



Evaluation of Speech Detection Algorithm



Project 1b

Due October 11



Overview

- Experiments to evaluate performance of your Speech Detection project (Project 1)
- Focus not only on how the algorithm performs, but also
 - the formulation of hypotheses
 - design, implementation and analysis of experiments to test the hypotheses
 - writeup



Measures of Performance

- *User perception.* Some possibilities are:
 - User opinion (rating) on quality
 - Understandability
 - Errors in listening ...
- *System impact.* Some possibilities are:
 - CPU load
 - Size (in bytes) of sound recorded (without silence)
 - Processing time
 - Memory use...
- Decide on how each is to be measured
 - Example: *Scale 1-10* for perception
 - Example: *Time* for CPU



Independent Variables

- At least two. Possibilities:
 - Speaking tests: counting, vocabulary, ...
 - Other languages: Hindi, Chinese, Pig-Latin, ...
 - Personal characteristics: Gender, Age, Shoe size ...
 - Background noise: quiet, noisy, Patriot's game, ...
 - Other systems: Win, Linux, Dell, Compaq, Amiga...
 - Hardware: cheap microphone, sound card
 - Other audio quality parameters: rate, size, ...
 - ...


Algorithm Modifications

- Must choose at least 1
- Possibilities include:
 - Thresholds.
 - Sound chunk size.
 - Endpoint detection length.
 - Other modifications specific to your implementation.
 - ...
- Formulate hypotheses
 - About how a change in the independent variables affects your measures of performance


Results and Analysis

- Details on results and analysis
- Results are numeric measures
 - graphs, charts or tables
- Analysis manipulates data
 - understand relationships
 - interpreting the results
- Consider if data supports or rejects the hypotheses



Report

- Introduction
 - hypotheses and motivation for them
 - (not on silence detection, in general)
- Background on your algorithm
- Design of your experiments
 - details on all of above
- Analysis
- Conclusions
 - summarize findings
- Abstract
 - 1 paragraph that abstracts whole report
 - Write last, goes first

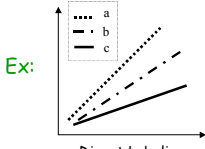
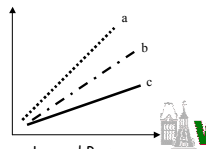



Guidelines for Good Graphs (1 of 5)

“Art” not “rules”. Learn with experience. Recognize good/bad when see it. Many trials *Require minimum effort from reader*

- Perhaps *most* important metric
- Given two, can pick one that takes less reader effort


Ex:



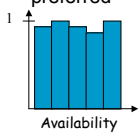

Guidelines for Good Graphs (2 of 5)

- *Maximize information*
 - Make self-sufficient
 - Key words in place of symbols
 - Ex: “PIII, 850 MHz” and not “System A”
 - Ex: “Daily CPU Usage” not “CPU Usage”
 - Axis labels as informative as possible
 - Ex: “Response Time in seconds” not “Response Time”
 - Can help by using captions, too
 - Ex: “Transaction response time in seconds versus offered load in transactions per second.”




Guidelines for Good Graphs (3 of 5)

- *Minimize ink*
 - Maximize information-to-ink ratio
 - Too much unnecessary ink makes chart cluttered, hard to read
 - Ex: no gridlines unless needed to help read
 - Chart that gives easier-to-read for same data is preferred





- Same data
- Unavail = 1 - avail
- Right better




Guidelines for Good Graphs (4 of 5)

- *Use commonly accepted practices*
 - Present what people expect
 - Ex: origin at (0,0)
 - Ex: independent (cause) on x-axis, dependent (effect) on y-axis
 - Ex: x-axis scale is linear
 - Ex: increase left to right, bottom to top
 - Ex: scale divisions equal
- Departures are permitted, but require extra effort from reader so use sparingly



Guidelines for Good Graphs (5 of 5)

- *Avoid ambiguity*
 - Show coordinate axes
 - Show origin
 - Identify individual curves and bars
 - Do not plot multiple variables on same chart



Hand In

- Online turnin (see Web page)
 - Send group info
- Turn in:
 - Any testing Code/Scripts used/modified
 - Makefile/Project file
- Hardcopy

