Data-Semantics-Aware Recommendation of Diverse Pivot Tables

Whanhee Cho
University of Utah
Salt Lake City, Utah, USA
whanhee@cs.utah.edu

Anna Fariha
University of Utah
Salt Lake City, Utah, USA
afariha@cs.utah.edu

Abstract

Data summarization is essential to discover insights from large datasets. In spreadsheets, *pivot tables* offer a convenient way to summarize tabular data by computing aggregates over some attributes, grouped by others. However, identifying attribute combinations that will result in *useful* pivot tables remains a challenge, especially for high-dimensional datasets. We formalize the problem of automatically recommending *insightful* and *interpretable* pivot tables, eliminating the tedious manual process. A crucial aspect of recommending a *set* of pivot tables is to *diversify* them. Traditional works inadequately address the table-diversification problem, which leads us to consider the problem of *pivot table diversification*.

We present SAGE, a data-semantics-aware system for recommending k-budgeted diverse pivot tables, overcoming the shortcomings of prior work for top-k recommendations that cause redundancy. SAGE ensures that each pivot table is *insightful*, *interpretable*, and *adaptive* to the user's actions and preferences, while also guaranteeing that the set of pivot tables are different from each other, offering a *diverse* recommendation. We make two key technical contributions: (1) a *data-semantics-aware model* to measure the utility of a single pivot table and the diversity of a set of pivot tables, and (2) a *scalable greedy algorithm* that can efficiently select a set of diverse pivot tables of high utility, by leveraging data semantics to significantly reduce the combinatorial search space. Our extensive experiments on three real-world datasets show that SAGE outperforms alternative approaches, and efficiently scales to accommodate high-dimensional datasets. Additionally, we present several case studies to highlight SAGE's qualitative effectiveness over commercial software and Large Language Models (LLMs).

1 Introduction

Data is at the heart of data-driven decision making. We rely on data—specifically, trends observed in the data—to obtain *insights* [18] that help us make informed decisions. However, due to limitations in human comprehensibility, data must be *summarized* [23, 29, 46] to enable humans observe trends and discover insights from the summaries—either directly or via visualizations [48] over the summaries. One of the most common techniques to summarize data is *aggregation*. Simple aggregations involve functions (e.g, `SUM`) to aggregate all rows. More nuanced aggregations involve multiple groupings of the entities (e.g., `GROUP BY GENDER, MARITAL_STATUS`) and then aggregating each group separately.

While SQL provides functionalities for any custom aggregation query, it is not suitable for novices, and has various interface-related limitations. Thanks to the ubiquity of spreadsheet software—such as Microsoft Excel [34], Google Sheets [24], Apple Numbers [27], etc.—a substantial portion of businesses (around 60% [17]) and about 2 billion people [37] use spreadsheets for data management and analysis. These users rely on *pivot tables*, a summary of tabular data that computes aggregates over a few data attributes, grouped by other data attributes. Most commercial spreadsheets include a built-in and user-friendly mechanism to construct pivot tables. Spreadsheet pivot tables are particularly suitable for novices, where they can rearrange, group, and aggregate data using intuitive interfaces such as drag-and-drop. Unlike SQL aggregates, spreadsheet pivot tables offer dynamic user interactions, allowing interactive exploration such as drilling down, filtering, sorting, etc.

ID	Gender	Age	Experience	Degree	Department	Salary
1	Male	48	3	PhD	IT	$50,000
2	Female	32	1	MS	Sales	$20,000
3	Male	45	12	PhD	HR	$100,000

Figure 1: A sample table from an employee compensation dataset.

In an exploratory setting where the goal is to discover interesting data trends, a key challenge in constructing insightful pivot tables lies in selecting the right *parameters*, i.e., determining which attributes to use for groupings and aggregations. This task becomes even harder when users lack domain knowledge, face missing or cryptic attribute names, or work with high-dimensional data. In such cases, users must manually explore a vast space of parameter combinations through a tedious trial-and-error process. This involves experimenting with various combinations of (1) grouping attributes, (2) aggregation attributes, and (3) aggregation functions, then manually assessing the insightfulness of the resulting pivot tables. We illustrate this challenge with the following example.

Example 1.1. *Sasha is investigating potential factors affecting salary across various groups in an employee compensation dataset over* 21 *attributes including* `ID`, `Gender`, `Age`, `Experience`, `Degree`, `Department`, `Salary`, *etc. (Figure 1). With an aim to discover salary discrepancies across various group combinations, she starts with the pivot table shown in Figure 2: she puts* `Gender` *in the row-groups* Ⓐ *and* `Department` *in the column-groups* Ⓑ*; and chooses* `Salary` *as an aggregate attribute* Ⓒ *and* `Average` *as the aggregate function* Ⓓ.

Sasha is interested in `Salary` *discrepancies, so her choice for* Ⓒ *is fixed. However, she still needs to explore various combinations for* Ⓐ, Ⓑ, *and* Ⓓ. *Sasha wishes to put demographic attributes (e.g.,* `Gender`, `Race`, `Age`, `Marital Status`, *etc.) in the row-groups, as any discrepancy across different rows will indicate discrimination, and all other attributes in the column-groups. For this particular dataset, there are* 5 *demographic attributes and* 15 *non-demographic attributes. Sasha decides to explore* 4 *aggregation functions:* `MAX`, `MIN`, `AVERAGE`, *and* `SUM`, *as all of them can expose different types of discrepancies.*

This leaves her $5 \times 15 \times 4 = 300$ *possible combinations,*[1] *and she must carefully inspect each pivot table to identify salary discrepancies, by manually contrasting the pivot table cells. Assuming each pivot*

[1] Sasha chose only one option for each parameter. Multiple options (e.g., `Gender` and `Race` for row-groups) will further increase the search space of possible pivot tables.

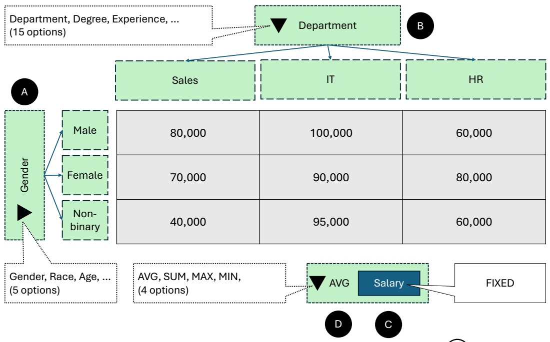


Figure 2: A pivot table requires 4 parameters: (A) row-groups, (B) column-groups, (C) aggregate attributes, and (D) aggregate functions for each aggregate attribute. Users can choose multiple values for each parameter. In Example 1.1, Sasha has fixed (C), but still needs to explore (A) (5 options), (B) (15 options), and (D) (4 options).

table has 10 *cells on average and it takes about* 2 *minutes to examine each pivot table, Sasha needs* $300 \times 2 = 600$ *minutes (*10 *hours)!*

Recommending Pivot Tables. Example 1.1 highlights the need for a smart *recommendation* system that can automatically suggest the "best" pivot tables, relieving Sasha from tedious, error-prone work. While existing spreadsheet software, such as Microsoft Excel and Google Sheets, are equipped with features for automatic pivot table recommendation, they have several shortcomings. We proceed to highlight some of their key limitations through Example 1.2.

Example 1.2. *Frustrated by the manual exploration, Sasha gives pivot table recommendation features of commercial spreadsheet software a try (Figure 3). Google Sheets provides three recommendations.*[2] *However, the recommended pivot tables often include too many aggregated attributes beyond what she wanted (*`Salary`*), resulting in convoluted and large pivot tables. Sasha also observes that most recommendations are redundant—they default to the groupings by* `Gender` *or* `Department`*—and lack diversity, causing her to miss out on insights involving other data attributes.*

While Microsoft Excel provides nine recommendations,[3] *it utilizes only 10 out of 21 possible attributes. Additionally, it suggests meaningless aggregations such as* `Sum(Employed Year)` *and* `Sum(Age)`*, revealing its shortcoming in grasping the data semantics. Furthermore, for both MS Excel and Google Sheets, Sasha failed to specify* `Salary` *as her intended aggregate attribute, restricting her ability to steer the recommendations towards her specific needs.*

Lastly, Sasha asks ChatGPT for three "insightful" and "diverse" pivot tables, focusing on average `Salary`*. While the recommendations initially seemed reasonable, she quickly realizes that the cell values of the pivot tables are entirely hallucinated,*[4] *revealing that ChatGPT ignored the actual data and failed to validate the insightfulness of the pivot tables. Sasha concludes that LLMs are ill-suited for this task, since they generate outputs heuristically, not by explicitly enumerating and scoring all options for pivot tables.*

Example 1.2 highlights several key limitations of existing tools for automatic recommendation of pivot tables. First, they do not cater to the user needs for a *focused* and *adaptive* recommendation of pivot tables. Second, they focus on top-k recommendations [16, 50, 51] and do not consider *diversification* [20], which may cause the users to miss certain data insights. Finally, existing approaches do not fully leverage the data and its semantics to ensure that the suggested pivot tables are *useful*, i.e., *insightful* and *interpretable*. We propose SAGE, a data-semantics-aware system for recommending k-budgeted diverse pivot tables, which overcomes the shortcomings of the existing approaches. We summarize the limitations of currently available tools and research work in Figure 4 to contrast them against SAGE.

