Smarten
Augmented Analytics

Clickless Analytics (NLP) Best Practices

Document Version: 1.0
Product Version: 5.3

www.Smarten.com

Document Information	
Document ID	Smarten-NLP-Best-Practices
Document Version	1.0
Product Version	5.3
Date	10-November-2022
Recipient	NA
Author	EMTPL

Statement of Confidentiality, Disclaimer and Copyright

Disclaimer

This document is intended to support administrators, technology managers or developers using and implementing Smarten. The business needs of each organization will vary and this document is expected to provide guidelines and not rules for making any decisions related to Smarten. The overall performance of Smarten depends on many factors, including but not limited to hardware configuration and network throughput.

Contents

1 Process

- Prepare a list of questions (user query in natural English language) for a use case or domain. Interaction with business users will help in deriving the most frequently asked questions they have for the domain.
- Aggregate all user queries, and prepare a list of metadata (measures and dimensions) used in the queries.
- Prepare a dataset that accommodates these dimensions and measures and other related columns.
- Iteratively analyze the dataset and list of queries throughout the process.
- Optimize the dataset and configuration for a clickless analytics NLP engine; this step is expanded below.
- Keep it simple, but make sure that it should be able to answer most of the user queries.

2 Optimize the dataset and configuration for a clickless analytics NLP engine

2.1 Dataset creation and curation

- Column labels should be curated before use:
  - Avoid duplication of words in column labels.
    - ✓ For example, ProductCategory and ProductName can be named as Category and Products).
  - Avoid underscore or any other character in column labels. Camel case is preferred for proper word segregation.
    - ✓ For example, Employee_Name can be modified as EmployeeName.
    - ✓ PRODUCTID can be modified as ProductID or ProductIdentificationNumber.
    - ✓ srno can be modified as SerialNumber.
  - Avoid abbreviations (instead, they can be mentioned in a list of synonyms).
    - ✓ For example, instead of a column name saying COGS, the column name could be CostOfGoods.
  - Remove unnecessary columns.
    - ✓ For example, such columns as serial numbers and identification numbers, or long descriptive data, such as addresses, can be avoided. Or columns that are not likely to be covered in the user queries can be avoided.
    - ✓ If necessary, it is advisable to create a separate data source for NLP to keep it simple and efficient.
  - Mark proper dimensions and measures.

- ✓ For example, a PIN code column may be marked as an integer measure column because of having numeric values; but for this purpose, that needs to be marked as a dimension.

- Cross-check the data types before finalizing.
  - ✓ For example, if a sales column contains data with a unit conversion added, such as 143.56K, such a column will be detected as a dimension; the unit needs to be removed, and the value needs to be converted accordingly—143.56K = 143560.

2.2 Configure Synonyms/Lingo/Business phrases

- Column name synonyms need to be configured in advance.
- Any organization-based lingo or business phrases used instead of dimension terms in a data domain have to be preliminarily mentioned for a better and easier user experience.
  - ✓ For example, a surgeon may be referred to as a doctor.
  - ✓ A product may also be referred to as an item.

- Columns with names that may have a general dictionary meaning can be enabled to be used with Auto synonyms.
  - ✓ For example, on an employee dataset, if we query about a worker, it will be able to automatically search for an employee as a synonym for worker.

- Phonetic Search can be enabled for similar sounding dictionary words that are detected based on a built-in library.
  - ✓ For example, if a user types Amdavad instead of Ahmedabad or Dilli instead of Delhi, the system will still be able to detect it if a phonetic search is enabled on the city column.

2.3 Configure Priority

- In case of similar measures, the order of priority needs to be set in advance. For example, if the data contains SalesCost, ProductionCost, RawMaterialCost, and CostOfGoods, which cost is considered a priority on the keyword **cost** is determined based on the priority defined.

- This priority will help you resolve the queries with high priority definition. For example, if you have two measures—sales quantity and sales amount—you may want to allocate a higher priority to the sales amount.

2.4 Configure Polarity

- A measure column may have different polarity based on the context of the data. This needs to be mentioned in advance. For example, Sales for an organization is usually the higher the better, while Cost is always the lower the better. So we need to mention that before user navigation from NLP access.
  - ✓ For example, Product with the best cost will be a product that has the least cost of manufacturing, while the best performing product will be the highest selling product. So, in this case, cost will have low polarity, while sales will have high

polarity.

