the system's understanding and interpretation of the search queries. Specifically, attributes and values in the data sources are linked to ontological concepts, including synonyms (e.g., 'film' and 'movie') [82] and related terms (e.g., 'theft,' 'burglary,' and 'crime') [73]. The system includes a small hierarchy of hypernyms and hyponyms, from Wordnet [37], whose depth typically ranges up or down to two hierarchical levels (e.g., [*'beverage,' 'drink'*] → [*'espresso,' 'cappuccino'*]). The metadata also includes data types (i.e., 'text,' 'date,' 'Boolean,' 'geospatial,' 'temporal,' and 'numeric') and attribute semantics, such as currency type (e.g., United States Dollar). This information could also be inferred using existing data pattern matching techniques [12, 23, 83]. The metadata also identifies attributes that are measures (i.e., attributes that can be measured, aggregated, or used for mathematical operations) and dimensions (i.e., fields that cannot be aggregated except as count). This final set of metadata information is then added to the semantic search framework.

The pre-authored content is a set of 75, 000 visualizations sourced from Tableau Public [100], a free community-based platform. The topics of the visualizations are reflective of that demographic of



```
"vizTypes": {
   "concepts": [
      "bar chart, bar group, bar graph, column chart, column graph",
      "line chart, line graph, timeline, trend, time series",
      "area chart, area graph",
      "scatterplot, correlation",
      "bubble chart, bubble graph, packed bubbles",
      "text table, tabular view, list, table",
      "heatmap, highlight table",
      "histogram, distribution",
      "radial chart",
      "sunburst chart",
      "waterfall chart",
      "slope chart, slope graph",
      "sankey, sankey chart, sankey diagram, sankey plot",
      "gantt chart"
   ]
}
```

Figure 5: A JSON list of visualization types and their concepts that are stored as metadata to support design search.

users and include themes such as natural calamities, health, world events, financial news, entertainment, and sports, for example.

Given the XML visual specification of the Tableau workbooks, the system traverses the DOM structure and indexes any text metadata that can be extracted from the visualizations, similar to techniques described in [48]. Extracted metadata includes the visualization title, caption, tags, description, author name, and profile, the visualization marks encoded in the visualization, and the visualization type. To support design search for recognizing visualization types mentioned in the search query (**DC2**), we include a general list of visualization types and their linguistic variants in the semantic search framework, as shown in Figure 5.

While we focused on CSV data sources and Tableau visualizations, the architecture for Olio is extensible to include any new or additional data repositories, including D3 and Vega-lite charts, and knowledgebase articles, for example.
We now describe the rest of Olio's system components in detail.

4.4 Query Classifier

Olio takes as input an NL search query that is passed to the *query classifier*. The classifier supports federated query search [91], which is the process of distributing a query to multiple search repositories and combining results into a single, consolidated search result. Thus, for users, it appears as if they were interacting with a single search instance (**DC1**). In this context, a user can search Olio over heterogeneous data repositories (i.e., both data sources and visualizations) without having to change or modify how they structure the query input. The query classifier passes the search tokens to a *parser* and the *data source search index* (which is part of the semantic search framework) and determines if Olio needs to generate a Q&A search to dynamically generate visualization responses, or simply general search that supports both exploratory and design searches. Algorithm 1 describes the query classification process. At a high level, the query classifier passes the query tokens to the parser (line 7) to determine if the query contains any analytic intents such as aggregation, correlation, temporal, or geospatial expression (refer to Section 4.5 for more details). The query classifier also passes the query tokens to the semantic search framework (refer to Section 4.6 for more details) to determine if the query tokens match fields in any of the data sources (e.g., 'prices' → `Price` in the housing data source) and the normalized match score is greater than a predetermined threshold (line 10). In practice, we found that $fieldMatch = 2$ and $normMatch = .3$ provided a reasonable threshold for relevant data source matches. If both conditions, i.e., the presence of an analytical intent and the match score meets the threshold criteria, then Q&A search is first invoked to dynamically generate visualization responses to the given query (line 13); else, general search is invoked to return pre-authored content from the data repository (line 16).

Algorithm 1 Classifies the search behavior based on whether the query contains an analytical intent and there is a match on one or more of the curated data sources in Olio.

```
1:  function QueryClassifier(query)
        ▷ Boolean to check if there is an analytical intent in query
2:      hasAnalyticalIntent ← False
        ▷ Boolean to check if there is a data source match
3:      hasDSMatch ← False
        ▷ Contains the match scores for query and each data source, ds
4:      dsScores ← getDSScores(query, ds)
        ▷ Contains the normalized match scores for query and each data source, ds
5:      normScores ← norm(dsScores)
        ▷ Predetermined thresholds set for field match in ds and normScores
6:      fieldMatch, normMatch
            ▷ Check if the parsed query contains an analytical intent
7:      if (parseForAnalyticalIntent(query) then
8:          hasAnalyticalIntent ← True
9:      end if
            ▷ Check if the query tokens match fields in ds and normalized match score to ds is greater than a pre-determined thresold
10:     if (dsScores['fields'] > fieldMatch) and (normScores > normMatch)) then
11:         hasDSMatch ← True
12:     end if
            ▷ If query has an analytical intent and contains tokens matching a ds, invoke Q&A search before general search, else just invoke general search.
13:     if (hasAnalyticalIntent and hasDSMatch) then
14:         invokeQ&ASearch(query, ds)
15:     end if
16:     invokeGeneralSearch(query)
17: end function
```

4.5 Parser

The parser removes stopwords (e.g., 'a', 'the') and conjunctions / disjunctions (e.g., 'and,' 'or') from the search query and extracts a list of n-gram tokens (e.g., "*Seattle house prices*" → [Seattle], [house], [prices], [house prices], [Seattle house prices], etc.). The parser employs a Cocke-Kasami-Younger (CKY) parsing algorithm [27, 57, 109] and generates a dependency tree to understand relationships between words in the query.

The input to the underlying CKY parser is a context-free grammar with production rules augmented with both syntactic and

