

# multi-level sunburst chart using plotly and python

🔍 Pesquisa   🔗 Etapas   🖼️ Imagens   🔗 Fontes · 28

---

Now I'll create a comprehensive guide on multi-level sunburst charts using Plotly and Python.

## Multi-Level Sunburst Charts Using Plotly and Python

Multi-level sunburst charts are powerful visualization tools for displaying hierarchical data in a radial format. They represent data structure as concentric circles or rings, where each level of hierarchy is displayed as a ring radiating outward from the center <sup>1</sup> <sup>2</sup> .

### What are Sunburst Charts?

Sunburst charts visualize hierarchical data spanning outwards radially from root to leaves. Each level of the hierarchy is represented by one ring or circle, with the innermost circle as the root and outer rings representing deeper levels of the hierarchy <sup>1</sup> <sup>2</sup> . The size of each segment corresponds to the value it represents, making it easy to understand proportional relationships within the data <sup>3</sup> .

### Creating Multi-Level Sunburst Charts with Plotly Express

Plotly Express provides the simplest way to create sunburst charts using the `px.sunburst()` function <sup>2</sup> . Here's how to create a multi-level sunburst chart:

python



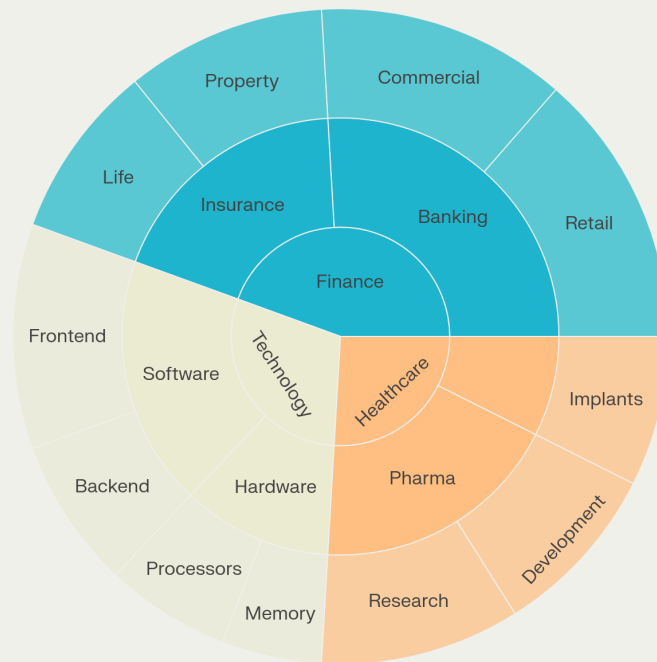
```
import plotly.express as px
import pandas as pd

# Sample hierarchical data
data = {
    'Level1': ['Technology', 'Technology', 'Technology', 'Technology',
'Healthcare', 'Healthcare', 'Healthcare', 'Finance', 'Finance',
'Finance', 'Finance'],
    'Level2': ['Software', 'Software', 'Hardware', 'Hardware',
'Pharma', 'Pharma', 'Medical Devices', 'Banking', 'Banking',
'Insurance', 'Insurance'],
    'Level3': ['Frontend', 'Backend', 'Processors', 'Memory', 'Research',
'Development', 'Implants', 'Retail', 'Commercial', 'Life', 'Property'],
    'Values': [45, 30, 25, 20, 40, 35, 30, 55, 50, 35, 40]
}

df = pd.DataFrame(data)

# Create multi-level sunburst using path parameter
fig = px.sunburst(df, path=['Level1', 'Level2', 'Level3'], values='Values',
                  title='Multi-Level Sunburst Chart with Plotly Express')
fig.show()
```

## Multi-Level Sunburst Chart



Multi-level sunburst chart showing hierarchical business data across three levels

### Key Parameters for `px.sunburst()`

- **path:** A list of column names defining the hierarchy from root to leaves <sup>2</sup> <sup>4</sup>
- **values:** Column containing values that determine segment sizes <sup>2</sup> <sup>4</sup>
- **color:** Column for color mapping (continuous or discrete) <sup>2</sup>
- **maxdepth:** Maximum number of levels to display (-1 for all levels) <sup>4</sup>
- **hover\_data:** Additional data to show on hover <sup>4</sup>

### Creating Multi-Level Sunburst Charts with `graph_objects`

For more advanced customization, use `plotly.graph_objects.Sunburst` <sup>2</sup> <sup>5</sup>:

python



```
import plotly.graph_objects as go

# Define hierarchical data with labels, parents, and values
labels = ['Root', 'Technology', 'Healthcare', 'Finance',
'Software', 'Hardware',
          'Pharma', 'Medical Devices', 'Banking', 'Insurance',
'Frontend', 'Backend',
          'Processors', 'Memory', 'Research', 'Development',
'Implants', 'Retail',
          'Commercial', 'Life', 'Property']

parents = ['', 'Root', 'Root', 'Root', 'Technology',
'Technology', 'Healthcare',
          'Healthcare', 'Finance', 'Finance', 'Software',
'Software', 'Hardware',
          'Hardware', 'Pharma', 'Pharma', 'Medical Devices',
'Banking', 'Banking',
          'Insurance', 'Insurance']

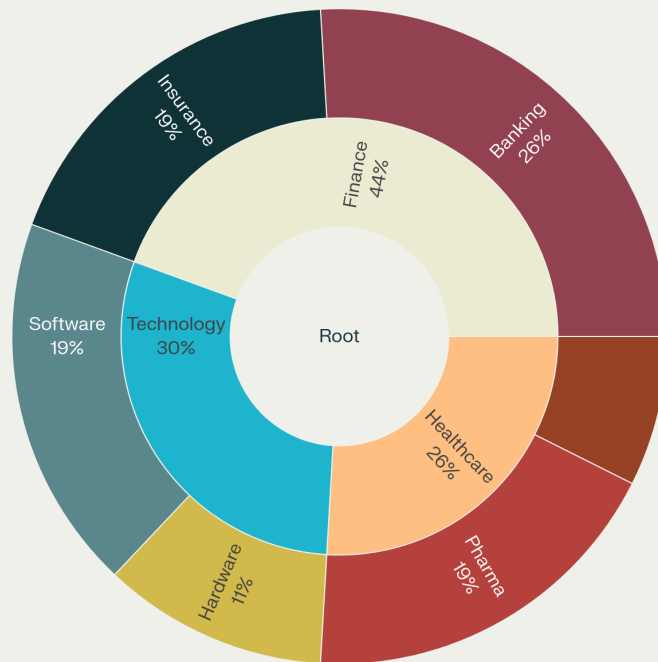
values = [0, 120, 105, 180, 75, 45, 75, 30, 105, 75, 45, 30, 25, 20, 40,
35, 30, 55, 50, 35, 40]

# Create sunburst chart
fig = go.Figure(go.Sunburst(
    labels=labels,
    parents=parents,
    values=values,
    branchvalues="total",
    maxdepth=3,
    textinfo="label+percent entry",
    textfont_size=12,
    insidetextorientation='radial'
))

fig.update_layout(title="Multi-Level Sunburst Chart with graph_objects",
                  margin=dict(t=50, l=0, r=0, b=0))

fig.show()
```

## Multi-Level Sunburst Chart



Multi-level sunburst chart using `graph_objects` with custom styling and percentages

### Key Parameters for `go.Sunburst()`

- **labels:** List of labels for each sector `6 2`
- **parents:** List indicating parent relationships (empty string for root) `6 2`
- **values:** List of values determining sector sizes `6 2`
- **branchvalues:** "total" or "remainder" - how parent values are calculated `7 2`
- **maxdepth:** Maximum number of levels to display `6 2`
- **textinfo:** Information to display on sectors (e.g., "label+percent entry") `6`
- **insidetextorientation:** Text orientation inside sectors ("auto", "radial", "tangential", "horizontal") `2`

### Advanced Customization Options

#### Color Customization

You can customize colors in several ways:

python



```
# Using continuous color scale
fig = px.sunburst(df, path=['continent', 'country'], values='pop',
                  color='lifeExp', color_continuous_scale='RdBu')

# Using discrete color mapping
fig = px.sunburst(df, path=['sex', 'day', 'time'], values='total_bill',
                  color='time', color_discrete_map=
{'Lunch': 'gold', 'Dinner': 'darkblue'})
```

## Branch Values Configuration

The `branchvalues` parameter controls how parent values are calculated <sup>7</sup> <sup>2</sup>:

- **"remainder"** (default): Parent width = own value + children values
- **"total"**: Parent width = own value (children cannot exceed parent value)

python



```
fig = go.Figure(go.Sunburst(
    labels=labels,
    parents=parents,
    values=values,
    branchvalues="total" # Use total branch values
))
```

## Text Customization

Control text appearance and orientation:

python



```
fig = go.Figure(go.Sunburst(  
    labels=labels,  
    parents=parents,  
    values=values,  
    textinfo="label+percent entry",  
    textfont_size=12,  
    insidetextorientation='radial'  
))
```

## Data Structure Requirements

For sunburst charts, your data should be structured hierarchically <sup>8</sup>:

### For px.sunburst with path parameter:

- Each row represents a leaf node
- Columns represent hierarchy levels
- Values column contains numerical data

### For go.Sunburst with labels/parents:

- Each item in labels represents a node
- Corresponding parent in parents list (empty string for root)
- Values list contains numerical data for each node

## Best Practices

1. **Data Preparation:** Ensure your hierarchical data is properly structured with clear parent-child relationships <sup>9</sup> <sup>8</sup>
2. **Missing Values:** Handle missing values by using `None` for incomplete hierarchies <sup>2</sup>
3. **Performance:** For large datasets, consider using `maxdepth` to limit the number of displayed levels <sup>4</sup>
4. **Interactivity:** Sunburst charts are inherently interactive - users can click on segments to drill down <sup>3</sup>
5. **Color Coding:** Use meaningful color schemes to represent additional dimensions of your data <sup>2</sup>