Tool	Recommended Pivot Tables
Google Sheets [24]	(1) Average of Age, Years of Experience, Annual Bonus, Overtime Hours, Sick Days, Training Hours, Satisfaction Score, #Projects, #Promotions by Gender (2) Average of Performance Rating for each Gender by Department (3) Average of Age, Years of Experience, Performance Rating, Salary, Annual Bonus, Overtime Hours, Sick Days, Training Hours, Satisfaction Score, #Projects, #Promotions by Gender
Microsoft Excel [34]	(1) Count of ID by Degree (2) Count of ID by Department (3) Sum of Employed Year by Department (4) Sum of #Promotions by Employed Year and Degree (5) Sum of Age, Children, Performance Rating by Degree (6) Sum of Children, Performance Rating, Salary by Degree (7) Sum of Children, Performance Rating, Salary by Department (8) Sum of Salary by Employed Year and Gender (9) Sum of Employed Year by Gender and Degree
ChatGPT [39]	(1) Average Salary by Years of Experience and Training Hours (2) Average Salary by Department and Children (3) Average Salary by Age and Satisfaction Score

Figure 3: Google Sheets give redundant and convoluted recommendations; Microsoft Excel includes meaningless recommendations such as `SUM(Age)`; while ChatGPT recommendations look reasonable, they are data-content-unaware as the pivot table values are hallucinated.

Problem. The problem we study in this paper is recommending a *diverse set* of pivot tables, under a *size* constraint, while ensuring that each recommended pivot table is *useful*, meaning it is *insightful* and *interpretable*. Furthermore, we want to achieve two usability goals during recommendation: (1) *adaptivity*, which takes into consideration already explored pivot tables by the users, and (2) *customizability*, which enables the users to guide the recommendation process by specifying certain data attributes to prioritize.

Challenges. We now highlight three key challenges that are associated with the problem:

Challenge 1: semantic modeling of pivot table utility. A useful pivot table must be *insightful*, to inform the users interesting data trends, and *interpretable*, by not overwhelming the users with too much information. Traditional approaches [14, 18, 25, 48] ignore the *data semantics*; they use simple statistical measures to model insightfulness, which often fails to ensure interpretability and interestingness. For instance, high variance among artists across 100 hobby-related attributes might indicate a statistically significant trend, but is not very insightful (since it is expected that artists will have various hobbies). Furthermore, semantically modeling insightfulness and interpretability of a pivot table in a *multi-group setting* (e.g., group by `Gender`, `Department`, `Degree`) is non-trivial and is not addressed in prior work. In summary, how to model insightfulness and interpretability of a pivot table while remaining aware of the data semantics is a key challenge.

[2] Test conducted in February 2025.
[3] Tested on Microsoft Excel (Windows) version 2501, February 2025.
[4] Tested on ChatGPT (GPT-4o), February 2025.

Legends
● Always
◒ Partially
○ Not supported
⊗ Unknown

Desirable Properties	Commercial software				Research work		LLMs			This work
	Microsoft Excel [34]	Google Sheets [24]	PowerBI [18, 35]	Tableau [44]	DAISY [50]	AutoSuggest [51]	ChatGPT [39]	Llama3-instruct [33]	TableGPT [47]	**SAGE**
Budgeted recommendations	○	○	○	○	○	○	●	●	●	●
Guarantees syntactic validity	●	●	●	●	●	●	○	○	○	●
Guarantees semantic validity	○	○	●	●	●	●	○	○	○	●
Ensures interpretability	●	○	●	●	⊗	●	●	◒	◒	●
Adaptive to user actions	○	○	○	○	○	●	●	●	●	●
Allows user specifications	○	○	○	●	○	○	●	●	●	●
Ensures diversity	○	○	○	○	○	○	●	●	●	●
Attribute-name semantics aware	●	●	●	○	○	●	●	●	●	●
Attribute-order insensitive	○	○	●	●	○	○	○	○	○	●
Data-semantics aware	◒	○	●	○	●	○	◒	◒	◒	●
No additional requirements	●	●	●	○	○	○	○	○	○	●
Low-cost	●	●	●	●	●	●	○	●	●	●
Open-source	○	○	○	○	○	○	○	◒	◒	●

Figure 4: SAGE satisfies all desirable properties. While PowerBI, Tableau, DAISY, and AutoSuggest do not directly/always recommend pivot tables, we include them due to their capability to recommend data summaries. Code/software for DAISY and AutoSuggest is unavailable, thus we obtain their properties from the papers, and mark certain things as unknown. LLMs are not designed to directly recommend pivot tables, but they can be prompted to do so.

Challenge 2: modeling table diversity. Beyond recommending insightful and interpretable pivot tables, our goal is to also *diversify* the set of pivot tables. To the best of our knowledge, the notion of diversity in the context of pivot tables is not defined in prior work. Existing diversification approaches [9, 19–21] do not trivially extend for "table diversification", where the items under consideration are entire tables rather than individual tuples. Prior works for recommending insightful data summaries [50] or visualizations [48] do not consider diversity. For pivot table diversification, the key challenge is to develop an appropriate distance metric to model both the syntactic (e.g., attribute coverage) and semantic (e.g., provided insights) distances between a pair of pivot tables.

Challenge 3: developing an efficient system. Our goal is to recommend highly insightful and interpretable pivot tables, while ensuring diversity among them. Unlike insightfulness and interpretability, which can be measured for each pivot table in isolation, diversity requires considering a *set* of pivot tables. This leads to an NP-hard combinatorial search problem. Furthermore, evaluating insightfulness of a pivot table requires its materialization, which adds to the computational complexity. While greedy approaches with approximation guarantees [9, 10, 20, 38, 41] can alleviate the problem of combinatorial search, the requirement of materializing candidate pivot tables remain. Even with approximation algorithms [3, 12, 26, 49] for efficient materialization, without aggressive pruning before materialization, far too many candidates become the bottleneck. Therefore, a key challenge here is to develop mechanisms that can leverage semantic understanding of the data to prune unpromising pivot tables and avoid unnecessary materialization. Another challenge is to discover effective techniques that can "push down" [52] components of the diversity requirements to the search process to further prevent unnecessary pivot table materialization.

Contributions. Our main contribution is development of a novel system SAGE, for recommendation of a diverse set of useful pivot tables under a budget (size) constraint. Below, we provide the key contributions we make in this paper:

- We motivate and *formalize the problem* of budgeted recommendation of a diverse set of useful pivot tables, model it as a *constrained optimization problem*, and establish its *desiderata* (Section 2).
- We provide a formal model to measure the *utility* of a pivot table in terms of insightfulness and interpretability. Unlike prior work, our utility model leverages data semantics (Section 3).
- We establish the notion of *pivot-table diversification*, a key component of the problem we study in this paper. Our contribution lies in the formulation of a suitable *distance metric*—which considers both structural and semantic properties of pivot tables—and its application to diversifying a set of pivot tables (Section 4).
- To ensure SAGE's efficiency and practicality, we must tackle the NP-hardness of the problem. To reduce the search space, we introduce *aggressive semantic pruning*. To expedite the recommendation process, we leverage offline computation and "push down" diversity requirements to the search process.
- We present a thorough empirical analysis over 3 real-world datasets and present 3 case studies to qualitatively contrast SAGE against commercial software and LLMs. We show that SAGE can recommend diverse and high-utility pivot tables, outperforming prior approaches and other variants. We also find SAGE scalable and efficient in practice.

2 Recommending a Diverse set of Pivot Tables

In this section, we motivate the need for diversity and adaptivity during recommending pivot tables (Section 2.1). Then we develop the desiderata for the problem (Section 2.2) and formalize it (Section 2.3). Finally, we provide an overview of SAGE (Section 2.4).

2.1 The Need for Diversity and Adaptivity

A key limitation of top-k recommendation is that it may provide redundant information, causing the users to miss out on relatively less useful, but complementary data insights. Such lack of *diversity* may even mislead the users to believe in partial insights that are "half-true". Another shortcoming of existing approaches is that they are not *adaptive* to user actions, i.e., when the user acknowledges a recommendation, it should be excluded from the subsequent iterations. However, commercial pivot table recommendation features are not adaptive (Figure 4). We proceed to provide an example to highlight the need for *diverse* recommendation to help the users get a broader picture of the dataset, and *adaptive* [31, 45] recommendation to enable the user guide the recommendation process.

Example 2.1. *Recall from Example 1.1 that Sasha is interested in salary discrepancies. She initially finds two pivot tables (Figure 5 (a) & (b)) suggesting gender-based pay gap. However, a deeper pattern emerges when she expands her analysis using other aggregate functions (e.g.,* `COUNT`*) and discovers the pivot tables shown in Figure 5 (c) & (d), which provide her an additional context that the discrepancy stems from the hiring process: employee counts are uneven across degrees and departments. Sasha also notes that IT employees earn more than those in Sales, and PhDs earn more than others. This indicates*

Gender	Degree BS	MS	PhD
Male	200K	300K	1000K
Female	100K	200K	300K

(a) Avg. Salary by Gender and Degree

Gender	Department IT	Sales
Male	1000K	500K
Female	400K	200K

(b) Avg. Salary by Gender and Dept.