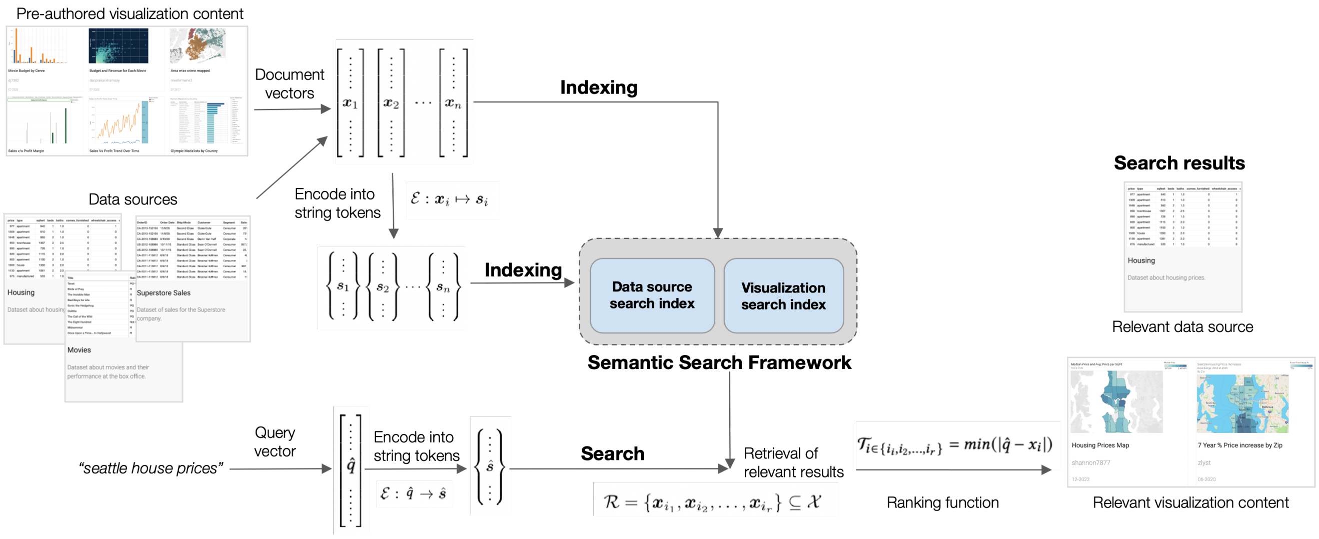

Pre-authored visualization content
Document vectors
x_1 x_2 $\cdots$ x_n
Indexing
Data sources
Housing
Dataset about housing prices.
Superstore Sales
Dataset of sales for the Superstore company.
Movies
Dataset about movies and their performance at the box office.
Encode into string tokens
$\mathcal{E}: x_i \mapsto s_i$
s_1 s_2 $\cdots$ s_n
Indexing
Data source search index
Visualization search index
Semantic Search Framework
Search results
Housing
Dataset about housing prices.
Relevant data source
"seattle house prices"
Query vector
$\hat{q}$
Encode into string tokens
$\mathcal{E}: \hat{q} \to \hat{s}$
$\hat{s}$
Search
Retrieval of relevant results
$\mathcal{R} = \{x_{i_1}, x_{i_2}, \ldots, x_{i_r}\} \subseteq \mathcal{X}$
$\mathcal{T}_{i \in \{i_i, i_2, \ldots, i_r\}} = min(|\hat{q} - x_i|)$
Ranking function
Housing Prices Map
7 Year % Price increase by Zip
Relevant visualization content

Figure 6: Pipeline of the semantic search framework. The document vectors, $\mathcal{X}$ from the pre-authored visualization content and data sources, along with their corresponding encoded string tokens, $\mathcal{S}$, are indexed in the semantic search framework as two data repository indices. At search time, the query vector, $\hat{q}$ from the input search query, *"seattle house prices,"* is encoded into string tokens, and a set of relevant results are returned for both the visualization content and data sources. Using the ranking function, $\mathcal{T}_{i \in \{i_i, i_2, \ldots, i_r\}} = min(|\hat{q} - x_i|)$, the result set is finally ranked to return the top-scoring results.

semantic predicates to detect the following analytical intents in the search query:

- **Grouping**. Partition the data into categories. E.g., 'by' a data attribute.
- **Aggregation**. Group values of multiple rows of data together to form a single value based on a mathematical operation. E.g., 'average,' 'median,' 'count,' 'distinct count.'
- **Correlation**. Statistical measure of the strength of the relationship between two data attributes (measures). E.g., 'correlate,' 'relate.'
- **Filters and limits**. Finite sets of operators that return a subset of the data attribute's domain. E.g., 'filter to,' 'at least,' 'between,' 'at most.' Limits are also a finite set of operators akin to filters that return a subset of the attribute's domain, restricting up to n rows. E.g., 'top,' 'bottom.'
- **Temporal**. Time and date expressions containing temporal tokens and phrases. E.g., 'over time,' 'year,' 'in 2020', 'when.'
- **Geospatial**. Geospatial expressions referring to location and place. E.g., 'in Canada,' 'by location,' 'where.'

To help with detecting data attributes and values along with the intents, the parser has access to the set of curated data sources and their metadata. The parser then compares the n-grams to available data attributes looking for both syntactic (e.g., misspellings) and semantic similarities (e.g., synonyms) using the Levenshtein distance [110] and the Wu-Palmer similarity score [108], respectively (**DC2**). If the parser detects one or more of the aforementioned analytical intents, it returns the intent(s) along with its corresponding data attributes and values to the query classifier.

4.6 Semantic Search Framework

The semantic search framework primarily comprises two phases: indexing and searching content and metadata in the data repositories. This two-phase process applies to content in the data repositories, i.e., both the curated data sources and visualization content. Figure 6 illustrates the pipeline of the semantic search framework.

4.6.1 Indexing. The indexing phase creates indices for each of the data repositories (data sources and visualization content) along with their metadata to support federated search in Olio (**DC1**).

Given a data source and visualization content with associated metadata (i.e., attributes, data values, chart type, author name), each file is represented as a document vector, x_i, where:

$$\mathcal{X} = \{x_1, x_2, ..., x_n\} \tag{1}$$

We also store n-gram string tokens from these document vectors to support partial and exact matches in the system (**DC2**):

$$\mathcal{S} = \{s_1, s_2, ...s_n\} \tag{2}$$

where $s_i = \varepsilon(x_i)$ for some encoder, ε that converts the document vectors into a collection of string tokens of cardinality n. The original vectors $\mathcal{X}$ and encoded tokens $\mathcal{S}$ are stored in the semantic search engine index by specifying the *mapping* of the content, i.e., defining the type and format of the fields in the index. Olio stores the text as keywords in the index, supporting exact-value search, fuzzy matching to handle typos and spelling variations, and n-ngrams for phrasal matching. A scoring algorithm, tokenizers, and filters are specified as part of the search index *settings* to determine how the matched documents are scored with respect to the input query and the handling of tokens, such as the adding of synonyms from a thesaurus, removal of stopwords (e.g., 'a,' 'the,' for') and duplicate tokens, and converting tokens to lowercase. The complete configuration specification is provided in supplementary material.

4.6.2 Search. Conceptually, the search phase has two steps: retrieval and ranking. Given an input query, q, that is represented as a query vector, $\hat{q}$ with query tokens $q_1, q_2, ..., q_j$; we encode the vector into string tokens, $\hat{s} = \varepsilon(\hat{q})$ using the same encoder, ε from the indexing phase. The search process retrieves the most relevant r document vectors, $\mathcal{R} = \{x_1, x_2, ...x_r\}$ as candidates based on the amount of overlap between the query string token set $\hat{s}$ and the document string tokens in $\{s_1, s_2, ..., s_n\}$. More specifically, the scoring function r_{max} maximizes search relevance by computing:

$$\{x_1, x_2, ..., x_r\} = r_{max\, i\in\{1,2,...,n\}} |\hat{s} \cap s_i| \tag{3}$$

Olio then ranks the vectors in the candidate search result set, $\mathcal{R}$ based on $BM25$ scoring [67] with respect to the query vector, $\hat{q}$. BM25 is essentially a bag-of-words retrieval scoring function that ranks documents based on the query terms appearing in each document, regardless of their proximity within the document. It is a preferred metric for computing similarities between vectors as the method corrects for variations in vector magnitudes resulting from uneven-length documents [67]. Given $\hat{q}$, the BM25 score of a document vector, x_i is:

$$BM25(\hat{q}, x_i) = \Sigma_{i=1}^{n} IDF(q_j).\frac{f(q_j, x_i).(k_1 + 1)}{f(q_j, x_i) + k_1.(1 - b + b.\frac{|x_i|}{avgdl})} \tag{4}$$

where $f(q_j, x_i)$ is the number of times that q_j occurs in the document vector, x_i and $avgdl$ is the average document vector length in the search index. k_1 and b are constants to further optimize the scoring function. In practice, we have found that $k_1 \in [1.2, 2.0]$ and $b = 0.75$ tend to provide reasonable ranking behavior. The Inverse Document Frequency, IDF, measures how often a term occurs in all of the documents and ranks unique terms in documents higher. It is computed as:

$$IDF = ln(1 + \frac{(docCnt - f(q_j) + 0.5)}{f(q_j) + 0.5} \tag{5}$$

where $docCnt$ is the total number of documents that have a value for the given query token, q_j and $f(q_j)$ is the number of documents that contains the i^{th} query term.

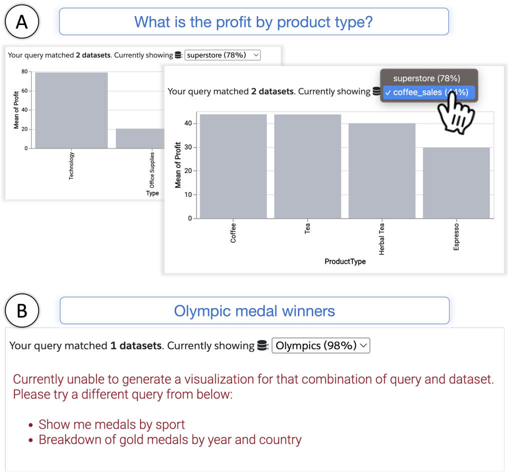
The $BM25$ scoring function sorts the vectors in descending order of normalized $BM25$ scores, $b \in [0, 1]$, i.e., the higher the score, the higher the rank, creating the final ranked search result set, $\mathcal{T}$, ranked based on the minimum difference between the query and each of the document vectors:

$$\mathcal{T}_{i\in\{i_i,i_2,...,i_r\}} = min(|\hat{q} - x_i|) \tag{6}$$

The search request is then passed to the Elasticsearch server to compute Equations 3 and 4 and the system returns a ranked result set of either data sources (used for Q&A) or visualization content used for both exploratory and design search scenarios.

4.7 Q&A Module

The *Q&A module* interprets the analytical intent expressed in the input search queries and dynamically generates visualization responses based on the list of top-matched data source(s) returned from the semantic search framework, as described in Section 4.6. The module accepts tabular CSV datasets for the top-matched data source(s) as input, and all the visualizations in the tool are created using Vega-Lite [85] and D3 [20].



Your query matched **2 datasets**. Currently showing: superstore (78%)

Your query matched **2 datasets**. Currently showing

Your query matched **1 datasets**. Currently showing: Olympics (98%)

Currently unable to generate a visualization for that combination of query and dataset. Please try a different query from below:

- Show me medals by sport
- Breakdown of gold medals by year and country

Figure 7: The Q&A portion of the interface in Olio provides interaction and scaffolding support (DC3). (A) In the case that multiple data sources are identified as top matches to a given search query, a drop-down list shows the ranked list of data sources along with their corresponding percentage match scores. (B) In the case that no valid visualization can be generated for a given search query even though there is a match to a data source, Olio displays a list of query suggestions that the user could choose from to generate a visualization response.

The interface and functionality for Q&A search in Olio is similar to that of NLIs for visual analysis [44, 87, 99] with a few extensions that are inherent to the Q&A behavior in the context of semantic search. For instance, the interface displays text showing a match (if any), to one or more data sources, along with a drop-down menu of the matched data sources (**DC3**). A visualization is rendered based on attributes, values, and the analytical intent in the query, along with a text summary describing the visualization (refer to Figure 1A). A user can peruse the drop-down list of other data source alternatives, along with their corresponding percentage match scores (as computed in Section 4.6.2), and choose to switch to another data source in the drop-down list as shown in Figure 7A. In cases where there is a match to a data source for the query, but the tokens in the query do not resolve to valid attributes and values within the data source, Olio displays suggested queries for the data source (**DC3**), shown in Figure 7B. These query suggestions are generated using a template-based approach presented by Srinivasan and Setlur [98] that is based on a combination of attributes from the data source and data interestingness metrics.

The visualization generation process for Q&A search supports three encoding channels (`x`, `y`, `color`) and four mark types (`bar`, `line`, `point`, and `geoshape`). These marks and encodings support the dynamic generation of bar charts, line charts, scatterplots, and maps that cover the range of analytic intents described in Section 4.5. Olio selects the default visualization using a simplified version of the Show Me system [66], employing similar rules to determine mark types based on the mappings between the visual encodings and attribute data types (e.g., showing a scatterplot if two quantitative attributes are mapped to the xy-channels and showing a line chart if a temporal attribute is visualized on the x-axis with a quantitative attribute on the y-axis).

Finally, Olio displays a dynamic text summary describing the generated visualization (**DC3**). While template-based approaches [36, 60, 74] are viable options for the summary generation process, we chose to employ a large language model (LLM)-based approach [79] to explore its capabilities and better understand its limitations. We initially attempted to pass the chart data as-is to ChatGPT to generate a description. However, we found the model was oftentimes generating wrong statistics or even hallucinating depending on the data domain context. To overcome these challenges but still provide an eloquent description, we instead opted for a combined approach of using both basic statistical computations and an LLM.

Specifically, the input to ChatGPT is a prompt containing a statistical description that is extracted from the generated visualization using a set of heuristics defined in prior data insight recommendation tools [31, 33, 96]. For instance, for bar charts, we identify the min/max and average values; for scatterplots, we compute the Pearson's correlation coefficient [41], and so on. Consider the search query, *"sales by region,"* which results in a bar chart displaying Sales across four Regions. An example of the statistical description, *keyStats* from this bar chart is:

```
Region: Central has a minimum value of $220 for Sales
Region: South has the maximum value of $240 for Sales
Average Sales across Region is: $230
```

The corresponding prompt to ChatGPT then becomes *Rephrase the following input more eloquently:* \n*'${keyStats}*\n', which ultimately generates the text summary: *"The Sales in Central Region had the lowest value of $220, while South Region had the highest value of $240. The average Sales across all Regions was $230.*

4.8 General Search Module

The *general search module* displays thumbnails of pre-authored visualization content along with information such as title and date. The thumbnail images are hyperlinked to the corresponding Tableau Public workbook URLs if users choose to download or analyze the visualization in more detail. The module enables two types of searches: exploratory and design (**DC1**). Exploratory search returns visualization results based on keyword matches (**DC2**) in the input search query (e.g., *"world population"* in Figure 8). Design search is a special form of exploratory search that returns visualization results specifically for keywords containing tokens referring to visualization types, their synonyms, and related concepts (e.g., *"covid correlations"*) (**DC2**). Figure 9 shows examples of design search results in Olio.

5 PRELIMINARY USER STUDY

Using Olio as a design probe, we conducted a preliminary user study to qualitatively assess the overarching idea of combining dynamically generated visualizations with pre-authored charts when searching data repositories. Note that while a comparison of Olio with other systems would be helpful in identifying their relative strengths and weaknesses, the current state-of-the-art semantic search engines [2, 6, 35] focus on web documents rather than data repositories. In addition, existing visualization search systems [52, 86] focus on a subset of search functionality supported in Olio. Removing individual components for an ablation study would be challenging due to Olio's unified hybrid search behavior. However, Olio does implement industry-standard performant recommendations like BM25 scoring and ElasticSearch indexing.

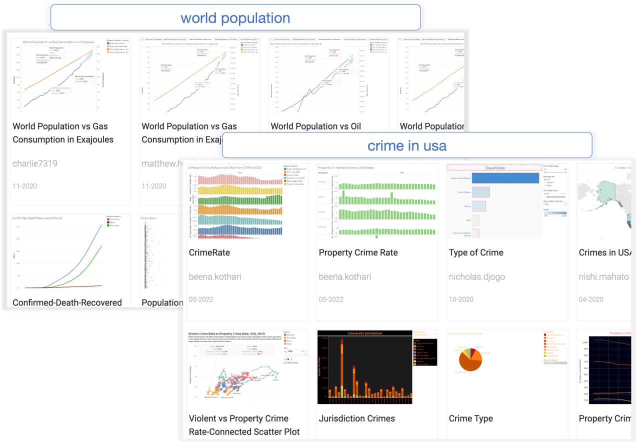


Figure 8: Exploratory search examples. Olio displays thumbnail results of pre-authored visualizations based on keywords found in the input search queries, *"world population"* and *"crime in usa."*

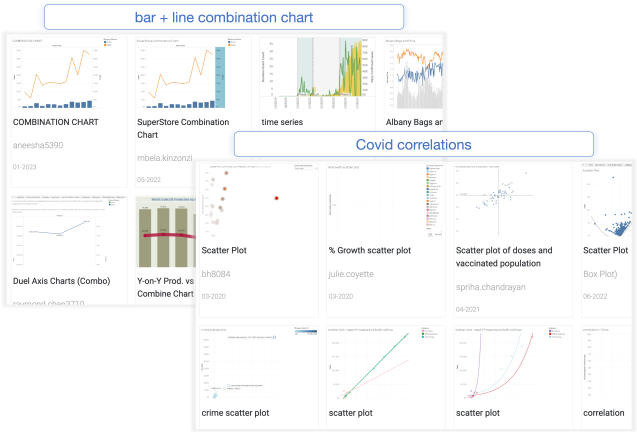


Figure 9: Design search examples. Olio displays thumbnail results of pre-authored visualizations for combinations of chart types (e.g., bar and line charts) as well as for analytical concepts that allude to a specific visualization type (e.g., 'correlation' for scatterplot).

5.1 Participants and Setup

We recruited 11 participants (P1-P11, 6 males and 5 females) through a mailing list at a data analytics software company. Based on self-reporting by the participants, five participants frequently searched for data or visualization content on data repositories, four participants had some experience with searching data repositories but did so infrequently, and two participants had minimal experience with search in the context of data and visualizations.

All sessions were conducted remotely via the Cisco WebEx video conferencing software [3]. The prototype was hosted on a local server running on the experimenter's laptop[2]. Participants were granted control over the experimenter's screen during the session, and all studies followed a think-aloud protocol. The audio, video, and on-screen actions were recorded for all sessions with permission from the participants.

5.2 Procedure

Sessions lasted between 39-60 minutes (mean: 46 min.) and were organized as follows:

Introduction [~10min]: After providing an overview of the study goal, the experimenter asked participants about their job roles and prior experience with search, particularly in the context of data and visualization. The participants were provided a brief introduction to Olio's interface, highlighting the four key components listed in Figure 3. Consistent with Jeopardy-style evaluations of prior NLIs for visualization [44], to avoid biasing participants, we did not provide any explicit training or queries and instead allowed participants to implicitly discover the system through the study tasks.

Task Phase [~25min]: Participants were asked to perform four tasks: one task corresponding to each search goal listed in Section 3 and a fourth open-ended task where participants were allowed to freely explore the available data sources and pre-authored visualizations.

For the *Q&A* task, participants were asked to use one or more of the available data sources for a Jeopardy-style fact [44] about college admissions and a directed analysis question of "listing 1-3 insights on differences between movie genres."

For *exploratory* search, participants were asked to use Olio to explore the topics of elections and colleges in the US. Participants were encouraged to use any search terms and phrase queries however they saw fit.

To assess Olio's support for *design* search, participants were given two images and were asked to search for similar examples using the tool. The images included a treemap showing stock data and a choropleth map of US states with an overlaid pie chart showing product sales data.

Debrief [~10min]: Sessions concluded with a semi-structured interview discussing the overall experience and utility of the underlying idea, support for different search goals, and areas for improvement.

5.3 Results

Overall, participants noted that the semantic search paradigm was useful and could help accomplish their search goals in the context of data repositories. Below, we detail participant feedback and usage behavior with respect to the three search goals listed in Section 3.

Q&A. All participants successfully completed the two Q&A tasks and generally appreciated the system's ability to interpret different phrasing variations (e.g., *"tuition across us regions," "compare movie genres," "What were covid cases across countries?"*́). P9, for instance, said, "*I think the system did better than what I would expect in terms answering questions even though my questions were not good enough to begin with.*" Participants also used the system's ability to dynamically generate visualizations for in-place data exploration. For example, P3 issued a query, *"What are movie budgets by genre?"* that resulted in a bar chart showing average `Budget` by `Genre`. Then, using the metadata tooltip (Figure 2), he inspected other fields to notice the `Gross` field and issued a query to visualize both `Budget` and `Gross`. P6, P7, and P9 also exhibited a similar behavior on multiple instances suggesting that the dynamic content promoted a state of analytic flow. Participants also appreciated the ability to view and choose from matched data sources (Figure 7A). Specifically, participants commented that Olio provided the freedom to "*get more with less*" by supporting keyword-based or open-ended queries to retrieve multiple data sources instead of focusing on well-phrased queries that were optimized to match a single data source. Proposing an improvement to the current interface, however, P10 suggested that instead of rendering a visualization by default, when there are multiple data source matches, the system could allow first choosing a data source and then rendering the chart to save computation resources at scale.

Exploratory search. Participants commented that Olio returned appropriate sets of pre-authored charts during open-ended exploratory searches (e.g., 'elections,' 'olympics winners,' 'covid trends'). However, we noticed that the quality of search results deteriorated when queries went beyond keywords and included additional information such as location (e.g., 'election results in Maryland'), subjective concepts (e.g., 'safest cities in the us'), or metadata properties like 'popularity' that were not included in our chart corpus (e.g., 'popular NBA charts'). Although there were mixed reactions to the quality of search results for exploratory scenarios, all participants appreciated the form and function of the dynamic filtering widgets (Figure 3D), commenting they "*loved it*" (P8, P11) and asked "*why these [dynamic filtering widgets] don't exist in all systems today?*" (P10). Participants predominantly used filters to facet the search results (e.g., choosing chart types to focus on a subset of results or specifying a time range to focus on recent results). On two occasions, participants (P6, P11) also leveraged the filters to chronologically compare search results. When exploring the topic of 'us elections,' for instance, P11 used the date range slider widget to focus on charts created during the 2020 elections to those created using the 2016 elections.

Design search. All participants successfully completed the two design search tasks except for P1 and P5, who found only one of the two required charts. Overall, participants were very positive about the system's support for searching for visualizations by design

[2] 2.4 GHz MacBook Pro running macOS Ventura 13.2.1 set to a resolution of 3072 × 1920.

features, with P8 stating, *"I would love to have this in Tableau Public today."* Similarly, P7 also noted, *"if all we do from this system is enable this search by design [in other visualization repositories], I'd argue that it can solve a lot of challenges for chart authors and especially for someone new to visualization tools."* In terms of user behavior, as we expected, some participants (6 out of 11) did not recollect 'treemap' as a chart type and instead used the data topic, the closest chart type they could think of, or the mark type (e.g., 'stock heatmap,' 'finance group blocks,' 'square chart'). However, since Olio inspects the content of the chart (e.g., titles) as well as design features (e.g., chart type, mark type), the system was able to return relevant charts as some of its top results. In combination with the chart type filter widget, this feature enabled participants to reliably find example charts even when they could not precisely describe them. Participants also successfully used a variety of phrasings to find the combined map and pie chart (e.g., *"examples of piemaps," "pie+map," "show me charts with sales on a map with overlaid pie charts"*).

6 DISCUSSION

Besides user feedback on Olio's support for different search scenarios, the study also helped identify high-level themes and user behaviors pertaining to the semantic search paradigm.

Hybrid results facilitate a fluid and analytical search experience. When talking about the utility of the presented idea, participants particularly appreciated the *complementary nature* of the dynamically generated content and the pre-authored visualizations. Noting the benefits of each component, P9, for instance, said, "*if there's a question that can be answered using a data source then dynamically generated content like this is going to save a lot of time... But when I'm looking for inspiration, of course, that's not the best way, and what I would look for is work by actual people so definitely I see applications for both. None of them are mutually exclusive, and I was able to utilize both of them.*" P2 viewed pre-authored content as a fail-safe for cases when there is no dynamic content stating, "*even if you don't have a dataset that's directly relevant to your query, if there are visualizations, then they come up immediately, which I really appreciate.*" We also observed that the combination of the two content types encouraged participants to introspect on the data and findings more closely. For example, P11 issued a query, *"compare movies by genre"* that generated a bar chart from one of the available data sources, depicting that the `Action` genre has the highest number of movies. However, she found a similar chart in the pre-authored set that showed a different result and correspondingly started inquiring about the data source, what dates it covered, if certain movies were excluded, etc.

The link (or the lack thereof) between the dynamic chart and the pre-authored content should be more apparent. Although participants understood the differences between the two types of results, some participants were initially confused that the dynamic and pre-authored content did not stem from the same data source. P7 alluded to this initial confusion about the visual layout of the page, stating that "*the page kind of creates a hierarchy that is difficult to break. I thought that there was the data I'm looking at at the top was getting visualized in different ways at the bottom, and that was that.*" P2 suggested adding a button above the filters in the interface (Figure 3D) to toggle the pre-authored results to only those that are created using the same data source as the dynamic result. This feedback suggests that for the semantic search experience to be effective, systems like Olio should explore interface designs that clearly depict the relationship between content types, providing users the option to update the content ad-hoc.