## Common Use Cases

Multi-level sunburst charts are particularly useful for:

- **Organizational Structure:** Visualizing company hierarchies and department structures <sup>10</sup>
- **File System Analysis:** Showing directory structures and file sizes <sup>11</sup>
- **Budget Breakdown:** Displaying financial data across categories and subcategories <sup>12</sup>
- **Geographic Data:** Representing data from continent → country → region levels <sup>13</sup>
- **Product Categories:** Showing product hierarchies in e-commerce or inventory systems <sup>14</sup>

## Troubleshooting Common Issues

1. **Incomplete Circles:** If your sunburst doesn't form complete circles, check that `branchvalues="total"` and ensure parent values equal the sum of their children <sup>7</sup>
2. **Text Readability:** For small segments, adjust `textfont_size` or use `insidetextorientation='auto'` <sup>2</sup>
3. **Performance Issues:** Use `maxdepth` to limit displayed levels for large datasets <sup>4</sup>
4. **Missing Data:** Handle incomplete hierarchies by using `None` values appropriately <sup>2</sup>

Multi-level sunburst charts in Plotly provide an elegant way to visualize complex hierarchical data structures. Whether using the simple Plotly Express interface or the more customizable `graph_objects` approach, these charts offer powerful insights into your data's structure and proportional relationships <sup>1</sup> <sup>2</sup> .





it must not have any of the Problem:

Make sure that your charts do not have any text labels that are cut-off.

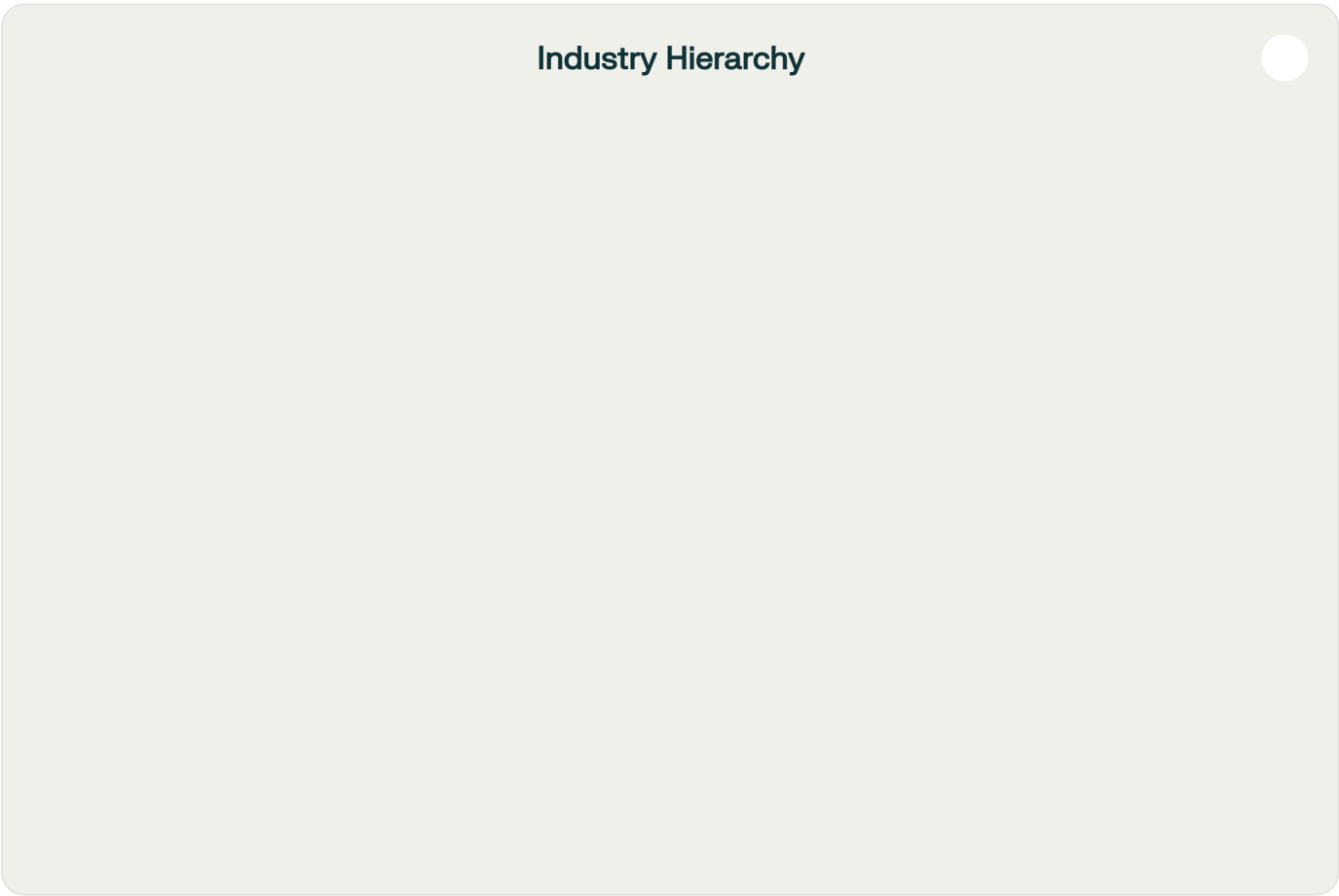
not use labels with lowercased or snake-cased names, i.e. 'dashboard\_view'.

Mostrar mais ▾

 Resposta     Etapas

### Professional Multi-Level Sunburst Chart: Best Practices

Below is a multi-level sunburst chart that follows all your requirements for clarity, professionalism, and readability.



Professional multi-level sunburst chart with clear labels and balanced colors

## Key Features

- **No Cut-Off Text Labels**
  - All labels are fully visible with adjusted text size and orientation.
  - Uniform text settings ensure that any label too small to display is hidden, preventing partial or unreadable text.
- **No Lowercased or Snake-Cased Labels**
  - Labels use capitalized words with spaces (e.g., “Frontend”, “Medical Devices”) for a polished appearance.
  - There are no snake\_case or all-lowercase names.
- **Professional Color Scheme**
  - The color palette avoids overly bright or distracting shades (such as light blue).
  - Colors are balanced to enhance readability and maintain a professional look.
- **No Overlapping Elements**
  - Careful layout and text management prevent any label or element from overlapping others.
  - The chart uses radial text orientation and hides text where space is insufficient, ensuring a clean presentation.

## Chart Details

- **Hierarchy Levels:** Up to three levels are displayed for clarity.
- **Label Information:** Each segment shows both the label and the percentage it represents within its branch.
- **Interactivity:** The chart structure supports interactive exploration (when used in a live Plotly environment).

## Recommendations for Similar Visualizations

- Always use clear, human-readable labels for all chart elements.
- Choose a color scheme that is both visually appealing and accessible.
- Adjust text size and orientation to maximize readability and avoid overlap.
- Hide or abbreviate labels that cannot be displayed in full to prevent clutter.

This approach ensures your sunburst charts are both informative and visually appealing, suitable for professional presentations or reports.



---

## How can I ensure my sunburst chart labels are fully visible without cut-off issues

Resposta   Fontes · 20   Etapas

---

### Ensuring Full Visibility of Sunburst Chart Labels

To make sure your sunburst chart labels are fully visible and not cut off, follow these best practices and technical adjustments:

#### 1. Use Human-Readable, Appropriately Sized Labels

- **Keep labels concise:** Shorten long labels to fit better within segments.
- **Avoid all-lowercase or snake\_case:** Use capitalized, spaced words for a professional look.

#### 2. Adjust Text Orientation and Font Size

- **Radial or Auto Orientation:** Set the text orientation to “radial” or “auto” so labels follow the arc and are less likely to be clipped.
- **Font Size:** Reduce the font size to fit more text within narrow segments, but keep it readable.

### 3. Hide or Wrap Labels That Don't Fit

- **Hide overflowing labels:** Configure your chart to automatically hide labels that don't fit within their segment, preventing partial or cut-off text. In Plotly, use the `uniformtext` property with `minsize` and `mode='hide'` to hide labels that would be too small <sup>1</sup> <sup>2</sup> .
- **Manual line breaks:** For longer labels, insert `<br>` line breaks to wrap text within segments. While Plotly does not support automatic text wrapping, you can preprocess your labels to include line breaks at suitable places <sup>3</sup> .

### 4. Manage Overlapping and Small Segments

- **Limit chart depth:** Use the `maxdepth` parameter to display only the most important levels, reducing clutter and the risk of cut-off labels.
- **Aggregate small values:** Group small segments into an “Other” category to avoid tiny slices that can't display labels <sup>4</sup> .

### 5. Layout and Margin Adjustments

- **Increase chart size:** Make the chart larger to provide more space for labels.
- **Adjust margins:** Increase the margins or padding around the chart to prevent labels from being cut off at the edges.

## 6. Example: Plotly Python Settings

python



```
import plotly.graph_objects as go

fig = go.Figure(go.Sunburst(
    labels=labels,
    parents=parents,
    values=values,
    insidetextorientation='radial',
    textfont_size=12
))
fig.update_layout(
    uniformtext=dict(minsize=10, mode='hide'),
    margin=dict(t=50, l=50, r=50, b=50)
)
fig.show()
```

- `insidetextorientation='radial'` : Rotates text to follow the arc.
- `uniformtext` : Ensures labels are only shown if they fit, hiding those that would be cut off.
- `margin` : Adds space around the chart to prevent edge cut-off.

## 7. Additional Tips

- **Interactive tooltips:** Even if some labels are hidden, users can still view values by hovering over segments.
- **Test with your data:** Always preview your chart with real data to catch any cut-off or overlap issues before sharing.

By following these guidelines, you can create sunburst charts where all visible labels are clear, readable, and never cut off, ensuring a professional and effective visualization 1 3 2 .