- Mention it for all the measurable data columns that may be queried for the performance aspects, KPI, and trend analysis.

2.5 Set default data operation

- Generally, most of the measures will have SUM as the default operation, but certain measures, such as count or ledger balance, need to have different data operation, e.g., COUNT operation for count, MOST RECENT for ledger balance, and AVERAGE for unit price.

2.6 Iterative Cleansing

- In case the output of a query is not as expected, check the explainer for the computation, and update/cleanse the dataset accordingly.

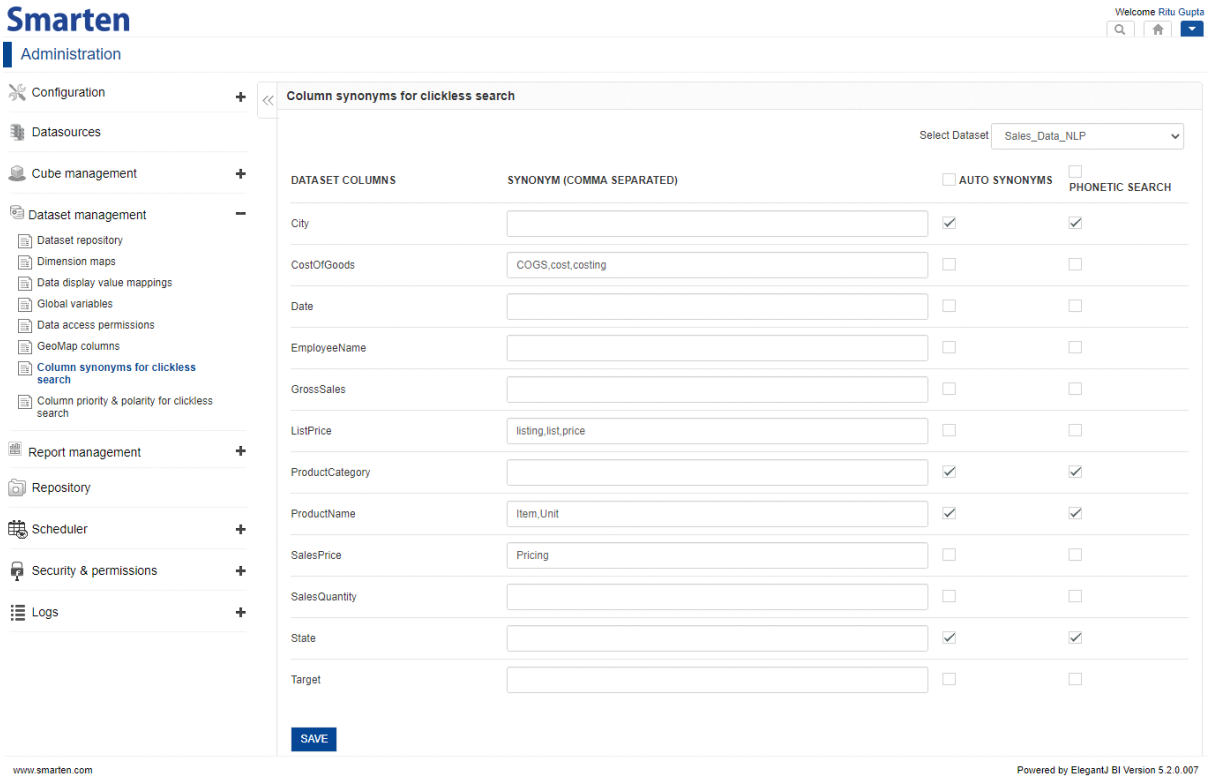
3 Scenario

Let us consider Sales Data with Productwise Gross Sales and Cost across various states and cities.