The inclusion of dynamically generated content changes user expectations. We noticed an intriguing change in the querying pattern for some participants (P3, P5, P7, P11) as they became familiar with the tool and experienced dynamic content as part of the results. Specifically, once the system generated charts for a few queries, they switched from treating Olio as a search tool using keyword-style queries as input to more of an NLI, issuing imperative system commands like *"Show me a chart of tuition cost by region"* and *"Display examples of treemaps showing stock market data."* While Olio's query parsing logic was able to accommodate most phrasing variations, there were cases where the system no longer met the participants' expectations. For instance, P3 issued a query, *"show examples of charts displaying sales by state"* and Olio returned a map and bar chart for the Superstore data sources as part of its dynamic content along with other pre-authored charts matching the search query. However, P3 was confused by this result as he expected the system to understand the phrase *'show examples'* and ignore the data source search and dynamic chart rendering altogether. When asked about the change in their querying patterns during the session, multiple participants (P3, P11) commented that it was a combination of Olio initially exhibiting an understanding of well-formed natural language utterances and their recent exposure to a slew of conversational interaction experiences through language models like ChatGPT. Such mismatches in the system's functionality (supporting search) and the user's expectation (conversational interaction with an agent) could lead to errors in a larger scale setting, however, and should be clarified through a combination of interface techniques and system guidance.

Textual descriptions should provide structure and contextual information. Participants' reactions to the system-generated descriptions were lukewarm at best, with only four participants (P4, P7, P9, and P10) commenting on them during the study. During their comments, participants noted that the text was helpful in that it re-iterated the key facts from the chart, making it easy to interpret the chart, particularly when it was very dense with overlapping marks (e.g., a multi-series line chart or a scatterplot). However, participants felt that "*text structure is too verbose*" (P7) and "*lacks contextual information about what it means for a value to be high or low,*" (P11) minimizing its overall utility. Such comments suggest that future systems investigating text generation in the context of data repository search should not only focus on the mapping between the generated text and chart, but also on the structure and degree of external information in the text itself.

7 LIMITATIONS AND FUTURE WORK

Semantic search interfaces for data repositories hold promise for helping a user navigate and explore the growing amount of visualizations and analytical assets available. While Olio received positive feedback as a research probe, research exploring semantic search for data repositories is still in its infancy. We identify various themes that highlight the challenges and opportunities for supporting semantic search that are unique to data repositories.

Search precision depends on the availability and curation quality of data sources. Similar to other semantic search experiences [59, 61, 62], Q&A search utilizes a small set of curated datasets to address analytical intents with focused responses. However, an important aspect of search precision, especially for dynamically generated responses, is having access to high-quality, curated data sources with well-understood semantics. However, there is often a disconnect between environments where users publish content and downstream applications like search that consume the content. Participants echoed this challenge with P10 stating, "*this is a great interface and experience but will have to overcome the data garbage problem at scale*." Authors tend to perform some amount of curation during the publishing process but often are not provided sufficient tools to annotate, tag, or enrich their content. The process of curation is often tedious and time-consuming. More research should explore techniques (both semi-automated and automated) [24, 56, 83] to reduce the friction while curating content in data repositories; this includes the de-duplication of similar or near-similar content and the suggestion of topics and tags to help with content discoverability and faceting. Future work should also explore techniques to help with data curation, such as employing LLMs for metadata enrichment, incorporating entity recognition, synonyms, and relational extraction to help automate curation for Q&A support.

Incorporating additional analytical assets and metadata. Olio currently searches over pre-authored singleton visualizations. Future extensions should consider expanding the repertoire of analytical assets to include dashboards, data tables, and computational notebooks [55, 77]. These forms of content have interesting implications for interpreting analytical intent, Q&A, and design search beyond data source and visualization repositories. Further, combining data repositories with document repositories could provide additional searchable metadata to improve search precision and for generating contextually relevant summaries alongside the results.

Need for scaffolding to orient the user. Semantic search interfaces support new techniques for information seeking but with the added complexity of determining the type of queries and understanding the search results. Guidance and scaffolding may need to be provided as users search across multiple data repositories of content. While Olio displays metadata for the available data sources along with query suggestions to guide a user toward a successful search, additional scaffolding could improve sensemaking and exploration. Recent work has explored data-driven autocompletion for helping users formulate targeted Q&A-type queries [88] and integrate contextual query suggestions within a person's sensemaking environment [81]. An interesting research direction would be to explore data scaffolds across different types of search, each unique in its own way, in the context of a semantic search system.

Explore new search paradigms and modalities. Olio indexes available textual content in the data repositories. However, akin to image search, content-based search [94] that leverages *visual* features could improve recall of sparse text content, particularly for design search. Reverse image search [112] addresses the challenge for a user to guess at keywords and terms to return a specific result that they may have in mind. Exploring reverse visualization search, wherein a user provides a sample visualization or sketch to discover content related to the sample visualization image, could support richer forms of expressing design search goals. In addition to new search paradigms, other modalities, and platforms should be explored. Mobile devices, for example, generate large amounts of sensor footprints (e.g., GPS, motion sensors) and user activity data that are often missing from their desktop counterparts [40]. These new sources of implicit and explicit user feedback are valuable for discovering actionable content which is both situationally and contextually relevant to the user. Further, voice and touch modalities could open new possibilities for query formulation and browsing content in the data repositories.

Trust and provenance. Trust is an important issue, and users would benefit from information that communicates the provenance of data sources used to generate the visualization responses, along with the ranking of pre-authored content. Exploring the inclusion of explanations for the search results could lead to increased transparency and understanding of the system behavior [84]. There are additional challenges in an enterprise context; data and visualization content may be private to certain teams and organizations due to the sensitivity of the data (e.g., a human resources department or the current revenue forecast of a business). More work needs to explore ways to support built-in data privacy for indexing and searching of content within these organizational boundaries.

Exploring the utility of LLMs for search. Due to their ease of use and their fluent text-generative capabilities, LLMs are garnering attention for search and conversational interfaces [71]. We explored the use of ChatGPT to generate a summary of the dynamically generated visualization response for Q&A. The model does have limitations in the types of summaries it can generate (as described in Section 6) and challenges around higher-order numeracy reasoning [42]. Custom-trained GPT models could potentially bridge this gap in higher-order analytical reasoning if they can be trained on the data repositories employed in a semantic search system. In addition to summary generation, other utilities for these custom LLMs could explore automatic metadata generation from data repositories to enrich sparse searchable text content. Understanding the quality and accuracy of the generated text both for metadata ingestion and summary generation, and comparing the resulting search experience to that of Olio, are important research directions to pursue as future work.

8 CONCLUSION

In this paper, we explore how we can support data sensemaking and exploration in a semantic search paradigm designed specifically for data repositories. We introduce Olio, a research probe that realizes semantic search behavior through three types of searches: Q&A, exploratory, and design. The system implements a novel semantic search framework that leverages analytical intent derived from the user's query, along with searchable metadata and content to provide a hybrid set of dynamically generated visualization responses with pre-authored visualizations. A preliminary evaluation of Olio indicates that users find the system helpful for supporting a range of both targeted and open-ended data exploration activities. As people continue to actively explore data and author visualizations, there will be an increasing amount of searchable analytical content made available in these data repositories. The ability to support more expressive ways to utilize the content for a wide range of search goals will become especially important. This work provides interesting opportunities for managing and interacting with data beyond search; data curation and enrichment, along with novel modalities for exploring more varieties of content can further scaffold analytical discovery and insights.

REFERENCES

[1] 2018. 120 years of Olympic history: Athletes and results. https://www.kaggle.com/datasets/heesoo37/120-years-of-olympic-history-athletes-and-results. Accessed: 2023.

[2] 2023. Bing Search. https://www.bing.com/.

[3] 2023. Cisco Webex$^{\text{TM}}$. https://www.webex.com.

[4] 2023. Covid-19 Dataset. CC-BY Dataset: https://covid19.ca.gov.

[5] 2023. Elasticsearch. https://www.elastic.co/elasticsearch/.

[6] 2023. Google Search. https://www.google.com/.

[7] 2023. IBM Watson Analytics. http://www.ibm.com/analytics/watson-analytics.

[8] 2023. Tableau Superstore. CC-BY Dataset: https://help.tableau.com/current/guides/get-started-tutorial/en-us/get-started-tutorial-connect.htm.

[9] 2023. ThoughtSpot. http://www.thoughtspot.com.

[10] 2023. U.S. Crimes. CC-BY Dataset: https://www.kaggle.com/datasets/johnybhiduri/us-crime-data.

[11] 2023. U.S. House Listings. CC-BY Dataset: https://www.kaggle.com/datasets/austinreese/usa-housing-listings.

[12] Marco D. Adelfio and Hanan Samet. 2013. Schema Extraction for Tabular Data on the Web. *Proc. VLDB Endow.* 6, 6 (apr 2013), 421–432. https://doi.org/10.14778/2536336.2536343

[13] Christopher Ahlberg and Ben Shneiderman. 1994. Visual information seeking: Tight coupling of dynamic query filters with starfield displays. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 313–317.

[14] Eija Airio, Kalervo Järvelin, Pirkko Saatsi, Jaana Kekäläinen, and Sari Suomela. 2004. *CIRI - An Ontology-based Query Interface for Text Retrieval* (1 ed.). Number 20 in Publications of the Finnish Artificial Intelligence Society. Finnish Artificial Intelligence Society, 73–82.

[15] Shivam Bansal. 2021. Netflix movies and TV shows. https://www.kaggle.com/shivamb/netflix-shows. Accessed: 2023.

[16] Cory Barr, Rosie Jones, and Moira Regelson. 2008. The linguistic structure of English web-search queries. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*. 1021–1030.

[17] Leilani Battle, Danni Feng, and Kelli Webber. 2021. Exploring Visualization Implementation Challenges Faced by D3 Users Online. *arXiv preprint arXiv:2108.02299* (2021).

[18] Ravish Bhagdev, Sam Chapman, Fabio Ciravegna, Vitaveska Lanfranchi, and Daniela Petrelli. 2008. Hybrid Search: Effectively Combining Keywords and Semantic Searches. 554–568. https://doi.org/10.1007/978-3-540-68234-9_41

[19] Arjun Bhaybhang. 2022. Coffee Chains Dataset. https://www.kaggle.com/datasets/arjunbhaybhang/coffee-chains-dataset. Accessed: 2023.

[20] Michael Bostock, Vadim Ogievetsky, and Jeffrey Heer. 2011. D3: Data-Driven Documents. *IEEE Transactions on Visualization and Computer Graphics* (2011). http://idl.cs.washington.edu/papers/d3

[21] D. Buscaldi, Paolo Rosso, and Emilio Sanchis Arnal. 2005. A WordNet-based Query Expansion Method for Geographical Information Retrieval. In *Conference and Labs of the Evaluation Forum*.

[22] Gong Cheng, Weiyi Ge, and Yuzhong Qu. 2008. Falcons: Searching and Browsing Entities on the Semantic Web. In *Proceedings of the 17th International Conference on World Wide Web* (Beijing, China) *(WWW '08)*. Association for Computing Machinery, New York, NY, USA, 1101–1102. https://doi.org/10.1145/1367497.1367676

[23] Christina Christodoulakis, Eric B. Munson, Moshe Gabel, Angela Demke Brown, and Renée J. Miller. 2020. Pytheas: Pattern-Based Table Discovery in CSV Files. *Proc. VLDB Endow.* 13, 12 (jul 2020), 2075–2089. https://doi.org/10.14778/3407790.3407810

[24] Xu Chu, Ihab F. Ilyas, Sanjay Krishnan, and Jiannan Wang. 2016. Data Cleaning: Overview and Emerging Challenges. In *Proceedings of the 2016 International Conference on Management of Data* (San Francisco, California, USA) *(SIGMOD '16)*. Association for Computing Machinery, New York, NY, USA, 2201–2206. https://doi.org/10.1145/2882903.2912574

[25] Jennifer Chu-Carroll, John Prager, Krzysztof Czuba, David Ferrucci, and Pablo Duboue. 2006. Semantic Search via XML Fragments: A High-Precision Approach to IR. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Seattle, Washington, USA) *(SIGIR '06)*. Association for Computing Machinery, New York, NY, USA, 445–452. https://doi.org/10.1145/1148170.1148247

[26] Philipp Cimiano, Peter Haase, Jörg Heizmann, Matthias Mantel, and Rudi Studer. 2008. Towards Portable Natural Language Interfaces to Knowledge Bases - The Case of the ORAKEL System. *Data Knowl. Eng.* 65, 2 (may 2008), 325–354. https://doi.org/10.1016/j.datak.2007.10.007

[27] John Cocke. 1969. *Programming languages and their compilers: Preliminary notes*. New York University.

[28] Joe Constantino. 2023. The Evolution of Tableau Search and Best Practices for Finding Relevant Content. https://www.tableau.com/blog/evolution-tableau-search-and-best-practices-finding-relevant-content.

[29] Olivier Corby, Rose Dieng-Kuntz, and Catherine Faron-Zucker. 2004. Querying the Semantic Web with the CORESE search engine. *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI'2004)*, 705–709.

[30] Paul A. Crook, Alex Marin, Vipul Agarwal, Samantha Anderson, Ohyoung Jang, Aliasgar Lanewala, Karthik Tangirala, and Imed Zitouni. 2018. Conversational Semantic Search: Looking Beyond Web Search,& and Dialog Systems. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining* (Marina Del Rey, CA, USA) *(WSDM '18)*. Association for Computing Machinery, New York, NY, USA, 763–766. https://doi.org/10.1145/3159652.3160590

[31] Zhe Cui, Sriram Karthik Badam, M Adil Yalçin, and Niklas Elmqvist. 2019. Datasite: Proactive visual data exploration with computation of insight-based recommendations. *Information Visualization* 18, 2 (2019), 251–267.

[32] Danica Damljanovic, Milan Agatonovic, and Hamish Cunningham. 2010. Natural Language Interfaces to Ontologies: Combining Syntactic Analysis and Ontology-Based Lookup through the User Interaction. In *Extended Semantic Web Conference*.

[33] Çağatay Demiralp, Peter J Haas, Srinivasan Parthasarathy, and Tejaswini Pedapati. 2017. Foresight: Recommending visual insights. *arXiv preprint arXiv:1707.03877* (2017).

[34] Kedar Dhamdhere, Kevin S. McCurley, Ralfi Nahmias, Mukund Sundararajan, and Qiqi Yan. 2017. Analyza: Exploring Data with Conversation. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces (IUI 2017)*. 493–504.

[35] Li Ding, Tim Finin, Anupam Joshi, Yun Peng, Rong Pan, and Pavan Reddivari. 2005. Search on the Semantic Web. *Computer* 38 (11 2005), 62 – 69. https://doi.org/10.1109/MC.2005.350

[36] Massimo Fasciano and Guy Lapalme. 1996. PostGraphe: A System for the Generation of Statistical Graphics and Text. In *Eighth International Natural Language Generation Workshop*. https://aclanthology.org/W96-0406

[37] Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. Bradford Books.

[38] Miriam Fernandez, Vanessa Lopez, Marta Sabou, Victoria Uren, David Vallet, Enrico Motta, and Pablo Castells. 2008. Semantic Search Meets the Web. In *2008 IEEE International Conference on Semantic Computing*. 253–260. https://doi.org/10.1109/ICSC.2008.52

[39] Tim Finin, Li Ding, Rong Pan, Anupam Joshi, Pranam Kolari, Akshay Java, and Yun Peng. 2005. Swoogle: Searching for Knowledge on the Semantic Web. In *Proceedings of the 20th National Conference on Artificial Intelligence - Volume 4* (Pittsburgh, Pennsylvania) *(AAAI'05)*. AAAI Press, 1682–1683.

[40] Pasi Fränti, Jaakko Sauvola, and Hannu Törmänen. 2005. Mobile information retrieval with local intent analysis. In *Mobile and Ubiquitous Information Access*. Springer, 179–192.

[41] David Freedman, Robert Pisani, and Roger Purves. 2007. Statistics (international student edition). *Pisani, R. Purves, 4th edn. WW Norton & Company, New York* (2007).