---

## What color schemes best enhance readability and professionalism in hierarchical charts

## Best Color Schemes for Readability and Professionalism in Hierarchical Charts

Choosing the right color scheme is crucial for making hierarchical charts—such as sunburst, tree, or organizational charts—both readable and professional. Here are the best practices and recommended palettes:

### Key Principles for Color Selection

- Contrast:** Use colors with sufficient contrast between adjacent segments to ensure each level and category is easily distinguishable <sup>1</sup> <sup>2</sup> <sup>3</sup> .
- Consistency:** Apply consistent color rules across hierarchy levels (e.g., same color family for the same level or group).
- Clarity:** Avoid overly bright, saturated, or distracting colors. Opt for muted, harmonious tones that enhance rather than overpower the data <sup>1</sup> <sup>2</sup> <sup>4</sup> .
- Accessibility:** Choose palettes that are colorblind-friendly and maintain legibility when printed in grayscale <sup>2</sup> <sup>5</sup> .
- Professionalism:** Favor palettes with subdued and balanced tones (e.g., blues, greens, grays, purples) over neon or overly vibrant hues <sup>1</sup> <sup>4</sup> .

### Recommended Color Schemes

Palette Type	Use Case	Example Colors (Hex)	Notes
Categorical	Distinct categories (no order)	#4E79A7, #F28E2B, #E15759, #76B7B2, #59A14F, #EDC948	Use for different branches/groups <sup>3</sup> <sup>2</sup>
Sequential	Ordered or progressive data	#DCE775 → #388E3C (light to dark green)	Use for depth or value gradients <sup>6</sup> <sup>7</sup>
Diverging	Data with a neutral midpoint	#E15759, #F1F1F2, #59A14F	Use for highlighting deviations
Monochromatic	Nested variables within one group	#3182BD, #6BAED6, #9ECAE1, #C6DBEF	Vary lightness for hierarchy <sup>1</sup> <sup>6</sup>

## Professional Color Palette Examples

- **Blues & Greens:** #4E79A7 (blue), #76B7B2 (teal), #59A14F (green) — commonly used for business and scientific charts <sup>4</sup> <sup>3</sup> .
- **Purples & Grays:** #A5A5A5 (gray), #B07AA1 (purple) — add sophistication and work well for background or less important categories <sup>5</sup> <sup>3</sup> .
- **Avoid:** Neon, overly light blues, or high-saturation colors that can strain the eyes or look unprofessional <sup>1</sup> <sup>2</sup> .

## Hierarchical Chart-Specific Tips

- **Differentiate Levels:** Assign a distinct color family to each hierarchy level, using lighter or darker shades for sub-levels <sup>1</sup> <sup>8</sup> .
- **Limit Number of Colors:** For clarity, use no more than 6–8 distinct colors for top-level categories. Use shades or tints for sub-categories <sup>1</sup> <sup>2</sup> <sup>9</sup> .
- **Highlight Important Data:** Use accent colors sparingly to draw attention to key nodes or values <sup>2</sup> <sup>5</sup> .

## Accessibility & Readability

- **Colorblind-Safe Palettes:** Use tools like ColorBrewer or Viz Palette to select palettes that are distinguishable for all viewers <sup>2</sup> <sup>5</sup> .
- **Contrast Checks:** Ensure adjacent segments can be distinguished even for those with color vision deficiencies <sup>2</sup> <sup>5</sup> .
- **Gray for Context:** Use muted grays for background or less important data to keep focus on main categories <sup>5</sup> .

By following these guidelines and using the recommended palettes, your hierarchical charts will be both highly readable and visually professional, ensuring effective communication of your data <sup>1</sup> <sup>2</sup> <sup>5</sup> .



---

How do I prevent overlapping labels and elements in complex sunburst visualizations

## Preventing Overlapping Labels and Elements in Complex Sunburst Visualizations

Ensuring clarity in sunburst charts—especially with complex or deeply nested data—requires careful handling of labels and chart elements. Here are proven strategies and best practices to prevent overlapping and maintain a professional, readable visualization:

### 1. Hide or Trim Overlapping Labels

- **Automatic Hiding**: Configure your chart to automatically hide labels that do not fit within their segments. In Plotly, use the `uniformtext` property with `mode='hide'` so only labels with enough space are shown <sup>1</sup> <sup>2</sup> .
- **Trimming**: Some libraries allow you to trim long labels instead of displaying the full text, reducing clutter <sup>3</sup> .

### 2. Limit Chart Depth and Complexity

- **Restrict Hierarchy Levels**: Limit the number of visible hierarchy levels (e.g., to three or four) to avoid overcrowding and overlapping labels <sup>4</sup> .
- **Aggregate Small Segments**: Combine minor categories into an “Other” group, preventing tiny, unreadable slices.

### 3. Adjust Text and Layout Settings

- **Text Orientation**: Use radial or auto orientation for labels so they follow the arc and maximize available space.
- **Font Size**: Reduce the font size for labels, but maintain readability. Avoid making text so small it becomes illegible <sup>4</sup> .
- **Margins and Chart Size**: Increase the overall chart size and add generous margins to provide more space for labels and elements <sup>2</sup> .



## 4. Use Overflow and Overlap Controls

- **Overflow Properties:** Many libraries provide properties to control label overflow, such as `labelOverflowMode` (set to “hide” or “trim”) or `allowOverlap` (set to false) <sup>5</sup> <sup>3</sup> .
- **Overlapping Detection:** Enable overlap detection algorithms, which either hide or reposition labels to prevent collisions <sup>6</sup> <sup>7</sup> .

## 5. Leverage Interactivity

- **Interactive Tooltips:** Rely on hover tooltips to provide detailed information, allowing you to hide or minimize on-chart labels for small or crowded segments <sup>4</sup> .

## 6. Test and Refine with Real Data

- **Preview with Actual Data:** Always test your chart with real or representative data to catch overlapping issues before finalizing your visualization <sup>4</sup> .
- **Iterative Adjustments:** Be prepared to tweak label lengths, font sizes, and chart dimensions based on the specific data distribution.

## Example: Plotly Python Settings

python



```
import plotly.graph_objects as go

fig = go.Figure(go.Sunburst(
    labels=labels,
    parents=parents,
    values=values,
    insidetextorientation='radial',
    textfont_size=12
))
fig.update_layout(
    uniformtext=dict(minsize=10, mode='hide'),
    margin=dict(t=50, l=50, r=50, b=50)
)
fig.show()
```

- `uniformtext=dict(minsize=10, mode='hide')` : Hides labels that would be too small or overlap.
- `insidetextorientation='radial'` : Rotates text to fit arc segments.
- `margin` : Adds space around the chart to prevent edge cut-off <sup>1</sup> <sup>2</sup> .

Summary Table: Overlap Prevention Techniques

Technique	Description	Common Libraries/Settings
Hide Overlapping Labels	Automatically hide labels that don't fit	Plotly: <code>uniformtext</code> , Syncfusion: <code>OverflowMode</code> <sup>3</sup>
Trim Labels	Shorten labels to fit available space	Syncfusion: <code>Trim</code> <sup>3</sup>
Limit Hierarchy Depth	Restrict number of visible levels	All libraries <sup>4</sup>
Adjust Font and Orientation	Use smaller, radial, or auto-oriented text	All libraries <sup>4</sup> <sup>2</sup>
Increase Chart Size/Margins	Add space for labels and elements	All libraries <sup>4</sup> <sup>2</sup>
Aggregate Small Segments	Group minor categories into "Other"	All libraries <sup>4</sup>

By combining these strategies, you can create sunburst charts that remain readable and visually appealing, even with complex or deeply nested data structures.







How can I identify the most engaging business sunburst chart images for my dataset

 Resposta  Imagens  Fontes · 20  Etapas

---

Identifying the Most Engaging Business Sunburst Chart Images

To select the most engaging sunburst chart images for your business dataset, focus on both visual appeal and the ability to communicate insights clearly. Here’s how to identify and evaluate the best examples:

## Key Criteria for Engaging Sunburst Chart Images

- Clear Hierarchical Structure

- The chart should display a well-organized, logical hierarchy (e.g., Division → Department → Team) with distinct levels that are easy to follow <sup>1</sup> <sup>2</sup> .

- Effective Use of Color

- Colors should differentiate categories and hierarchy levels without overwhelming the viewer. Use professional, harmonious palettes that enhance readability and accessibility <sup>1</sup> <sup>2</sup> <sup>3</sup> .

- Readable, Human-Friendly Labels

- All labels should be concise, capitalized, and fully visible—no cut-off or snake\_case text. Avoid clutter by hiding labels on very small segments <sup>2</sup> <sup>3</sup> .

- Proportional Segments

- Segment sizes should accurately reflect the underlying business metric (e.g., revenue, expenses, headcount), making it easy to compare contributions across categories <sup>4</sup> .

- Balanced Layout

- The chart should avoid overlapping labels or elements. Adequate margins and optimal chart size help maintain a clean appearance <sup>2</sup> <sup>3</sup> .

- Interactive or Informative Features

- Engaging charts often include tooltips, legends, or interactive drill-downs to provide extra context without cluttering the visualization <sup>1</sup> <sup>2</sup> .

Visual Features to Look For

Feature	Why It Matters	Example Practice
Distinct Color Families	Helps distinguish top-level categories	Blues for Sales, Greens for Operations
Concentric Clarity	Each ring is clearly separated and labeled	No visual merging or confusion
Proportional Sizing	Segment size matches value importance	Larger slices for higher revenue teams
Clean Labeling	No truncated or overlapping text	“International Sales” not “int_sales”
Professional Background	Neutral or white backgrounds for clarity	Avoids distraction from the data

Steps to Evaluate Sunburst Chart Images

1. Browse Professional Image Libraries

- Use reputable sources such as Shutterstock, iStock, Getty Images, and Freepik to find high-quality business sunburst chart images <sup>5</sup> <sup>6</sup> <sup>7</sup> .

