**5 WHYS ANALYSIS DASHBOARD**

**ELASTIC SEARCH DATABASE SCHEMA**

**FIELDS:**

* **channel\_name:** keyword
* **app\_name:** keyword
* **severity:** integer
* **issue\_id:** keyword
* **lessons\_learned:** text
* **why1:** text
* **why2:** text
* **why3:** text
* **why4:** text
* **why5:** text
* **category:** keyword
* **issue\_start\_date:** date
* **issue\_end\_date:** date
* **lessons\_learned\_vector:** dense\_vector (dims: 768)
* **why1\_vector:** dense\_vector (dims: 768)
* **why2\_vector:** dense\_vector (dims: 768)
* **why3\_vector:** dense\_vector (dims: 768)
* **why4\_vector:** dense\_vector (dims: 768)
* **why5\_vector:** dense\_vector (dims: 768)

**Explanation of Fields and Indexing Choices**

1. **channel\_name (keyword)**:
   * **Type**: keyword
   * **Reason**: Channels are categorical and will be used for exact matches and aggregations.
2. **app\_name (keyword)**:
   * **Type**: keyword
   * **Reason**: Application names are also categorical. Exact matches and aggregations will be performed based on the app name.
3. **severity (integer)**:
   * **Type**: integer
   * **Reason**: Severity is a numerical value that will be used in range queries and aggregations.
4. **issue\_id (keyword)**:
   * **Type**: keyword
   * **Reason**: Issue IDs are unique identifiers and will be used for exact matches.
5. **lessons\_learned (text)**:
   * **Type**: text
   * **Reason**: This field is editable and contains free-form text, so it needs to be indexed for full-text search to allow efficient updates and searches.
6. **why1, why2, why3, why4, why5 (text)**:
   * **Type**: text
   * **Reason**: These fields contain free-form text for the "5 Whys" analysis and will need to be indexed for full-text search.
7. **category (keyword)**:
   * **Type**: keyword
   * **Reason**: Categories are categorical data used for exact matches and aggregations.
8. **issue\_start\_date, issue\_end\_date (date)**:
   * **Type**: date
   * **Reason**: Dates will be used in range queries to filter incidents within a specific time frame.
9. **lessons\_learned\_vector (dense\_vector)**:
   * **Type**: dense\_vector
   * **dims**: 768
   * **Reason**: To store vector representations for semantic search. Enables searching for semantically similar content.
10. **why1\_vector, why2\_vector, why3\_vector, why4\_vector, why5\_vector (dense\_vector)**:
    * **Type**: dense\_vector
    * **dims**: 768
    * **Reason**: To store vector representations of the "5 Whys" fields for semantic search, allowing for the retrieval of semantically similar analysis.

**Indexing Strategy**

* **Keyword Fields**: Fields like channel\_name, app\_name, category, and issue\_id are set as keyword because we will perform exact match queries and aggregations on these fields. Using keyword ensures that the

field values are not tokenized, maintaining their original format for exact matches.

* **Integer Field**: The severity field is an integer because it allows for numerical operations such as range queries and aggregations. This is important for filtering incidents by severity levels.
* **Text Fields**: The lessons\_learned, why1, why2, why3, why4, and why5 fields are text fields because they need to support full-text search capabilities. This allows users to search within these fields for specific keywords or phrases, facilitating a deeper analysis of the "5 Whys."
* **Date Fields**: The issue\_start\_date and issue\_end\_date fields are set to date type to allow efficient range queries. These fields are crucial for generating reports based on a specified date range.
* **Dense Vector Fields**: The lessons\_learned\_vector and the why1\_vector, why2\_vector, why3\_vector, why4\_vector, why5\_vector fields are dense\_vector fields with 768 dimensions. These fields store vector representations of the corresponding text fields and enable semantic search capabilities.