[42] Simon Frieder, Luca Pinchetti, Ryan-Rhys Griffiths, Tommaso Salvatori, Thomas Lukasiewicz, Philipp Christian Petersen, Alexis Chevalier, and Julius Berner. 2023. Mathematical Capabilities of ChatGPT. arXiv:2301.13867 [cs.LG]

[43] Sainyam Galhotra and Udayan Khurana. 2020. Semantic Search over Structured Data. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management* (Virtual Event, Ireland) *(CIKM '20)*. Association for Computing Machinery, New York, NY, USA, 3381–3384. https://doi.org/10.1145/3340531.3417426

[44] Tong Gao, Mira Dontcheva, Eytan Adar, Zhicheng Liu, and Karrie G. Karahalios. 2015. DataTone: Managing Ambiguity in Natural Language Interfaces for Data Visualization. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software Technology (UIST 2015)*. ACM, New York, NY, USA, 489–500.

[45] Thomas R. Gruber. 1993. A translation approach to portable ontology specifications. *Knowledge Acquisition* 5, 2 (1993), 199–220. https://doi.org/10.1006/knac.1993.1008

[46] Ramanathan V. Guha, Rob McCool, and Eric Miller. 2003. Semantic search. In *The Web Conference*.

[47] Samira Hammiche, Salima Benbernou, Mohand-Saïd Hacid, and Athena Vakali. 2004. Semantic Retrieval of Multimedia Data. In *Proceedings of the 2nd ACM International Workshop on Multimedia Databases* (Washington, DC, USA) *(MMDB*

'04). Association for Computing Machinery, New York, NY, USA, 36–44. https://doi.org/10.1145/1032604.1032612

[48] Jonathan Harper and Maneesh Agrawala. 2014. Deconstructing and restyling D3 visualizations. *UIST 2014 - Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (10 2014), 253–262. https://doi.org/10.1145/2642918.2647411

[49] A. Harth, Aidan Hogan, Renaud Delbru, Jürgen Umbrich, Seán O'Riain, and Stefan Decker. 2007. SWSE: Answers Before Links!. In *Semantic Web Challenge*.

[50] Marti Hearst. 2009. *Search user interfaces*. Cambridge university press.

[51] Jeff Heflin, James A. Hendler, and Sean Luke. 2003. SHOE: A Blueprint for the Semantic Web. In *Spinning the Semantic Web*.

[52] Enamul Hoque and Maneesh Agrawala. 2019. Searching the visual style and structure of D3 visualizations. *IEEE Transactions on Visualization and Computer Graphics* 26, 1 (2019), 1236–1245.

[53] Enamul Hoque, Vidya Setlur, Melanie Tory, and Isaac Dykeman. 2017. Applying pragmatics principles for interaction with visual analytics. *IEEE Transactions on Visualization and Computer Graphics* 24, 1 (2017), 309–318.

[54] Yushi Jing and Shumeet Baluja. 2008. VisualRank: Applying PageRank to Large-Scale Image Search. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30, 11 (2008), 1877–1890. https://doi.org/10.1109/TPAMI.2008.121

[55] Jupyter. 2023. Jupyter. https://jupyter.org/.

[56] Sean Kandel, Andreas Paepcke, Joseph Hellerstein, and Jeffrey Heer. 2011. Wrangler: Interactive Visual Specification of Data Transformation Scripts. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, BC, Canada) *(CHI '11)*. Association for Computing Machinery, New York, NY, USA, 3363–3372. https://doi.org/10.1145/1978942.1979444

[57] Tadao Kasami. 1966. An efficient recognition and syntax-analysis algorithm for context-free languages. *Coordinated Science Laboratory Report no. R-257* (1966).

[58] Gjergji Kasneci, Fabian M. Suchanek, Georgiana Ifrim, Shady Elbassuoni, Maya Ramanath, and Gerhard Weikum. 2008. NAGA: harvesting, searching and ranking knowledge. In *SIGMOD Conference*.

[59] Esther Kaufmann, Abraham Bernstein, and Renato Zumstein. 2006. Querix: A natural language interface to query ontologies based on clarification dialogs. (01 2006).

[60] Dae Hyun Kim, Vidya Setlur, and Maneesh Agrawala. 2021. Towards Understanding How Readers Integrate Charts and Captions: A Case Study with Line Charts. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 610, 11 pages. https://doi.org/10.1145/3411764.3445443

[61] Dan Klein and Christopher D. Manning. 2003. Accurate Unlexicalized Parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Sapporo, Japan, 423–430. https://doi.org/10.3115/1075096.1075150

[62] Oleksandr Kolomiyets and Marie-Francine Moens. 2011. A survey on question answering technology from an information retrieval perspective. *Information Sciences* 181, 24 (2011), 5412–5434. https://doi.org/10.1016/j.ins.2011.07.047

[63] Yuangui Lei, Victoria S. Uren, and Enrico Motta. 2006. SemSearch: A Search Engine for the Semantic Web. In *International Conference Knowledge Engineering and Knowledge Management*.

[64] V. López, Michele Pasin, and Enrico Motta. 2005. AquaLog: An Ontology-Portable Question Answering System for the Semantic Web. In *Extended Semantic Web Conference*.

[65] Vanessa Lopez, Marta Sabou, and Enrico Motta. 2006. PowerMap: Mapping the Real Semantic Web on the Fly. In *Proceedings of the 5th International Conference on The Semantic Web* (Athens, GA) *(ISWC'06)*. Springer-Verlag, Berlin, Heidelberg, 414–427. https://doi.org/10.1007/11926078_30

[66] Jock Mackinlay, Pat Hanrahan, and Chris Stolte. 2007. Show me: Automatic presentation for visual analysis. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (2007), 1137–1144.

[67] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press, Cambridge, UK. http://nlp.stanford.edu/IR-book/information-retrieval-book.html

[68] Gary Marchionini. 2006. Exploratory search: from finding to understanding. *Commun. ACM* 49, 4 (2006), 41–46.

[69] Merriam-Webster. 2023. Olio. In *Merriam-Webster Dictionary*. https://www.merriam-webster.com/dictionary/olio

[70] Observable Metadata. 2023. Searching on Observable. https://observablehq.com/@observablehq/searching-on-observable#attributes.

[71] Selina Meyer, David Elsweiler, Bernd Ludwig, Marcos Fernandez-Pichel, and David E. Losada. 2022. Do We Still Need Human Assessors? Prompt-Based GPT-3 User Simulation in Conversational AI. In *Proceedings of the 4th Conference on Conversational User Interfaces* (Glasgow, United Kingdom) *(CUI '22)*. Association for Computing Machinery, New York, NY, USA, Article 8, 6 pages. https://doi.org/10.1145/3543829.3544529

[72] Microsoft. 2023. Microsoft PowerBI. https://powerbi.microsoft.com/.

[73] Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). http://arxiv.org/abs/1301.3781

[74] Vibhu Mittal, Steven Roth, Johanna Moore, Joe Mattis, and Giuseppe Carenini. 1995. Generating Explanatory Captions for Information Graphics. *Proceeedings of the International Joint Conference on Artificial Intelligence*, 1276–1283.

[75] Dan I. Moldovan and Rada Mihalcea. 2000. Using WordNet and Lexical Operators to Improve Internet Searches. *IEEE Internet Computing* 4, 1 (jan 2000), 34–43. https://doi.org/10.1109/4236.815847

[76] Kristi Morton, Magdalena Balazinska, Dan Grossman, Robert Kosara, and Jock Mackinlay. 2014. Public Data and Visualizations: How are Many Eyes and Tableau Public used for Collaborative Analytics? *ACM SIGMOD Record* 43, 2 (2014), 17–22.

[77] Observable. 2023. Observable Notebooks. https://observablehq.com/.

[78] Observable. 2023. Observable Notebooks. https://observablehq.com/@observablehq/searching-on-observable#attributes.