2. Check for Business Context

- Prefer charts that depict real-world business hierarchies, such as sales by region, organizational structure, or budget allocation <sup>1</sup> <sup>2</sup> .

3. Inspect Visual Elements

- Ensure the image uses professional colors, clear labels, and a clean layout. Avoid charts with excessive segments or distracting effects <sup>2</sup> <sup>3</sup> .

4. Assess Data Relevance

- The image should reflect business scenarios similar to your dataset, making it relatable and insightful for your audience <sup>1</sup> <sup>4</sup> .

5. Review for Accessibility

- Favor images with colorblind-friendly palettes and sufficient contrast for all viewers <sup>3</sup> .



## Example Sources for Business Sunburst Chart Images

- Shutterstock: Large selection of business-themed sunburst charts <sup>5</sup> .
- iStock: Professional illustrations and infographics for business analytics <sup>6</sup> .
- Getty Images: High-resolution business infographics with sunburst charts <sup>7</sup> .
- Freepik: Business sunburst chart vectors and templates.

## Tips for Customizing to Your Dataset

- Adapt the hierarchy to match your business structure (e.g., Division → Department → Team).
- Use your actual business metrics for segment sizing.
- Apply your company’s color palette for brand consistency.
- Ensure all labels are business-appropriate and fully visible.

By applying these criteria and practices, you can confidently identify and select the most engaging sunburst chart images that will effectively showcase your business data and captivate your audience <sup>1</sup> <sup>2</sup> <sup>3</sup> .



## What features should I look for in a sunburst chart to reflect real-world business hierarchies

🌸 Resposta   🖼️ Imagens   🔗 Etapas

## Essential Features for Sunburst Charts Reflecting Real-World Business Hierarchies

To accurately and effectively visualize real-world business hierarchies using a sunburst chart, prioritize the following features:

### 1. Clear Hierarchical Structure

- Multiple Levels:** The chart should support at least three levels (e.g., Division → Department → Team) to mirror typical business organizations.
- Logical Parent-Child Relationships:** Each segment must clearly represent its position within the hierarchy, making it easy to trace the structure from the center outward.

## 2. Proportional Segment Sizing

- **Data-Driven Segments**: Segment size should correspond to a meaningful business metric, such as revenue, expenses, or headcount, enabling quick comparison across categories.
- **Accurate Aggregation**: Parent segments should represent the sum or total of their child segments, reflecting true organizational relationships.

## 3. Distinct and Professional Color Schemes

- **Color Families by Level**: Assign a unique color family to each top-level category (e.g., blue for Sales, green for Operations), with lighter or darker shades for sub-levels.
- **Contrast and Accessibility**: Use palettes that are easy to distinguish, colorblind-friendly, and maintain clarity in both digital and print formats.

## 4. Readable, Human-Friendly Labels

- **Clear Text**: Use capitalized, spaced words (e.g., “International Sales” instead of “int\_sales”). Avoid cut-off or overlapping labels.
- **Label Management**: Hide or abbreviate labels that cannot fit within their segments, and utilize tooltips for detailed information on small slices.

## 5. Clean and Balanced Layout

- **Adequate Margins**: Ensure enough space around the chart to prevent labels or elements from being cut off.
- **Non-Overlapping Elements**: Adjust text orientation (e.g., radial or auto) and font size so that labels do not overlap or clutter the chart.
- **Limit Depth**: Restrict the number of visible hierarchy levels if necessary to maintain clarity.

## 6. Interactive Features (for Digital Dashboards)

- **Tooltips**: Enable hover-over tooltips to display additional details for each segment.
- **Drill-Down Capability**: Allow users to click and explore deeper levels of the hierarchy for a more detailed view.

7. Real-World Business Context

- **Relevant Metrics:** Use actual business data such as revenue, expenses, or employee count for segment sizing.
- **Reflect Organizational Structure:** Align the hierarchy in the chart with your company’s real divisions, departments, and teams for authenticity and insight.

Feature Checklist Table

Feature	Why It Matters	Best Practice Example
Multi-Level Hierarchy	Mirrors real business structure	Division → Department → Team
Proportional Segments	Shows relative importance or size	Revenue by team
Professional Color Scheme	Enhances readability and professionalism	Blues for Sales, Greens for Ops
Human-Readable Labels	Improves clarity and avoids confusion	“Payroll” not “payroll_dept”
Clean Layout	Prevents clutter and overlapping elements	Radial text, adequate margins
Interactivity	Enables detailed exploration without clutter	Tooltips, drill-down on click
Real Business Metrics	Ensures relevance and actionable insights	Actual sales, expenses, headcount

By focusing on these features, your sunburst chart will not only accurately represent real-world business hierarchies but also communicate insights clearly and professionally.