	Degree, Department					
	BS		MS		PhD	
Gender	IT	Sales	IT	Sales	IT	Sales
Male	4	1	8	1	10	1
Female	2	8	2	3	1	2

(c) Count ID by Gender, Degree, and Dept.

Degree	Department IT	Sales
BS	200K	100K
MS	300K	200K
PhD	900K	400K

(d) Avg. Salary by Degree and Dept.

Figure 5: Four pivot tables over the dataset of Figure 1. While (a) and (b) indicate gender-based salary gap, (c) and (d) add additional context.

degree- and department-based discrepancies, which are expected and acceptable. Sasha concludes that males earn more on average not due to gender bias, but largely because more male PhDs work at IT.[5]

Furthermore, in an incremental setting where Sasha iteratively requests for recommendations of a few pivot tables at a time, she expects the system to adapt to her actions. For instance, after she accepts or rejects the suggestions of Figure 5 (a) and (b), the system should avoid recommending redundant pivot tables that reiterate the same concept (gender-based salary gap) across other aspects (e.g., marital status).

2.2 Desiderata

We now provide five key desiderata for an ideal system for recommending a set of pivot tables:

D1. Each recommended pivot table must provide *insightful* [14, 48] and *semantically interesting* information. For instance, a significant gap in average salary across genders provides insight into gender-based pay gap. Furthermore, "average salary" is semantically interesting, where "sum of zip code" is not.

D2. Each pivot table must be *interpretable*, ensuring ease of comprehension by humans. For instance, a concise table with 10 cells is more interpretable than one with 1000 cells.

D3. While insightfulness and interpretability model the goodness of a single pivot table, a desirable property for a set of pivot tables is *diversity*. Thus, the recommended set of pivot tables must minimize redundancy, covering various data aspects.

D4. The system for pivot table recommendation must allow (I) *customizability*—allowing users to specify the desired size of the recommendation set, degree of diversity, data scope, etc.—and (II) *adaptiveness* to user actions.

D5. Finally, the system must be *efficient* and *scalable*—to ensure handling large and high-dimensional data effectively—and *accessible*—in terms of cost and availability.

2.3 Problem Formulation

We now formalize our problem for a single-relation database instance (dataset) D by defining a pivot table.

Definition 2.1 (Pivot Table). Given a dataset D over a set of attributes $\mathbf{A}$ and the domain of aggregate functions $\mathcal{F} = \{$COUNT, SUM, AVG, MIN, MAX$\}$, a pivot table $T(\mathbf{F}(\mathbf{V}), \mathbf{G})$ takes the following form:

$$\texttt{SELECT } \mathbf{F}(\mathbf{V}) \texttt{ FROM } D \texttt{ GROUP BY } \mathbf{G}$$

where, $\mathbf{G}=\{G_1, G_2, \dots\}\subseteq\mathbf{A}$ is a subset of attributes for grouping; $\mathbf{V}=\{V_1, V_2, \dots\}\subseteq\mathbf{A}$ is a subset of attributes for computing aggregates over; $\mathbf{G} \cap \mathbf{V} = \emptyset$ ensures that no attribute is used for both grouping and aggregation; $\mathbf{F}=\{F_1, F_2, \dots\}$ is a set of aggregate functions where $F_i\in\mathcal{F}$ and $|\mathbf{F}|=|\mathbf{V}|$; and with slight abuse of notation, $\mathbf{F}(\mathbf{V})$ denotes $F_1(V_1), F_2(V_2), F_3(V_3)\dots$

Tabular representation of a pivot table. For $T(\mathbf{F}(\mathbf{V}), \mathbf{G})$, with $|\mathbf{G}| \geq 2$, we fix as row-groups and column-groups (Figure 2) two non-empty sets $\mathbf{R}, \mathbf{C} \subset \mathbf{G}$, respectively, where $\mathbf{R} \cup \mathbf{C} = \mathbf{G}$, $\mathbf{R} \cap \mathbf{C} = \emptyset$.

Example 2.2. *Figure 5(c) represents a possible tabular representation for the pivot table:* `SELECT COUNT(ID) FROM D GROUP BY Gender, Degree, Department`. *Here,* $\mathbf{R}$=`{Gender}` *and* $\mathbf{C}$=`{Degree, Dept.}`

Pivot table canonicalization. The above mechanism allows structurally different tabular representations of semantically equivalent pivot tables. However, our focus is on the semantics of a pivot table and not the specific orientation of its tabular representation. Thus, we canonicalize a pivot table $T(\mathbf{F}(\mathbf{V}), \mathbf{G})$ by lexicographically sorting $\mathbf{F}(\mathbf{V})$ and $\mathbf{G}$ to obtain $\mathbf{F}(\mathbf{V})_{\leq}$ and $\mathbf{G}_{\leq}$, respectively, and derive the canonical pivot table $T(\mathbf{F}(\mathbf{V})_{\leq}, \mathbf{G}_{\leq})$. Furthermore, we obtain a *canonical tabular representation* of a pivot table $T(\mathbf{F}(\mathbf{V}), \mathbf{G})$ by assigning to the row-groups ($\mathbf{R}_{\leq}$) the first $\lceil \frac{|\mathbf{G}|}{2} \rceil$ elements of $\mathbf{G}_{\leq}$ and the remaining elements to the column-groups ($\mathbf{C}_{\leq}$). This process ensures that pivot tables remain *organization-invariant* (e.g., transpose-invariant), i.e., $T(\mathbf{F}(\mathbf{V}), \mathbf{G})$ and all its variants derived from different permutations of $\mathbf{F}(\mathbf{V})$ and $\mathbf{G}$ result in an identical canonical tabular representation.

Example 2.3. *The canonical pivot table for Figure 5(c) is:* `SELECT COUNT(ID) FROM D GROUP BY Degree, Department, Gender`. *The canonical tabular representation is obtained by setting* $\mathbf{R}_{\leq}$=⟨`Degree, Dept.`⟩ *and* $\mathbf{C}_{\leq}$=⟨`Gender`⟩*, which is simply the transpose of Figure 5(c).*

Based on the desiderata of Section 2.2, we set our goal to find a bounded sized (D4-I) set of pivot tables such that the overall utility (D1 & D2) of the pivot tables are maximized while the set of pivot tables meet the minimum diversity requirement (D3).

Problem 2.1 (Recommending a set of Pivot Tables). *Given (i) a set of possible pivot tables $\mathcal{T}_{\mathbf{A}}$ over a dataset D with attributes $\mathbf{A}$, (ii) a function $Utility : \mathcal{T}_{\mathbf{A}} \mapsto [0, 1]$ that returns the utility of a pivot table $T \in \mathcal{T}_{\mathbf{A}}$, (iii) a function $Diversity : 2^{\mathcal{T}_{\mathbf{A}}} \mapsto [0, 1]$ that returns the diversity of a set of pivot tables $\mathbf{T} \subseteq \mathcal{T}_{\mathbf{A}}$, (iv) a budget $k \in \mathbb{N}^+$, and (v) a threshold $\theta \in [0, 1]$, find a set of pivot tables $\mathbf{T}^* \subseteq \mathcal{T}_{\mathbf{A}}$ s.t:*

$$\begin{array}{ll} \textit{\textbf{(objective)}} & \mathbf{T}^* = \underset{\mathbf{T}\subseteq\mathcal{T}_{\mathbf{A}}}{\arg\max} \sum_{T\in\mathbf{T}} Utility(T), \\ \textit{\textbf{(size constraint)}} & |\mathbf{T}^*| \leq k, \textit{ and} \\ \textit{\textbf{(diversity constraint)}} & Diversity(\mathbf{T}^*) \geq \theta \end{array}$$

Problem 2.1 balances utility and diversity by maximizing utility while putting a constraint on diversity. Other variants of this problem are possible such as maximizing a linear combination of the objective and the diversity constraint.

[5] This phenomenon is commonly known as Simpson's paradox [7, 8]. While our focus is not to explicitly expose Simpson's paradox, we show this as a motivating use-case to highlight the need for diversification in pivot table recommendations.

Symbol	Description
$D, \mathbf{A}, D[\mathbf{A}]$	Database, attributes, possible unique value combinations
$\mathbf{G}, \mathbf{R}, \mathbf{C}$	Grouping attributes, row-groups, column-groups; $\mathbf{R} \cup \mathbf{C} = \mathbf{G}$
F, V	Aggregation function and attribute
$T(F(V), \mathbf{G})$ or T	A pivot table for the query SELECT $F(V)$ FROM D GROUP BY $\mathbf{G}$
T_{r_i}/T^{c_j}	The row/column of T with row/column header r_i/c_j
$T_{r_i}^{c_i}$	The pivot table cell with row header r_i and column header c_j
n, m	The cardinality of $D[\mathbf{R}]$, $D[\mathbf{C}]$

Figure 6: Table of notations. We use bold letters to denote sets.

Adaptive Recommendation of a set of Pivot Tables. In the adaptive version (D4-II), we discard the already explored pivot tables by the user $\mathbf{T}_u$ from the set $\mathcal{T}_\mathbf{A}$ to obtain $\mathcal{T}_\mathbf{A} - \mathbf{T}_u$. When the user highlights a data scope (D4-I) by specifying a subset of attributes $\mathbf{A}_u \subseteq \mathbf{A}$ they want to focus on, we set the possible pivot tables to $\mathcal{T}_{\mathbf{A}_u}$.

Considerations. Multiple aggregates within a pivot table is essentially equivalent to concatenating the corresponding single-aggregate pivot tables, i.e.,

$$T(\mathbf{F}(\mathbf{V}), \mathbf{G}) \equiv \bigcup_{F,V \in \mathbf{F},\mathbf{V}} T(F(V), \mathbf{G})$$

Therefore, for simplicity and to promote interpretability, we limit each pivot table to have exactly one aggregate. We use F and V to denote the aggregation function and attribute, respectively. We summarize the notations used in the rest of this paper in Figure 6.

A Note on Generalizability. While our work focuses on recommending pivot tables in spreadsheet environments, the techniques can be generalized to recommend aggregate queries in relational databases. Pivot tables can be represented by SQL aggregate queries involving `Group-by`, allowing SAGE's adaptation in RDBMS.

2.4 SAGE Overview

To model utility of a single pivot table, SAGE utilizes two properties, *insightfulness* and *interpretability*, covering both syntactic and semantic aspects (Section 3). SAGE models diversity based on the *semantic distance* between a pair of pivot tables (Section 4). SAGE performs some offline precomputation, based on which, it applies an online greedy algorithm to generate a set of diverse and useful pivot tables as a solution for Problem 2.1.

3 Utility of Pivot Table

In this section, we describe how we quantify the goodness or *utility* of a single pivot table. Based on the desiderata of Section 2.2, a pivot table has high utility if it offers *insights* (D1) while being easily *interpretable* by humans (D2). Thus, we use insightfulness (Section 3.1) and interpretability (Section 3.2) as the two primary building blocks to model the utility of a pivot table.

3.1 Insightfulness

Intuitively, an insightful pivot table must involve attributes that are *significant*, i.e., inherently interesting and relevant (§ 3.1.1). Furthermore, it should satisfy at least one of the following criteria: (1) provide high *informativeness* (§ 3.1.2), (2) highlight meaningful *trends* (§ 3.1.3), or (3) reveal *surprising* [11, 28, 46] findings (§ 3.1.4). We build on prior works [11, 14, 18, 25, 28, 48] that model insightfulness based on only statistical properties, but significantly extend it by taking a *semantics-aware* approach, enabled by LLMs [33].

3.1.1 *Attribute significance.*

Typically, not all data attributes are of interest by human users. For instance, grouping data by `Name` is typically much less insightful than by `Gender`. However, semantic understanding is required to figure out attribute significance. To this end, we consult an LLM [33] to determine the significance for an attribute A. When attribute name is missing or semantically meaningless (e.g., "Column 1"), we first query an LLM to suggest appropriate names for attributes by providing it with a small sample of the data. LLM's semantic-reasoning capability allows us to achieve this without any domain-specific pre-configuration. To avoid noise, we employ multiple paraphrased prompts while querying the LLM. In this work, we condition the LLM to return a simple binary answer (yes $\rightarrow$ 1/no$\rightarrow$ 0). However, this component can be replaced by a domain-aware model that can return the likelihood of an attribute being significant for a specific context. We compute attribute significance of a pivot table T, $S_{\text{sig}} : \mathcal{T}_\mathbf{A} \mapsto [0, 1]$ as follows:

$$S_{\text{sig}}(T) = \prod_{A \in \{V\} \cup \mathbf{G}} \text{Significance}(A) \tag{1}$$

Here, Significance : $\mathbf{A} \mapsto [0, 1]$ denotes the probability that an attribute $A \in \mathbf{A}$ is a significant attribute w.r.t human interest.

Example 3.1. *For Figure 5(d),* `Degree`, `Department`*, and* `Salary`*, all are significant attributes. Thus,* $S_{sig}(T) = 1 \times 1 \times 1 = 1$.

3.1.2 *Informativeness.*

A statistical way to measure informativeness within data is to measure spread of the values. Intuitively, if values in a pivot table deviate from each other significantly, the "entropy" is high, and so is the informativeness. In this work, we use *deviation* across different groups to model informativeness. Unlike prior works [14, 25, 48] that only consider a two-group setting (Male vs Female), we consider a multi-group setting.

Given a database D and a pivot table T with row-groups $\mathbf{R}$ and column-groups $\mathbf{C}$, let $D[\mathbf{R}]$ and $D[\mathbf{C}]$ be the set of row and column headers for T, respectively. E.g., for Figure 5(c), the row headers are {`Male`, `Female`} and the column headers are {(`BS`, `IT`), (`BS`, `Sales`), (`MS`, `IT`), (`MS`, `Sales`), (`PhD`, `IT`), (`PhD`, `Sales`)}. We use T_{r_i} (T^{c_i}) to denote the row (column) of T with row (column) header r_i (c_i). We use n and m to denote the number of rows $|D[\mathbf{R}]|$ and columns $|D[\mathbf{C}]|$ in T, respectively. We compute the row-wise and column-wise informativeness scores $S_{\text{inf}}^{\text{row}}$ and $S_{\text{inf}}^{\text{col}}$ as follows:

$$S_{\text{inf}}^{\text{row}}(T) = \frac{1}{\binom{n}{2}} \sum_{r_i, r_j \in D[\mathbf{R}] \text{ s.t } i<j} \frac{||T_{r_i}, T_{r_j}||_2}{\gamma \cdot m}$$

$$S_{\text{inf}}^{\text{col}}(T) = \frac{1}{\binom{m}{2}} \sum_{c_i, c_j \in D[\mathbf{C}] \text{ s.t } i<j} \frac{||T^{c_i}, T^{c_j}||_2}{\gamma \cdot n}$$

Here, γ is a normalization parameter, set to $\max(T) - \min(T)$, ensuring that $S_{\text{inf}}^{\text{row}}$ and $S_{\text{inf}}^{\text{col}}$ are bounded between 0 and 1. Also note that while we use Euclidean distance (L_2 distance), any other distance function such as L_1 distance can be used here. We compute the informativeness score $S_{\text{inf}} : \mathcal{T}_\mathbf{A} \mapsto [0, 1]$ by taking the maximum of the row-wise and column-wise informativeness scores:

$$S_{\text{inf}}(T) = \max(S_{\text{inf}}^{\text{row}}(T), S_{\text{inf}}^{\text{col}}(T)) \tag{2}$$

Example 3.2. *We first compute the pairwise distances along the rows of Figure 5(d):* $||T_{BS}, T_{MS}||_2$=141.4K, $||T_{BS}, T_{PhD}||_2$=761.6K*, and* $||T_{MS}, T_{PhD}||_2$=632.5K*. We normalize using* γ=900K−100K=800K *and* m=2*, resulting in normalized distances of* $[0.088, 0.476, 0.395]$.

Taking an average gives us $S^{row}_{inf}(T)$=0.32. *We similarly compute* $S^{col}_{inf}(T)$=0.22 *and obtain* $S_{inf}(T)$= max(0.32, 0.22)=0.32.

3.1.3 Trend. Trends observed in a pivot table provide insights. However, the degree of insightfulness hinges on two key factors: the *magnitude* of the trend metric and how *atypical* or rare it is. For instance, a positive correlation between income and years of service is generally expected—employees with longer tenures typically earn more. In contrast, a trend showing that new hires earn more on average than long-serving employees contradicts this expectation and thus is particularly insightful.

We use two metrics to quantify the magnitude of a trend: *correlation* and *ratio*. Furthermore, to assess the degree of a trend's rarity, we query an LLM, which is aware of a broader semantic context. Thus, our definition of the *trend score* for a pivot table combines (1) purely statistical insights, reflected in high correlation and consistent ratio across pivot table values and (2) semantic insights, captured through the LLM's assessment of the trend's atypicality.

Correlation. We use $\rho_{i,j}$ to denote the Pearson correlation coefficient between the rows T_{r_i} and T_{r_j}. Since consulting LLMs is costly, we only consider significant correlations and require the magnitude to be at least τ_ρ, a customizable threshold parameter, with a default value of 50%. The indicator function $[[|\rho_{i,j}|\geq\tau_\rho]]$ denotes if the correlation between the rows T_{r_i} and T_{r_j} is significant. We compute the row-wise correlation-trend score $S^{\text{row}}_{\text{cor}} : \mathcal{T}_{\mathbf{A}} \mapsto [0, 1]$ as follows:

$$S^{\text{row}}_{\text{cor}}(T) = \frac{1}{\binom{n}{2}} \sum_{r_i,r_j \in D[\mathbf{R}] \text{ s.t. } i<j} |\rho_{i,j}| \cdot [[\,|\rho_{i,j}| \geq \tau_\rho]] \cdot \widetilde{Pr}_{\text{cor}}(r_i, r_j)$$

Here, $\widetilde{Pr}_{\text{cor}}(r_i, r_j)$ is the likelihood of *not* observing a high correlation between the groups r_i and r_j determined by an LLM. We prompt an LLM "In a five-point scale from *very likely* to *very unlikely*, how likely is it that the <$F(V)$> for <r_i> and <r_j> are <positively/negatively> correlated across <$D[\mathbf{C}]$>?", where each part within <> is replaced with actual values such as "Average Salary" for <$F(V)$>. We then map the LLM-provided likelihood to a numerical value using the mappings: *very likely* → 20%, *likely* → 40%, *neutral* → 60%, *unlikely* → 80%, and *very unlikely* → 100%. The intuition behind this *inverse* mapping is that the more unlikely it is to observe a trend, the more insightful it is. We compute $S^{\text{col}}_{\text{cor}}(T)$ similarly and set the correlation-trend score $S_{\text{cor}}(T) = \max(S^{\text{row}}_{\text{cor}}(T), S^{\text{col}}_{\text{cor}}(T))$.

Example 3.3. *In Figure 5(d),* T_{BS} *and* T_{PhD} *exhibit a positive correlation of 98%, which is likely (from LLM consultation), leading* $\widetilde{Pr}_{cor}(BS, PhD)$ *to be* 40%. *The correlation between* T_{BS} *and* T_{MS} *is 100% (very likely); and* T_{MS} *and* T_{PhD} *is 100% (likely). Since all correlations meet the threshold 50%, we compute* $S^{row}_{cor}(T) = (0.98 \times 0.4 + 1.0 \times 0.2 + 1.0 \times 0.4)/3 = 0.33$. *The column-wise correlation-trend score* $S^{col}_{cor}(T)$=(0.98 × 0.4)/1=0.39. *Thus,* $S_{cor}(T)$= max(0.33, 0.39)=0.39.

Ratio. Since correlation fails to capture the relative *magnitude*, we use *ratio trends* based on *persistent* ratios between two groups, e.g., T_{PhD} earning at least 5× more than T_{BS} across all departments. Similar to correlation trends, LLMs inform us the rarity of ratio trends. We compute the row-wise ratio-trend score $S^{\text{row}}_{\text{ratio}} : \mathcal{T}_{\mathbf{A}} \mapsto [0, 1]$ as follows:

$$S^{\text{row}}_{\text{ratio}}(T) = \frac{1}{\binom{n}{2}} \sum_{r_i,r_j \in D[\mathbf{R}]} (1 - \frac{1}{\pi_{i,j}}) \cdot [[\pi_{i,j} \geq \tau_\pi]] \cdot \widetilde{Pr}_{\text{ratio}}(r_i, r_j)$$

Here, $\pi_{i,j}$ denotes the minimum element-wise ratio between T_{r_i} and T_{r_j}, i.e., the smallest factor by which any value in T_{r_i} exceeds its corresponding value in T_{r_j}. To reduce LLM consultation cost, we use a threshold τ_π and require $\pi_{i,j}$ to be at least τ_π before consulting an LLM. While we set τ_π=2.0, it is a customizable parameter (but must be ≥ 1). $\widetilde{Pr}_{\text{ratio}}(r_i, r_j)$ denotes the LLM-provided likelihood of *not* observing the ratio trend between T_{r_i} and T_{r_j}. Note that for any i and j, at most one of $\pi_{i,j}$ or $\pi_{j,i}$ can contribute to this score, hence we fix the scaling factor to $\binom{n}{2}$. We normalize the trend magnitude by subtracting the inverse of $\pi_{i,j}$ from 1, so that larger ratios yield higher scores. We compute the column-wise ratio-trend score $S^{\text{col}}_{\text{ratio}}(T)$ similarly and set the ratio-trend score $S_{\text{ratio}}(T) = \max(S^{\text{row}}_{\text{ratio}}(T), S^{\text{col}}_{\text{ratio}}(T))$.

Example 3.4. *In Figure 5(d), the minimum ratio between* T_{MS} *and* T_{BS} *is 1.5; between* T_{PhD} *and* T_{BS} *is 4.0; and between* T_{PhD} *and* T_{MS} *is 2.0. After applying the threshold* $\tau_\pi = 2.0$, *the ratio trends for* (T_{PhD}, T_{MS}) *and* (T_{PhD}, T_{BS}) *are retained. The LLM returns the likelihoods:* `[Very Unlikely, Unlikely]` → [1.0, 0.8] *for these two trends. This gives us the row-wise ratio-trend score* $S^{row}_{ratio}(T) = (3/4 \times 1.0 + 1/2 \times 0.8)/3 = 0.37$. *For the column-wise ratio-trend score, no pair satisfies the threshold requirement and thus the score is* 0.0. *Therefore, the ratio-trend score* $S_{ratio}(T) = \max(0.37, 0.0) = 0.37$.

Finally, we compute the trend score by taking the maximum of the correlation-trend and ratio-trend scores:

$$S_{\text{trend}}(T) = \max\left(S_{\text{cor}}(T),\ S_{\text{ratio}}(T)\right) \tag{3}$$

Example 3.5. *For the pivot table of Figure 5(d), we computed the correlation-trend score as 0.39 in Example 3.3 and the ratio-trend score as 0.37 in Example 3.4 Thus,* $S_{trend}(T) = \max(0.39, 0.37) = 0.39$.

3.1.4 Surprise. Surprising values or *outliers* often indicate existence of insights. E.g., in Figure 5(d), $T^{\text{IT}}_{\text{PhD}}$ is exceptionally high (900K) in its column. While such outliers can be insightful, not all are. Some, like this one, are expected: IT is high-paying, and PhDs earn more. In contrast, an unusually high $T^{\text{Sales}}_{\text{BS}}$ would be surprising, and thus insightful. Beyond simply identifying outliers, we incorporate the unexpectedness of observing outliers using LLM's semantic knowledge. We compute the row-wise surprise score $S^{\text{row}}_{\text{sur}} : \mathcal{T}_{\mathbf{A}} \mapsto [0, 1]$ as follows:

$$S^{\text{row}}_{\text{sur}}(T) = \frac{1}{n} \textstyle\sum_{r_i \in D[\mathbf{R}]} OutlierScore(T_{r_i})$$

$$\text{where, } OutlierScore(T_{r_i}) = \begin{cases} 1 - \frac{\sum_{c \in O_{r_i}} \widetilde{Pr}_{\text{outlier}}(r_i, c, T^{c}_{r_i})}{|O_{r_i}|+1}, & \text{if } |O_{r_i}| > 0 \\ 0, & \text{otherwise} \end{cases}$$

$$O_{r_i} = \{c_j \in D[\mathbf{C}] \ s.t.\ |T^{c_j}_{r_i} - \mu(T_{r_i})| \geq \tau_O \cdot \sigma(T_{r_i})\}$$

O_{r_i} is the set of column headers for each outlier in T_{r_i}. E.g., O_{PhD} = {IT}, if $900K$ is an outlier for the row T_{PhD}. We ensure that the score increases with the number of outliers by subtracting the inverse of their count from 1. However, we believe that even a single outlier should contribute meaningfully to the score. To reflect this, we add 1 to the denominator so that having only one outlier results in a score multiplier of 0.5. The threshold τ_O is set to 4, since, assuming normal distribution, 99.99% of the population is expected to lie within 4 standard deviations from the average [22][6]; and anything

[6]Data in pivot tables may not be normally distributed. For simplicity, we use this heuristic for outlier detection. Our contribution is not inventing outlier-detection methods and the user is free to replace this with more suitable methods.

outside this range is an outlier. $\widetilde{Pr}_{\text{outlier}}(r_i, c, T^c_{r_i})$ denotes the LLM-obtained likelihood of $T^c_{r_i}$ *not* being an outlier w.r.t T_{r_i}. We compute $S^{\text{col}}_{\text{sur}}(T)$ similarly and compute the surprise score:

$$S_{\text{sur}}(T) = \max(S^{\text{row}}_{\text{sur}}(T), S^{\text{col}}_{\text{sur}}(T)) \tag{4}$$

Computing Insightfulness. A pivot table is considered insightful if it exhibits any of the key characteristics: informativeness, trend, or surprise. Therefore, we take the maximum of the three scores—S_{inf}, S_{trend}, and S_{sur}—as computed in Equations 2, 3, and 4 to compute $Insightfulness : \mathcal{T}_{\mathbf{A}} \mapsto [0, 1]$. This approach prioritizes the strongest signal among the three insight indicators. To prioritize pivot tables involving significant attributes, we scale this score by S_{sig} (Equation 1).

$$Insightfulness(T) = S_{\text{sig}}(T) \cdot \max\left(S_{\text{inf}}(T),\ S_{\text{trend}}(T),\ S_{\text{sur}}(T)\right) \tag{5}$$

Example 3.6. *For the pivot table of Figure 5(d), the attribute significance score $S_{sig}(T)$=1 (Example 3.1). The informativeness score $S_{inf}(T)$=0.32 (Example 3.2), the trend score $S_{trend}(T)$=0.32 (Example 3.5), and since there is no outlier in the pivot table, the surprise score $S_{sur}(T)$=0.0. Thus $Insightfulness(T)$=1 × max(0.32, 0.39, 0.0)=0.39.*

Remark. Presence of outliers can inflate the normalization factor γ (Section 3.1.2), causing S_{inf} to shrink significantly due to the compression of the value range of non-outliers. However, such cases typically yield a high S_{sur}, complementing the low S_{inf}. Since the *Insightfulness* score is the maximum of the three components, a strong signal from any source suffices to indicate insightfulness.