[79] OpenAI. 2023. ChatGPT. Retrieved March 23, 2023 from https://openai.com/blog/chatgpt

[80] Eyal Oren, Christophe Guéret, and Stefan Schlobach. 2008. Anytime Query Answering in RDF through Evolutionary Algorithms. In *Proceedings of the 7th International Conference on The Semantic Web* (Karlsruhe, Germany) *(ISWC '08)*. Springer-Verlag, Berlin, Heidelberg, 98–113. https://doi.org/10.1007/978-3-540-88564-1_7

[81] Srishti Palani, Yingyi Zhou, Sheldon Zhu, and Steven P. Dow. 2022. InterWeave: Presenting Search Suggestions in Context Scaffolds Information Search and Synthesis. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology* (Bend, OR, USA) *(UIST '22)*. Association for Computing Machinery, New York, NY, USA, Article 93, 16 pages. https://doi.org/10.1145/3526113.3545696

[82] Programmable Web. 2022. *Thesaurus API*. http://www.programmableweb.com/apitag/thesaurus

[83] Vijayshankar Raman and Joseph M. Hellerstein. 2001. Potter's Wheel: An Interactive Data Cleaning System. In *Proceedings of the 27th International Conference on Very Large Data Bases (VLDB '01)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 381–390.

[84] Jerome Ramos and Carsten Eickhoff. 2020. Search Result Explanations Improve Efficiency and Trust. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, China) *(SIGIR '20)*. Association for Computing Machinery, New York, NY, USA, 1597–1600. https://doi.org/10.1145/3397271.3401279

[85] Arvind Satyanarayan, Dominik Moritz, Kanit Wongsuphasawat, and Jeffrey Heer. 2016. Vega-lite: A grammar of interactive graphics. *IEEE Transactions on Visualization and Computer Graphics* 23, 1 (2016), 341–350.

[86] Jordan Sechler, Lane Harrison, and Evan M Peck. 2017. Sightline: Building on the web's visualization ecosystem. In *Proceedings of the 2017 CHI Extended Abstracts*. 2049–2055.

[87] Vidya Setlur, Sarah E. Battersby, Melanie Tory, Rich Gossweiler, and Angel X. Chang. 2016. Eviza: A Natural Language Interface for Visual Analysis. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (Tokyo, Japan) *(UIST 2016)*. ACM, New York, NY, USA, 365–377.

[88] Vidya Setlur, Enamul Hoque, Dae Hyun Kim, and Angel X. Chang. 2020. Sneak Pique: Exploring Autocompletion as a Data Discovery Scaffold for Supporting Visual Analysis. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '20)*. Association for Computing Machinery, New York, NY, USA, 966–978. https://doi.org/10.1145/3379337.3415813

[89] Vidya Setlur, Melanie Tory, and Alex Djalali. 2019. Inferencing Underspecified Natural Language Utterances in Visual Analysis. In *Proceedings of the 24th International Conference on Intelligent User Interfaces* (Marina del Ray, California) *(IUI '19)*. Association for Computing Machinery, New York, NY, USA, 40–51. https://doi.org/10.1145/3301275.3302270

[90] Leixian Shen, Enya Shen, Yuyu Luo, Xiaocong Yang, Xuming Hu, Xiongshuai Zhang, Zhiwei Tai, and Jianmin Wang. 2021. Towards natural language interfaces for data visualization: A survey. *arXiv preprint arXiv:2109.03506* (2021).

[91] Milad Shokouhi and Luo Si. 2011. Federated Search. In *Foundations and Trends in Information Retrieval*.

[92] Craig Silverstein, Hannes Marais, Monika Henzinger, and Michael Moricz. 1999. Analysis of a very large web search engine query log. In *Acm sigir forum*, Vol. 33. ACM New York, NY, USA, 6–12.

[93] Sivic and Zisserman. 2003. Video Google: a text retrieval approach to object matching in videos. In *Proceedings Ninth IEEE International Conference on Computer Vision*. 1470–1477 vol.2. https://doi.org/10.1109/ICCV.2003.1238663

[94] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. 2000. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22, 12 (2000), 1349–1380. https://doi.org/10.1109/34.895972

[95] Rohini K. Srihari and W. Li. 1999. Information Extraction Supported Question Answering. In *Text Retrieval Conference*.

[96] Arjun Srinivasan, Steven M Drucker, Alex Endert, and John Stasko. 2018. Augmenting visualizations with interactive data facts to facilitate interpretation and communication. *IEEE transactions on visualization and computer graphics* 25, 1 (2018), 672–681.

[97] Arjun Srinivasan, Nikhila Nyapathy, Bongshin Lee, Steven M Drucker, and John Stasko. 2021. Collecting and characterizing natural language utterances for specifying data visualizations. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–10.

[98] Arjun Srinivasan and Vidya Setlur. 2021. Snowy: Recommending utterances for conversational visual analysis. In *The 34th Annual ACM Symposium on User Interface Software and Technology*. 864–880.

[99] Arjun Srinivasan and John Stasko. 2018. Orko: Facilitating multimodal interaction for visual exploration and analysis of networks. *IEEE Transactions on Visualization and Computer Graphics* 24, 1 (2018), 511–521.

[100] Tableau Software. 2023. Tableau Public. Retrieved March 23, 2023 from https://public.tableau.com/

[101] Edward Thomas, Jeff Z. Pan, and Derek H. Sleeman. 2007. ONTOSEARCH2: Searching Ontologies Semantically. In *OWL: Experiences and Directions*.

[102] Melanie Tory and Vidya Setlur. 2019. Do what i mean, not what i say! design considerations for supporting intent and context in analytical conversation. In *2019 IEEE conference on visual analytics science and technology (VAST)*. IEEE, 93–103.

[103] Thanh Tran, Philipp Cimiano, Sebastian Rudolph, and Rudi Studer. 2007. Ontology-Based Interpretation of Keywords for Semantic Search. In *ISWC/ASWC*.

[104] Gonzalo Vaca-Castano and Mubarak Shah. 2015. Semantic Image Search From Multiple Query Images. In *Proceedings of the 23rd ACM International Conference on Multimedia* (Brisbane, Australia) *(MM '15)*. Association for Computing Machinery, New York, NY, USA, 887–890. https://doi.org/10.1145/2733373.2806356

[105] Fernanda B Viegas, Martin Wattenberg, Frank Van Ham, Jesse Kriss, and Matt McKeon. 2007. Manyeyes: a site for visualization at internet scale. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (2007), 1121–1128.

[106] Chong Wang, Miao Xiong, Qi Zhou, and Yong Yu. 2007. PANTO: A Portable Natural Language Interface to Ontologies. In *Proceedings of the 4th European Conference on The Semantic Web: Research and Applications* (Innsbruck, Austria) *(ESWC '07)*. Springer-Verlag, Berlin, Heidelberg, 473–487. https://doi.org/10.1007/978-3-540-72667-8_34

[107] Wesley Willett, Jeffrey Heer, and Maneesh Agrawala. 2007. Scented widgets: Improving navigation cues with embedded visualizations. *IEEE transactions on visualization and computer graphics* 13, 6 (2007), 1129–1136.

[108] Zhibiao Wu and Martha Palmer. 1994. Verbs semantics and lexical selection. In *Proceedings of ACL*. 133–138.

[109] Daniel H Younger. 1967. Recognition and parsing of context-free languages in time n3. *Information and control* 10, 2 (1967), 189–208.

[110] Li Yujian and Liu Bo. 2007. A normalized Levenshtein distance metric. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29, 6 (2007), 1091–1095.

[111] Gideon Zenz, Xuan Zhou, Enrico Minack, Wolf Siberski, and Wolfgang Nejdl. 2009. From Keywords to Semantic Queries-Incremental Query Construction on the Semantic Web. *Web Semant.* 7, 3 (sep 2009), 166–176. https://doi.org/10.1016/j.websem.2009.07.005

[112] Andrew Zhai, Hao-Yu Wu, Eric Tzeng, Dong Huk Park, and Charles Rosenberg. 2019. Learning a Unified Embedding for Visual Search at Pinterest. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (Anchorage, AK, USA) *(KDD '19)*. Association for Computing Machinery, New York, NY, USA, 2412–2420. https://doi.org/10.1145/3292500.3330739