**Justification for Indexing Choices**

* **Efficiency in Search and Filtering**: By indexing the channel\_name, app\_name, category, and severity fields, we ensure that searches and filters applied on these fields are efficient and quick. This is particularly important for the internal search functionality where users can filter incidents based on these criteria.
* **Aggregations**: Indexing these fields also facilitates aggregations, which are essential for generating reports that show trends and counts of incidents per application or category. Aggregations are a core feature in Elasticsearch and are used extensively for analytics.
* **Full-Text Search**: The text fields (lessons\_learned, why1, why2, why3, why4, why5) need to be indexed for full-text search to support querying on the content of these fields. This is important for analyzing and drawing insights from the qualitative data contained in the "5 Whys" analysis.
* **Date Range Queries**: Indexing the issue\_start\_date and issue\_end\_date as date types ensures that we can perform efficient range queries. This is crucial for the reports page where users filter incidents based on date ranges to analyze trends over time.
* **Semantic Search**: The dense\_vector fields are indexed to store vector representations of the text fields, enabling semantic search. This allows for finding documents that are semantically similar based on their vector representations rather than just keyword matching.

**Node, Cluster, Shard, and Replica Configuration**

**Nodes and Clusters**

1. **Nodes**: A minimum of 3 nodes is recommended to ensure high availability and fault tolerance.
   * **Master Nodes**: 3 master-eligible nodes to manage the cluster state.
   * **Data Nodes**: At least 2 data nodes to store and manage the data.
2. **Cluster**: A single Elasticsearch cluster that includes all the nodes.

**Shards and Replicas**

1. **Shards**:
   * **Primary Shards**: 5 primary shards to distribute the data and manage parallel processing. This helps in balancing the load and improving query performance.
   * **Reason**: The number of primary shards can be based on the expected size of the data and the query load. Each shard can hold a subset of the data, allowing Elasticsearch to parallelize the operations.
2. **Replicas**:
   * **Replica Shards**: 1 replica per primary shard.
   * **Reason**: This provides redundancy and high availability. If a primary shard goes down, the replica shard can serve the requests without any data loss.

**Configuration Summary**

Here's how you can configure the cluster:

1. **Nodes**: 3 nodes for high availability.
2. **Cluster**: 1 cluster.
3. **Shards**: 5 primary shards for scalability.
4. **Replicas**: 1 replica for fault tolerance and improved search performance.

**MAPPING DEFINITION**

{

"settings": {

"index": {

"number\_of\_shards": 5,

"number\_of\_replicas": 1

}

},

"mappings": {

"properties": {

"channel\_name": {

"type": "keyword"

},

"app\_name": {

"type": "keyword"

},

"severity": {

"type": "integer"

},

"issue\_id": {

"type": "keyword"

},

"lessons\_learned": {

"type": "text"

},

"why1": {

"type": "text"

},

"why2": {

"type": "text"

},

"why3": {

"type": "text"

},

"why4": {

"type": "text"

},

"why5": {

"type": "text"

},

"category": {

"type": "keyword"

},

"issue\_start\_date": {

"type": "date"

},

"issue\_end\_date": {

"type": "date"

},

"lessons\_learned\_vector": {

"type": "dense\_vector",

"dims": 768

},

"why1\_vector": {

"type": "dense\_vector",

"dims": 768

},

"why2\_vector": {

"type": "dense\_vector",

"dims": 768

},

"why3\_vector": {

"type": "dense\_vector",

"dims": 768

},

"why4\_vector": {

"type": "dense\_vector",

"dims": 768

},

"why5\_vector": {

"type": "dense\_vector",

"dims": 768

}

}

}

}

When using dense vectors for semantic search in Elasticsearch, the dimension (dims) of the vector depends on the specific NLP model used to generate these vectors. Here are a few examples of different NLP models and how the dims might change:

1. **BERT (Base)**:
   * Dimensions: 768
2. **DistilBERT**:
   * Dimensions: 768
3. **RoBERTa (Base)**:
   * Dimensions: 768
4. **GPT-2**:
   * Dimensions: 768 or 1024 depending on the model variant
5. **Word2Vec**:
   * Dimensions: Typically between 100 and 300, depending on training parameters
6. **GloVe**:
   * Dimensions: Typically between 100 and 300, depending on the specific pre-trained model used
7. **FastText**:
   * Dimensions: Typically between 100 and 300, depending on training parameters

The field definitions for dense vectors would look similar, but the dims parameter would be set according to the specific model used. Here’s how the schema might change for a few different models: