**Information Visualization**

**Task Abstraction**

*Tamara Munzner*

Department of Computer Science
University of British Columbia

**Nested model: Four levels of visualization design**

- domain situation
- task abstraction
- interaction idiom
- algorithm

**Domain characterization**

- details of an application domain
- group of users, target domain, their questions, & their data
  - varies widely by domain
  - must be specific enough to get traction
- domain questions/problems
  - break down into simpler abstract tasks

**Design Process**

- Characterize Domain Situation
- Map Domain-Language Task to Abstract Task
  - Identify/Create Suitable Idiom/Technique
  - Identify/Create Suitable Algorithm

**Example: Find good movies**

- **Identify good movies in genres I like**
  - domain: general population, movie enthusiasts
- **Tasks**
  - highly rated by critics
  - highly rated by audiences
  - successful at the box office
  - similar to movies I liked
  - matches specific genre
- **Data**
  - IMDb, Rotten Tomatoes

**Example: Horrorified**

- **Identify high-score horror movies**
  - domain: general population, movie enthusiasts
- **Tasks**
  - highly rated by critics
  - highly rated by audiences
- **Data**
  - IMDb, Rotten Tomatoes

**Example: Find good movies**

- **Identify good movies in genres I like**
  - domain: general population, movie enthusiasts
- **Tasks**
  - highly rated by critics
  - highly rated by audiences
- **Data**
  - IMDb, Rotten Tomatoes

**Why: Task Abstraction**

- means and ends
- how
- what
- why

**Analysis example: Derive one attribute**

- smaller number
  - centrality metric for networks
  - derived quantitive attributes
  - derived qualitative attributes

- larger number
  - connectivity metric for networks
  - derived qualitative attributes
  - derived quantitative attributes

**Tasks: Actions and targets**

- very high-level pattern
  - zoom-in/out
  - hover
  - refine

**Actions: Analyze**

- consume
  - discover vs present
  - classic split
- enjoy
  - skrope vs explain
- derive
  - present
  - produce
  - analyze
  - search
  - query

**Tasks: Actions and targets**

- what does user know?
  - target, location
  - lookup
    - ex: word in dictionary
    - alphabetical order
  - locate
    - ex: keys in your house
    - ex: node in network
- browse
  - ex: books in bookstore
- explore
  - ex: cool neighborhood in new city

**Actions: Search**

- what does user know?
  - target, location
  - lookup
    - ex: word in dictionary
    - alphabetical order
  - locate
    - ex: keys in your house
    - ex: node in network
- browse
  - ex: books in bookstore
- explore
  - ex: cool neighborhood in new city
### Abstraction

- These (action, target) pairs are good starting points for vocabulary
- But sometimes you’ll need more precision
- Rule of dumb: systemically remove all domain jargon
- Interplay: task and data abstraction
  - Need to use data abstraction within task abstraction
  - To specify your target
  - But task abstraction can lead you to transform the data
  - Iterate back and forth
  - First pass data, first pass task, second pass data...

### Examples: Job market

- Trends: How did job market develop since recession overall?
- Outliers: Real estate related jobs

### Exercise: Rating Charts for Tasks

- **Task A:** sort attributes
- **Task B:** compare pair of attributes (Direct vs Distributor)
- **Task C:** compare pair of attributes (Distributor vs OEM)
- **Task D:** present trends across all attributes
- **Task E:** spot outlier attributes
- **Task F:** enjoy / engage

### Example: Genomics II

- **Task**
  - Goal: control data quality for gene expression data
  - Tasks: Judge magnitude of sample
  - Compare samples, identify within-group variance & outliers
  - Compare groups, identify between-group variance

### Example: Economics

- **Task**
  - Compare two pairs of attributes (Direct vs Distributor)
  - Analyze: search, query & match

### Example: Horrid vs stacked bars

- Horrid: browse/explore
- Stacked bars: location/lookup

### Exercise: Task abstraction in genomics I

- **Derive**
  - Only some samples show the desired effect
  - Derive two groups of samples
  - The difference between the samples is caused by differential expression (different activity) of genes in a particular pathway
  - She would like to understand which genes are likely to cause the difference
  - Identify those genes
  - Compare gene expression of pathway genes between two groups
  - Identify the outliers
  - Explore the annotation

### Exercise: Task abstraction in genomics I

- **Locate**
  - The difference between the samples is caused by differential expression (different activity) of genes in a particular pathway
  - She would like to understand which genes are likely to cause the difference
  - Identify those genes
  - Compare gene expression of pathway genes between two groups
  - Identify the outliers
  - Explore the annotation

### Exercise: Task abstraction in genomics I

- **Locate**
  - The difference between the samples is caused by differential expression (different activity) of genes in a particular pathway
  - She would like to understand which genes are likely to cause the difference
  - Identify those genes
  - Compare gene expression of pathway genes between two groups
  - Identify the outliers
  - Explore the annotation
Todo this week
• D3 videos to watch this week
  – Making a Bar Chart with D3 and SVG [30 min]
• Quiz 2 to do this week, due by Fri Jan 17, 8am
• Labs start this week!
  – Fri 9-10, 11-12, 4-5
  – strongly recommended but optional; we do not track attendance
• TA office hours for individual consultation and help
  – TAs will typically alternate weeks
  – if you can’t register, try attending the one you want
• Foundations Exercise 2 out next time (Thu Jan 16)
  – due Wed Jan 22
• Programming Exercise 1 out next time (Thu Jan 16)
  – due Wed Jan 29

Credits
• Visualization Analysis and Design (Ch 3)
• Alex Lex & Miriah Meyer, http://dataviscourse.net/
• Marti Hearst, exercise (tasks & charts)
  – Teaching as Coaching (VIS 2015 panel on Vis, The Next Generation)