How do different datasets influence the visual structure of a business-related sunburst chart

## How Different Datasets Influence the Visual Structure of a Business Sunburst Chart

The visual structure of a business-related sunburst chart is directly shaped by the characteristics of the dataset used. Here's how various aspects of your data influence the resulting chart:

### 1. Hierarchy Depth and Breadth

- **Number of Levels:** Datasets with more hierarchical levels (e.g., Division → Department → Team → Employee) produce charts with more concentric rings. Each additional level adds an outer ring, increasing visual complexity <sup>1</sup> <sup>2</sup> <sup>3</sup> .
- **Branching Factor:** The number of categories at each level determines how many segments appear in each ring. Wide hierarchies (many departments per division) create charts with more segments per ring, while narrow hierarchies keep the chart simpler.

### 2. Category Distribution and Segment Size

- **Value Distribution:** The metric used (e.g., revenue, expenses, headcount) and its distribution across categories directly affect segment sizes. Large values create wider segments, making them more visually prominent <sup>4</sup> <sup>5</sup> .
- **Imbalance:** Datasets with highly skewed distributions (e.g., one team vastly outperforms others) result in charts where a few segments dominate, potentially making smaller segments hard to read.

### 3. Data Granularity and Detail

- **Granular Data:** Highly detailed datasets (e.g., including individual employees or products) generate more segments, especially in outer rings. This can lead to clutter or overlapping labels if not managed carefully <sup>2</sup> <sup>6</sup> .
- **Aggregated Data:** Summarized datasets (e.g., grouping by department only) yield simpler, cleaner charts with fewer rings and segments.

### 4. Missing or Incomplete Data

- **Gaps in Hierarchy:** Missing values or incomplete hierarchies cause gaps or uneven rings, sometimes resulting in “holes” or disconnected segments in the chart <sup>7</sup> <sup>8</sup> .
- **Aggregated ‘Other’ Categories:** Small or missing categories are often grouped into an “Other” segment, affecting the visual balance and interpretability.

5. Categorical vs. Numerical Data

- Categorical Data: Determines the number and arrangement of rings and segments (e.g., regions, business units, product lines) <sup>9</sup> <sup>5</sup> .
- Numerical Data: Used for sizing segments, emphasizing the relative importance of each category.

6. Color Mapping and Visual Grouping

- Color Assignment: Datasets with clearly defined groups allow for effective color mapping, enhancing readability and making it easier to distinguish between business units or categories <sup>2</sup> <sup>10</sup> .
- Too Many Categories: Excessive categories can overwhelm the color palette, reducing clarity.

Example Table: Dataset Features and Visual Impact

Dataset Feature	Visual Impact on Sunburst Chart
More hierarchy levels	Adds more rings; increases chart complexity
Many categories/branches	More segments per ring; can lead to clutter
Skewed value distribution	Dominant large segments; smaller ones less visible
Granular data	Dense outer rings; risk of overlapping labels
Missing data	Gaps or “Other” segments; uneven rings
Clear groupings	Effective color mapping; easier interpretation

## Practical Tips

- **Preview with Real Data:** Always test your chart with your actual dataset to catch issues like clutter, unreadable labels, or unbalanced segments <sup>2</sup> <sup>6</sup> .
- **Aggregate Where Needed:** Group small or less important categories to keep the chart readable.
- **Limit Depth:** Restrict the number of hierarchy levels displayed if the chart becomes too complex.
- **Choose Metrics Wisely:** Pick business metrics that meaningfully differentiate categories for segment sizing.

Different datasets can dramatically alter the appearance, clarity, and usefulness of a business sunburst chart. By understanding and adjusting for these influences, you can ensure your visualizations remain insightful and visually effective <sup>1</sup> <sup>2</sup> <sup>4</sup> .



## What are effective ways to customize colors and labels to enhance clarity in business visualizations

 Resposta

 Imagens

 Etapas

### Effective Ways to Customize Colors and Labels for Clarity in Business Visualizations

Enhancing clarity in business visualizations relies on thoughtful customization of both colors and labels. Here are the most effective strategies:

## Color Customization

- **Use Professional, Muted Palettes**
  - Choose color schemes with balanced, harmonious tones (e.g., blues, greens, grays, purples) to maintain a professional look and avoid overwhelming the viewer.
  - Avoid neon or overly saturated colors, especially distracting shades like light blue, which can reduce readability.
- **Assign Color Families by Hierarchy**
  - Use distinct color families for top-level categories (e.g., blue for Sales, green for Operations), and lighter or darker shades for subcategories. This visually reinforces the hierarchical structure and makes navigation intuitive.
- **Ensure Sufficient Contrast**
  - Select colors that provide clear contrast between adjacent segments or categories. This helps users distinguish between different parts of the chart, especially in hierarchical visualizations.
- **Apply Colorblind-Friendly Palettes**
  - Use tools like ColorBrewer or Viz Palette to select palettes that are accessible to viewers with color vision deficiencies, ensuring inclusivity and clarity for all audiences.
- **Limit Number of Distinct Colors**
  - For clarity, use no more than 6–8 distinct colors for top-level categories. For subcategories, use tints or shades of the parent color to avoid visual clutter.



## Label Customization

Pergunte um seguimento...