3.2 Interpretability

While insightfulness is a key measure of a pivot table's utility, it does not take into account the cognitive constraints of human users, who require *interpretability.* Consider the pivot table in Figure 7, which shows `SUM(AGE)` grouped by `Degree`, `Employed Year`, and `Department`. Its interpretability suffers due to: (i) high sparsity resulting from many value combinations yielding empty sets, such as no MS hire in IT in 2011, yielding *nulls* after aggregation (§ 3.2.1), (ii) semantically invalid aggregate `SUM(AGE)` (§ 3.2.2), and (iii) excessive columns from fine-grained yearly grouping, which compromises conciseness of the pivot table (§ 3.2.3). We proceed to describe three desirable interpretability properties of a pivot table.

3.2.1 Density. Each pivot table cell maps to a data subset under a specific value combination (e.g., MS hires in IT in 2011), so empty subsets are expected. When aggregated, these empty subsets produce *null* values. However, excessive nulls hinder interpretability [1], as humans struggle to draw insights from sparse tables. A common workaround is imputation with zeros. However it is misleading, as nulls denote missing data, whereas zeros may imply valid values (e.g., 0 for average temperature suggests an actual recording). This motivates a key interpretability criterion: high *density.* We compute the *density score* $S_{\text{den}} : \mathcal{T}_{\mathbf{A}} \mapsto [0, 1]$ as follows:

$$S_{\text{den}}(T) = \frac{\sum_{(r_i, c_j) \in D[\mathbf{R}] \times D[\mathbf{C}]} [[T^{c_j}_{r_i} \neq null]]}{n \cdot m} \tag{6}$$

Example 3.7. *The pivot table of Figure 5(d) has 3 rows and 2 columns (total 6 cells) and no null values. Hence, $S_{den}(T) = \frac{6}{2\times3} = 1.0$.*

	Employed Year						
	2011		2012		...	2024	
Degree	**IT**	**Sales**	**IT**	**Sales**		**IT**	**Sales**
BS	428	304	251	*null*	...	256	192
MS	*null*	*null*	*null*	*null*	...	*null*	*null*
PhD	603	650	441	450	...	*null*	*null*

Figure 7: pivot table for the query `SELECT SUM(AGE) GROUP BY EMPLOYED_YEAR, DEGREE` with several interpretability issues.

3.2.2 Semantic validity. Row and column headers in a pivot table represent unique values of the grouping attributes in **G**. For interpretability, these headers must be semantically meaningful [6]. E.g., `Degree` with values `{MS, BS, PhD}` is interpretable, while a functionally equivalent `Degree_ID` with values `{1, 2, 3}` is not, due to the lack of direct semantic meaning [13]. Similarly, the aggregate function F must be semantically valid w.r.t V: `AVG` is semantically valid for `AGE`, but `SUM` is not [30]. Though intuitive for humans, such judgments require domain knowledge. Thus, we leverage an LLM to mimic human reasoning and assess aggregation semantics. We define the *semantic validity score* of $T(F(V), \mathbf{G})$ based on two criteria: (1) whether the data types of attributes in **G** are textual, and (2) the extent to which F is semantically valid w.r.t V.

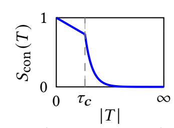
$$S_{\text{sem}}(T) = \frac{|\{A \in \mathbf{G} \text{ s.t } \mathtt{DataType}(A) \text{ is Text}\}|}{|\mathbf{G}|} \cdot Pr_{\text{agg}}(F, V) \tag{7}$$

We compute $Pr_{\text{agg}}(F, V)$ from an LLM-generated ranking of $F \in \mathcal{F}$ based on its semantic validity w.r.t V. We score the best function 1.0, the next 0.8, and so on, which ensures that $S_{\text{sem}}(T) \in [0, 1]$.

Example 3.8. *The pivot table of Figure 5(d), contains only textual headers. LLM responds to our prompt "Rank the functions* `COUNT`, `AVG`, `SUM`, `MIN`, *and* `MAX`, *based on their appropriateness for analyzing* `Salary`*" with {*`AVG`*, ...}. Thus, $S_{sem}(T) = \frac{2}{2} \times 1.0 = 1.0$.*

3.2.3 Conciseness. While multiple grouping attributes may enhance a pivot table's insightfulness, too many cells reduce its comprehensibility [40], and, thus, interpretability. To model this, we define conciseness score $S_{\text{con}} : \mathcal{T}_{\mathbf{A}} \mapsto [0, 1]$ using a piecewise function [15]:

$$S_{\text{con}}(T) = \begin{cases} 1 - z|T|, & \text{if } |T| \leq \tau_c \\ (1 - z\tau_c)e^{-\lambda(|T| - \tau_c)}, & \text{if } |T| > \tau_c \end{cases} \tag{8}$$



Here, $|T|$ denotes the number of cells in T. This formula captures the intuition that interpretability declines gradually at first, but drops sharply once $|T|$ exceeds a threshold τ_c, set to 16. We apply a 3% linear decrease ($z = 0.03$) until $|T|$ exceeds τ_c, and an exponential decay at a rate of 50% ($\lambda = 0.5$) beyond that. This is grounded in cognitive load theory [5, 36], which states that performance declines sharply when cognitive demand exceeds working-memory capacity.

Example 3.9. *The pivot table of Figure 5(d) has* 6 *cells. Since* 6<16, *we compute the linear part:* 1−0.03×6=0.82. *Thus, $S_{con}(T)$=0.82.*

Computing Interpretability. Unlike insightfulness, where a strong signal from *any* single type of insight is sufficient, interpretability demands that *all* criteria be met simultaneously. Therefore, we compute $Interpretability : \mathcal{T}_{\mathbf{A}} \mapsto [0, 1]$ as the average of the three scores: S_{den}, S_{sem}, and S_{con} (Equations 6, 7, and 8):

$$Interpretability(T) = \frac{S_{\text{den}}(T) + S_{\text{sem}}(T) + S_{\text{con}}(T)}{3} \tag{9}$$

Example 3.10. *For Figure 5(d), we obtained values for S_{den} (T), S_{sem} (T), and S_{con} (T) to be 1.0, 1.0, and 0.82, respectively, in the previous examples. Thus, $Interpretability(T)$ =(1.0+1.0+0.82)/3=0.94.*

3.3 Computing Utility

We now define $Utility : \mathcal{T}_{\mathbf{A}} \mapsto [0, 1]$ of a pivot table T by combining $Insightfulness$ (Eq. 5) and $Interpretability$ (Eq. 9). To balance their contributions, we introduce a tunable parameter α, set to 0.5 by default to give equal weight to both components. However, α can be adjusted to reflect application-specific preferences.

$$Utility(T) = \alpha \cdot Insightfulness(T) + (1 - \alpha) \cdot Interpretability(T)$$

Example 3.11. *For Figure 5(d), the $Insightfulness$ and $Interpretability$ scores are 0.39 and 0.94, respectively. Thus, Utility(T) = $0.5 \times 0.39 + 0.5 \times 0.94 = 0.67$.*

4 Diversity in a Set of Pivot Tables

While utility quantifies the goodness of a single pivot table in isolation, *diversity* captures how well a *set* of pivot tables, *collectively*, provide complementary and unique perspectives on the data (D3). High diversity in a set of pivot tables is achieved when the pivot tables are distant from each other with respect to data coverage and the insights they provide.

Diversity. Following Max-Min diversification [2], we define diversity of a set of pivot tables $\mathbf{T} = \{T_1, T_2, \dots\}$ by the smallest pairwise distance between $\mathbf{T}$'s elements. More formally, given a symmetric distance function $dist : \mathcal{T}_{\mathbf{A}} \times \mathcal{T}_{\mathbf{A}} \mapsto [0, 1]$, we define $diversity : 2^{\mathcal{T}_{\mathbf{A}}} \mapsto [0, 1]$ of a set of pivot tables $\mathbf{T} \subseteq \mathcal{T}_{\mathbf{A}}$ as follows:

$$Diversity(\mathbf{T}) = \min_{T_i, T_j \in \mathbf{T} \text{ s.t. } i<j} dist(T_i, T_j)$$

Distance between pivot tables. A simple heuristic to model $dist$ is the degree of disjointness between the attributes that define the pivot-table queries—if two pivot tables operate on the same set of attributes, the distance is 0; for completely disjoint set of attributes, the distance is 1. However, this heuristic fails to account for the *structural semantics* of the pivot-table queries and the *content semantics* of the data in the pivot tables. To this end, we employ a semantics-preserving embedding function $E : \mathcal{T}_{\mathbf{A}} \mapsto [-1, 1]^p$, which maps a pivot table $T \in \mathcal{T}_{\mathbf{A}}$ to a p-dimensional vector. Then, we compute distances between pivot tables in this embedding space:

$$dist(T_1, T_2) = \frac{1 - cosine\_similarity(E(T_1), E(T_2))}{2}$$

Here, $cosine\_similarity : \mathbb{R}^p \times \mathbb{R}^p \to [-1, 1]$ is a widely used measure for comparing embeddings [4, 43]. We divide by 2 to achieve normalization s.t $dist(T_1, T_2) \in [0, 1]$.

Pivot-table embedding. Pivot-table embedding should capture both the syntactic and semantic characteristics of the pivot table. To this end, we combine the query embedding E_Q and the content embedding E_C through concatenation, i.e., $E(T) = [E_Q(T); E_C(T)]$. Concatenating embeddings is a widely used technique in machine learning, natural language processing, and multi-modal learning [54].

Query embedding. Queries define the structural intent of a pivot table. While they reference data attributes and are aware of the schema, they are agnostic to the pivot table's content. Thus, the same pivot-table query over different database instances with an identical schema should yield the same query embedding. Effective query embeddings must also capture the semantics of the attribute names and reflect the query semantics. E.g., `GROUP BY Income` and `GROUP BY Salary` are semantically similar and should therefore be close in the query-embedding space. To this end, we use T5 [42], a natural-language encoder fine-tuned over a Text-to-SQL dataset [53], to obtain the query embedding $E_Q : \mathcal{T}_{\mathbf{A}} \mapsto [-1, 1]^{1024}$.

Content embedding. The content embedding must capture both the statistical and distributional properties of the pivot-table data and structural relationships among its attributes and tuples. In this work, we leverage TAPEX [32], a pre-trained encoder trained on sentence-table pairs, which is designed to understand both the structure and content of tabular data. This results in a content embedding $E_C : \mathcal{T}_{\mathbf{A}} \mapsto [-1, 1]^{1024}$.

References

[1] 2023. Microsoft Community Forum: Issues with null values in pivot tables. https://answers.microsoft.com/en-us/msoffice/forum/all/how-to-get-rid-of-blank-appearing-in-pivot-table/449f785a-f993-4c40-922b-1b52f02571ce. Accessed: 2025-06-02.

[2] Raghavendra Addanki, Andrew McGregor, Alexandra Meliou, and Zafeiria Moumoulidou. 2022. Improved Approximation and Scalability for Fair Max-Min Diversification. In *25th International Conference on Database Theory, ICDT 2022, March 29 to April 1, 2022, Edinburgh, UK (Virtual Conference) (LIPIcs, Vol. 220)*, Dan Olteanu and Nils Vortmeier (Eds.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 7:1–7:21. doi:10.4230/LIPICS.ICDT.2022.7

[3] Sameet Agarwal, Rakesh Agrawal, Prasad Deshpande, Ashish Gupta, Jeffrey F. Naughton, Raghu Ramakrishnan, and Sunita Sarawagi. 1996. On the Computation of Multidimensional Aggregates. In *VLDB'96, Proceedings of 22th International Conference on Very Large Data Bases, September 3-6, 1996, Mumbai (Bombay), India*, T. M. Vijayaraman, Alejandro P. Buchmann, C. Mohan, and Nandlal L. Sarda (Eds.). Morgan Kaufmann, 506–521.

[4] Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2017. A Simple but Tough-to-Beat Baseline for Sentence Embeddings. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

[5] Pierre Barrouillet, Sophie Bernardin, Sophie Portrat, Evie Vergauwe, and Valérie Camos. 2007. Time and cognitive load in working memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 33, 3 (2007), 570.

[6] Leilani Battle and Jeffrey Heer. 2019. Characterizing Exploratory Visual Analysis: A Literature Review and Evaluation of Analytic Provenance in Tableau. *Comput. Graph. Forum* 38, 3 (2019), 145–159. doi:10.1111/CGF.13678

[7] P. J. Bickel, E. A. Hammel, and J. W. O'Connell. 1975. Sex Bias in Graduate Admissions: Data from Berkeley. *Science* 187, 4175 (1975), 398–404. doi:10.1126/science.187.4175.398 arXiv:https://www.science.org/doi/pdf/10.1126/science.187.4175.398

[8] Colin R. Blyth. 1972. On Simpson's Paradox and the Sure-Thing Principle. *J. Amer. Statist. Assoc.* 67, 338 (1972), 364–366.

[9] Allan Borodin, Hyun Chul Lee, and Yuli Ye. 2012. Max-Sum diversification, monotone submodular functions and dynamic updates. In *Proceedings of the 31st ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS 2012, Scottsdale, AZ, USA, May 20-24, 2012*, Michael Benedikt, Markus Krötzsch, and Maurizio Lenzerini (Eds.). ACM, 155–166. doi:10.1145/2213556.2213580

[10] Jaime G. Carbonell and Jade Goldstein. 1998. The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries. In *SIGIR '98: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August 24-28 1998, Melbourne, Australia*, W. Bruce Croft, Alistair Moffat, C. J. van Rijsbergen, Ross Wilkinson, and Justin Zobel (Eds.). ACM, 335–336. doi:10.1145/290941.291025

[11] Minmin Chen, Yuyan Wang, Can Xu, Ya Le, Mohit Sharma, Lee Richardson, Su-Lin Wu, and Ed H. Chi. 2021. Values of User Exploration in Recommender Systems. In *RecSys '21: Fifteenth ACM Conference on Recommender Systems, Amsterdam, The Netherlands, 27 September 2021 - 1 October 2021*, Humberto Jesús Corona Pampín, Martha A. Larson, Martijn C. Willemsen, Joseph A. Konstan, Julian J. McAuley, Jean Garcia-Gathright, Bouke Huurnink, and Even Oldridge (Eds.). ACM, 85–95. doi:10.1145/3460231.3474236

[12] Graham Cormode and S. Muthukrishnan. 2005. An improved data stream summary: the count-min sketch and its applications. *J. Algorithms* 55, 1 (2005), 58–75. doi:10.1016/J.JALGOR.2003.12.001

[13] Anna DeCastellarnau. 2018. A classification of response scale characteristics that affect data quality: a literature review. *Quality & Quantity* 52, 4 (2018), 1523–1559.

[14] Çagatay Demiralp, Peter J. Haas, Srinivasan Parthasarathy, and Tejaswini Pedapati. 2017. Foresight: Recommending Visual Insights. *Proc. VLDB Endow.* 10, 12 (2017), 1937–1940. doi:10.14778/3137765.3137813

[15] Dazhen Deng, Aoyu Wu, Huamin Qu, and Yingcai Wu. 2023. DashBot: Insight-Driven Dashboard Generation Based on Deep Reinforcement Learning. *IEEE Trans. Vis. Comput. Graph.* 29, 1 (2023), 690–700. doi:10.1109/TVCG.2022.3209468

[16] Mukund Deshpande and George Karypis. 2004. Item-based top-n recommendation algorithms. *ACM Transactions on Information Systems (TOIS)* 22, 1 (2004), 143–177.

[17] Diginomica. 2025. *Shadow IT Never Dies: Why Spreadsheets Are Still Running Your Business.* Accessed: 2025-01-21.

[18] Rui Ding, Shi Han, Yong Xu, Haidong Zhang, and Dongmei Zhang. 2019. QuickInsights: Quick and Automatic Discovery of Insights from Multi-Dimensional Data. In *Proceedings of the 2019 International Conference on Management of Data, SIGMOD Conference 2019, Amsterdam, The Netherlands, June 30 - July 5, 2019*, Peter A. Boncz, Stefan Manegold, Anastasia Ailamaki, Amol Deshpande, and Tim Kraska (Eds.). ACM, 317–332. doi:10.1145/3299869.3314037

[19] Marina Drosou, H. V. Jagadish, Evaggelia Pitoura, and Julia Stoyanovich. 2017. Diversity in Big Data: A Review. *Big Data* 5, 2 (2017), 73–84. doi:10.1089/BIG.2016.0054

[20] Marina Drosou and Evaggelia Pitoura. 2010. Search result diversification. *SIGMOD Rec.* 39, 1 (2010), 41–47. doi:10.1145/1860702.1860709

[21] Marina Drosou and Evaggelia Pitoura. 2012. DisC diversity: result diversification based on dissimilarity and coverage. *Proc. VLDB Endow.* 6, 1 (2012), 13–24. doi:10.14778/2428536.2428538

[22] Anna Fariha, Ashish Tiwari, Arjun Radhakrishna, Sumit Gulwani, and Alexandra Meliou. 2021. Conformance Constraint Discovery: Measuring Trust in Data-Driven Systems. In *SIGMOD '21: International Conference on Management of Data, Virtual Event, China, June 20-25, 2021*, Guoliang Li, Zhanhuai Li, Stratos Idreos, and Divesh Srivastava (Eds.). ACM, 499–512. doi:10.1145/3448016.3452795

[23] Kareem El Gebaly, Parag Agrawal, Lukasz Golab, Flip Korn, and Divesh Srivastava. 2014. Interpretable and Informative Explanations of Outcomes. *PVLDB* 8, 1 (2014), 61–72. doi:10.14778/2735461.2735467

[24] Google. 2025. Google Sheets - Online Spreadsheet Editor. Accessed: 2025-01-21.

[25] Camille Harris, Ryan A. Rossi, Sana Malik, Jane Hoffswell, Fan Du, Tak Yeon Lee, Eunyee Koh, and Handong Zhao. 2023. SpotLight: Visual Insight Recommendation. In *Companion Proceedings of the ACM Web Conference 2023, WWW 2023, Austin, TX, USA, 30 April 2023 - 4 May 2023*, Ying Ding, Jie Tang, Juan F. Sequeda, Lora Aroyo, Carlos Castillo, and Geert-Jan Houben (Eds.). ACM, 19–23. doi:10.1145/3543873.3587302

[26] Wassily Hoeffding. 1994. *Probability Inequalities for sums of Bounded Random Variables.* Springer New York, New York, NY, 409–426. doi:10.1007/978-1-4612-0865-5_26

[27] Apple Inc. 2025. Numbers - Apple (IN). Accessed: 2025-01-21.

[28] Mahmood Jasim, Christopher Collins, Ali Sarvghad, and Narges Mahyar. 2022. Supporting Serendipitous Discovery and Balanced Analysis of Online Product Reviews with Interaction-Driven Metrics and Bias-Mitigating Suggestions. In *CHI '22: CHI Conference on Human Factors in Computing Systems, New Orleans, LA, USA, 29 April 2022 - 5 May 2022*, Simone D. J. Barbosa, Cliff Lampe, Caroline Appert, David A. Shamma, Steven Mark Drucker, Julie R. Williamson, and Koji Yatani (Eds.). ACM, 9:1–9:24. doi:10.1145/3491102.3517649

[29] Manas Joglekar, Hector Garcia-Molina, and Aditya G. Parameswaran. 2019. Interactive Data Exploration with Smart Drill-Down. *IEEE Trans. Knowl. Data Eng.* 31, 1 (2019), 46–60. doi:10.1109/TKDE.2017.2685998

[30] Ralph Kimball. 2013. *The data warehouse toolkit : the definitive guide to dimensional modeling* (third edition. ed.). John Wiley & Sons, Incorporated.

[31] Liangda Li, Hongbo Deng, Anlei Dong, Yi Chang, Ricardo Baeza-Yates, and Hongyuan Zha. 2017. Exploring Query Auto-Completion and Click Logs for Contextual-Aware Web Search and Query Suggestion. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*, Rick Barrett, Rick Cummings, Eugene Agichtein, and Evgeniy Gabrilovich (Eds.). ACM, 539–548. doi:10.1145/3038912.3052593

[32] Qian Liu, Bei Chen, Jiaqi Guo, Morteza Ziyadi, Zeqi Lin, Weizhu Chen, and Jian-Guang Lou. 2022. TAPEX: Table Pre-training via Learning a Neural SQL Executor. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

[33] Meta. 2025. Llama3-Intruct. Accessed: 2025-02-25.

[34] Microsoft. 2025. Microsoft Excel - Spreadsheet Software. Accessed: 2025-01-21.

[35] Microsoft Corporation. 2025. Microsoft Power BI. Accessed: 2025-02-16.

[36] George A Miller. 1956. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review* 63, 2 (1956), 81.

[37] MOOC.org. n.d.. How Important is Excel in Business? Accessed: 2025-01-21.

[38] Zafeiria Moumoulidou, Andrew McGregor, and Alexandra Meliou. 2021. Diverse Data Selection under Fairness Constraints. In *24th International Conference on Database Theory, ICDT 2021, March 23-26, 2021, Nicosia, Cyprus (LIPIcs, Vol. 186)*, Ke Yi and Zhewei Wei (Eds.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 13:1–13:25. doi:10.4230/LIPICS.ICDT.2021.13

[39] OpenAI. 2025. ChatGPT. Accessed: 2025-02-14.

[40] Zening Qu and Jessica Hullman. 2018. Keeping Multiple Views Consistent: Constraints, Validations, and Exceptions in Visualization Authoring. *IEEE Trans. Vis. Comput. Graph.* 24, 1 (2018), 468–477. doi:10.1109/TVCG.2017.2744198

[41] Filip Radlinski and Susan T. Dumais. 2006. Improving personalized web search using result diversification. In *SIGIR 2006: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, Washington, USA, August 6-11, 2006*, Efthimis N. Efthimiadis, Susan T. Dumais, David Hawking, and Kalervo Järvelin (Eds.). ACM, 691–692. doi:10.1145/1148170.1148320

[42] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *J. Mach. Learn. Res.* 21 (2020), 140:1–140:67.

[43] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (Eds.). Association for Computational Linguistics, 3980–3990. doi:10.18653/V1/D19-1410

[44] Salesforce. 2024. Tableau. https://www.tableau.com.

[45] Sunita Sarawagi. 2000. User-Adaptive Exploration of Multidimensional Data. In *VLDB 2000, Proceedings of 26th International Conference on Very Large Data Bases, September 10-14, 2000, Cairo, Egypt*, Amr El Abbadi, Michael L. Brodie, Sharma Chakravarthy, Umeshwar Dayal, Nabil Kamel, Gunter Schlageter, and Kyu-Young Whang (Eds.). Morgan Kaufmann, 307–316.

[46] Sunita Sarawagi. 2001. User-cognizant multidimensional analysis. *VLDB J.* 10, 2-3 (2001), 224–239. doi:10.1007/s007780100046

[47] Aofeng Su, Aowen Wang, Chao Ye, Chen Zhou, Ga Zhang, Guangcheng Zhu, Haobo Wang, Haokai Xu, Hao Chen, Haoze Li, Haoxuan Lan, Jiaming Tian, Jing Yuan, Junbo Zhao, Junlin Zhou, Kaizhe Shou, Liangyu Zha, Lin Long, Liyao Li, Pengzuo Wu, Qi Zhang, Qingyi Huang, Saisai Yang, Tao Zhang, Wentao Ye, Wufang Zhu, Xiaomeng Hu, Xijun Gu, Xinjie Sun, Xiang Li, Yuhang Yang, and Zhiqing Xiao. 2024. TableGPT2: A Large Multimodal Model with Tabular Data Integration. arXiv:2411.02059 [cs.LG]

[48] Manasi Vartak, Sajjadur Rahman, Samuel Madden, Aditya G. Parameswaran, and Neoklis Polyzotis. 2015. SEEDB: Efficient Data-Driven Visualization Recommendations to Support Visual Analytics. *Proc. VLDB Endow.* 8, 13 (2015), 2182–2193. doi:10.14778/2831360.2831371

[49] Zhewei Wei and Ke Yi. 2011. Beyond simple aggregates: indexing for summary queries. In *Proceedings of the 30th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS 2011, June 12-16, 2011, Athens, Greece*, Maurizio Lenzerini and Thomas Schwentick (Eds.). PODS, 117–128. doi:10.1145/1989284.1989299

[50] Junjie Xing, Xinyu Wang, and H. V. Jagadish. 2024. Data-Driven Insight Synthesis for Multi-Dimensional Data. *Proc. VLDB Endow.* 17, 5 (2024), 1007–1019.

[51] Cong Yan and Yeye He. 2020. Auto-Suggest: Learning-to-Recommend Data Preparation Steps Using Data Science Notebooks. In *Proceedings of the 2020 International Conference on Management of Data, SIGMOD Conference 2020, online conference [Portland, OR, USA], June 14-19, 2020*, David Maier, Rachel Pottinger, AnHai Doan, Wang-Chiew Tan, Abdussalam Alawini, and Hung Q. Ngo (Eds.). ACM, 1539–1554. doi:10.1145/3318464.3389738

[52] Cong Yan, Yin Lin, and Yeye He. 2023. Predicate Pushdown for Data Science Pipelines. *Proc. ACM Manag. Data* 1, 2 (2023), 136:1–136:28. doi:10.1145/3589281

[53] Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, Zilin Zhang, and Dragomir R. Radev. 2018. Spider: A Large-Scale Human-Labeled Dataset for Complex and Cross-Domain Semantic Parsing and Text-to-SQL Task. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (Eds.). Association for Computational Linguistics, 3911–3921. doi:10.18653/V1/D18-1425

[54] Ye Zhang, Stephen Roller, and Byron C. Wallace. 2016. MGNC-CNN: A Simple Approach to Exploiting Multiple Word Embeddings for Sentence Classification. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, Kevin Knight, Ani Nenkova, and Owen Rambow (Eds.). The Association for Computational Linguistics, 1522–1527. doi:10.18653/V1/N16-1178