Below is the list of Dataset Columns for the same:

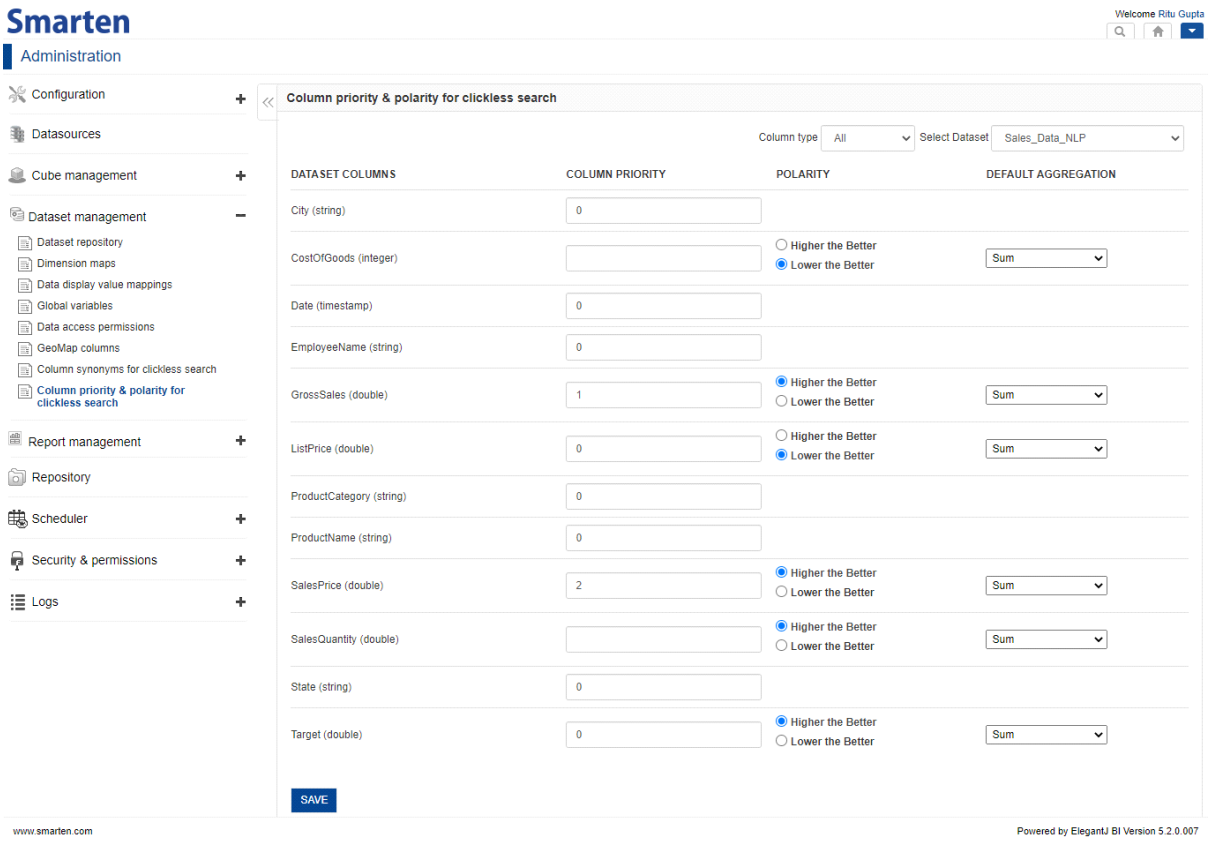
- Date
- ProductCategory
- ProductName
- State
- City
- EmployeeName
- SalesQty
- CostofGoods
- ListPrice
- SalesPrice
- Target
- Emp_ID
- GrossSales

Synonym and Polarity configuration are done as follows:



Here, we have added synonyms for the following columns. These are the terms that can be used interchangeably while referring to these columns in an NLP query.

- CostofGoods—COGS, cost, costing
- ListPrice—listing price, price
- ProductName—Item, Unit
- SalesPrice—Pricing



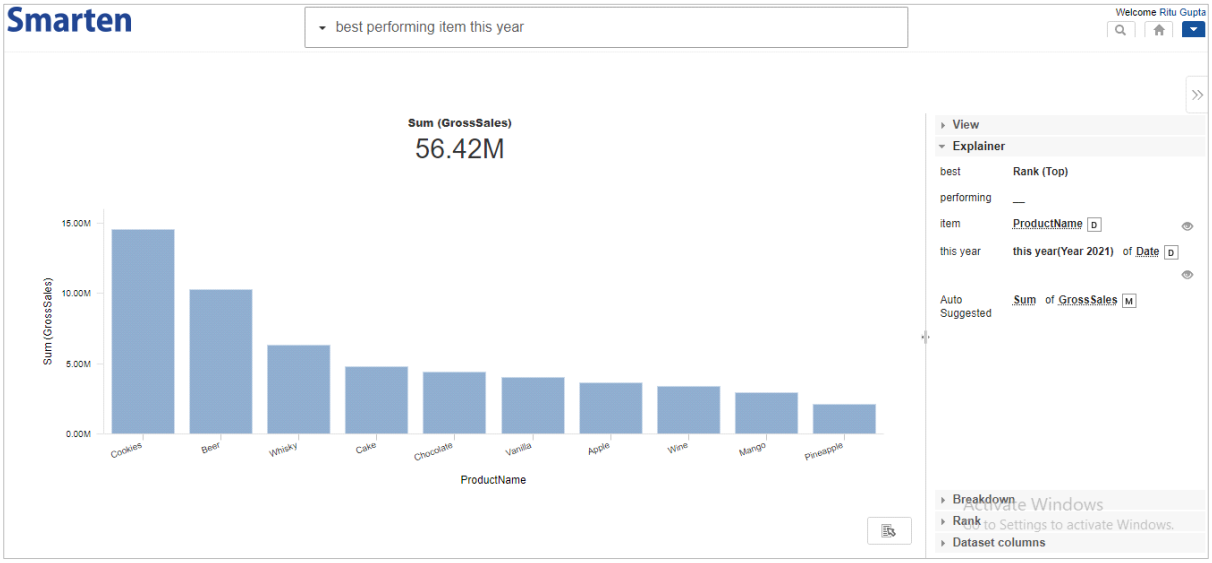
Here, we are setting priority of columns. We can see that GrossSales has been given the highest priority (priority value 1), while SalesPrice has been given a lower priority (priority value 2). This will ensure that when a user queries with only the keyword “Sales” then first priority will be given to the GrossSales column value.

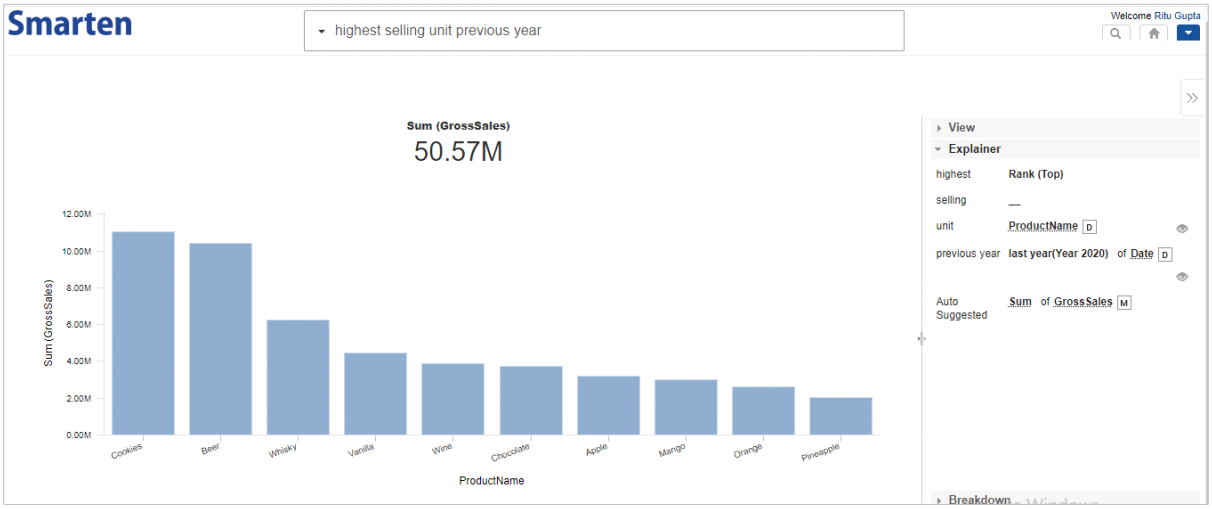
We also need to set polarity as the higher the better/lower the better to define how the measure value should be assessed while querying.

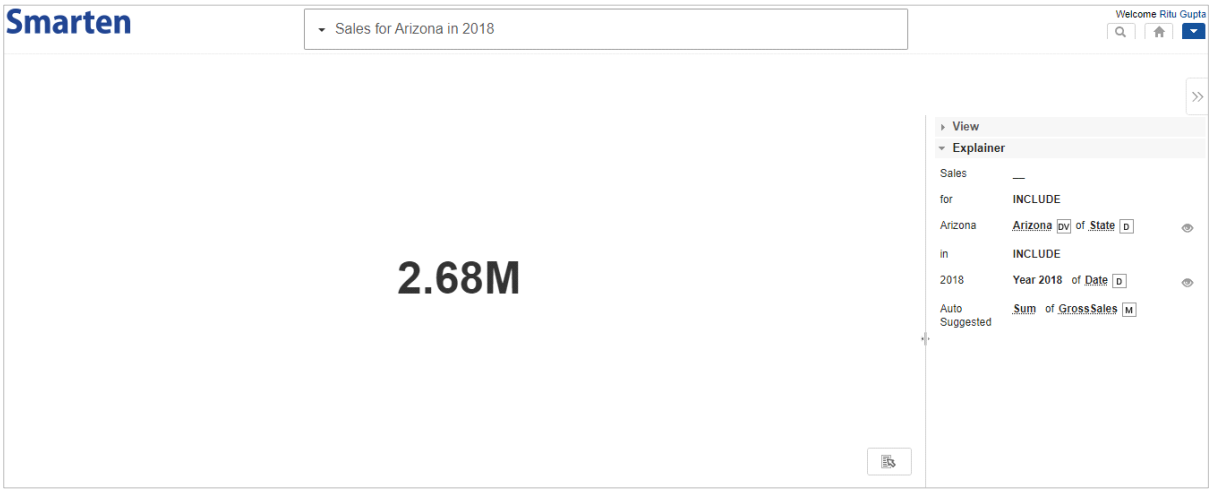
In the above example, when a user queries Best Sales, then the highest sales value should be displayed as per Higher the Better polarity.

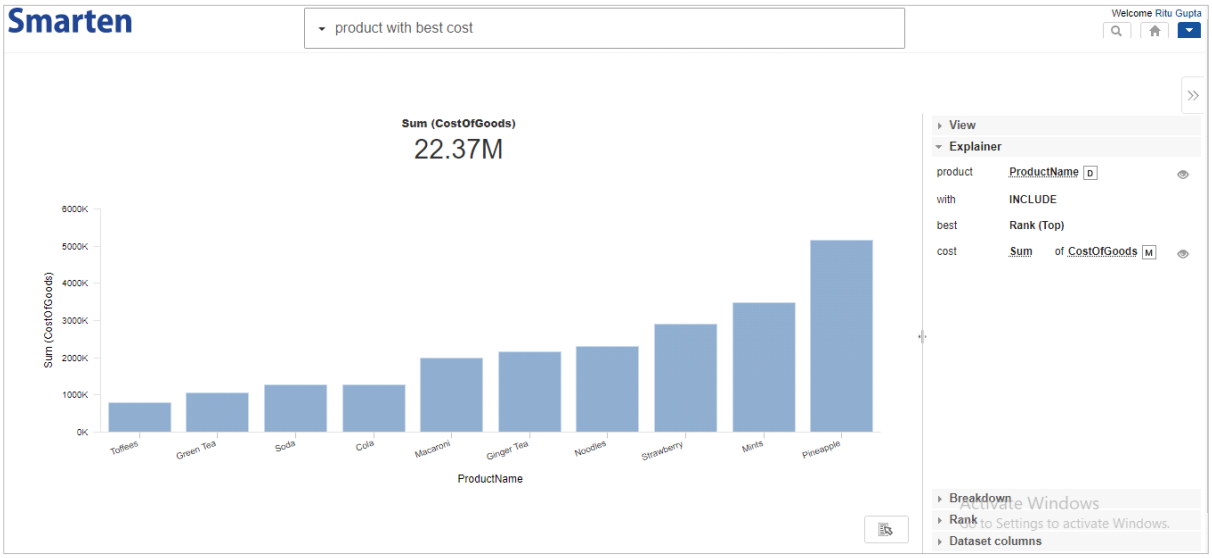
Whereas when the query is for best cost, then the lowest cost will be considered as the best with Lower the Better polarity.

PFB the screenshots that depict how Smarten through its NLP functionality is able to answer the user queries:









4 Product and Support Information

Find more information about Smarten and its features at www.smarten.com
Support: support@smarten.com
Sales: sales@smarten.com
Feedback & Suggestions: support@smarten.com
Support & Knowledgebase Portal: support.smarten.com