# Content Selection: Graphs, Supervision, HMMs

LING 573
Systems & Applications
April 5, 2018



UNIVERSITY OF WASHINGTON

PROFESSIONAL MASTER'S IN COMPUTATIONAL LINGUISTICS

# Announcements

- Thanks for the TA survey, but David helpfully reminded me that 2 hours are req'd
  - …so everyone gets their first choice!
  - David's office hours are posted on the Canvas syllabus homepage.

*Begin Recording!*

# Clarification of the Task Data

- I have created a README.md in the **`dropbox/17-18/573`** directory
  - Should clarify WTH is happening inside the various folders.
- Data in **`dropbox/17-18/573/Data`**
  - From TAC 2009 AESOP Track (documentation here)
- This task:
  - Guided (Document sets/clusters are provided)
  - Evaluated against humans (**`./models`**)
  - …and other automatic systems (**`./peers`**)

# Clarification of the Task Data

- You will note, within the **`/models`** directory, there are:
  - training / devtest / evaltest
- …why training?
  - We will get to that in today's lecture.
  - (TL;DL – Supervision)

# Clarification of the Task Data

- **Input**:
  - Topic Categories:
    1. Weather/Natural Events/Disasters
    2. Violence/Uprisings/Terror
    3. Disease/Disorders/Health
    4. Wildlife
    5. Legal Cases
  - Topics
  - Document Sets (two sets per topic, ten documents per set)

# Clarification of the Task Data

- **Evaluation Data:**
  - Comparable extractive summaries provided by
    - humans
    - “peer” automatic summarization systems

# Roadmap

- MEAD
  - Classic end-to-end System
  - Cues to content Extraction
- Bayesian Topic Models
- Graph-based Approaches
  - Random Walks
- Supervised selection
  - Term ranking with rich features

# MEAD

- Radev et al, 2000, 2001, 2004
- Centroid-Based Summarization System
- tf•idf similarity features
- Multiple-Document Summarizer
- Publicly available implementation (no warranty!)
- Solid performance in DUC tasks
- Standard non-trivial evaluation baseline

# Main Ideas

- **Select sentences central to cluster**
  - "cluster-based sentence utility" (CBSU)
    - Measure of sentence relevance to cluster (Score from 0–10)
- **Select distinct representative from equivalence classes**
  - "cross-sentence information subsumption" (CSIS)
    - Sentences including same information content said to "subsume"
      A. John fed Spot
      B. John gave food to Spot and water to the plants
    - $I(B) \subset I(A)$
    - If mutually subsume, form equivalence class.

# Centroid-based Models

- Assume clusters of topically related documents
  - Provided by automatic or manual clusters
- Centroid — pseudo-document of terms with $Count \times IDF$ above some threshold
  - Intuition: centroid terms indicative of topic
  - Count: average # of term occurrences in cluster
  - IDF computed over larger side corpus (e.g. full AQUAINT)

# MEAD Content Selection

- **Input:**
  - Sentence segmented, cluster documents (*n* sents)
  - Compression rate (e.g. 20%)
- **Output:**
  - n×r sentence summary
- Select highest scoring sentences based on
  - Centroid score
  - Position score
  - First-sentence overlap
  - Redundancy

# Score Computation

$$\boldsymbol{Score(s_i) = \sum_{\mathrm{i}} w_c C_i + w_p P_i + w_f F_i}$$

$\boldsymbol{i}$ = i<sup>th</sup> sentence in doc

- $C_i = \sum_i C_{w,i}$
  - Sum over centroid values of words in sentence
- $\boldsymbol{F_i = S_1 \cdot S_i}$
  - Overlap with first sentence
  - TF-based inner product of sentence with first sentence in document.
- $P_i = \left(\frac{(n-i+1)}{n}\right) \times C_{max}$
  - Positional score: $C_{max}$ — score of highest sentence in document
    - Scaled by distance from beginning of document
    - $n$ = doc length

$\boldsymbol{w_c,\ w_p,\ w_f}$ = Weights for different components

# Managing Redundancy

- Alternative redundancy approaches:
  - RedundancyMax:
    - Excludes sentences with cosine overlap > threshold
- Redundancy penalty
  - $R_s = 2 \cdot \frac{\text{\# of overlapping words}}{\text{\# words in sentence pair}}$
    - Subtracted from $Score(s_i) = \sum_i w_c C_i + w_p P_i + w_f F_i - \boxed{w_R R_s}$
    - Weighted by highest scoring sentence in set $(w_R = Max_s(Score(s))$
    - $R_s = 1$ when identical, 0 when no words in common

# System Overview

- **Information Ordering**
  - Chronological by document date
- **Information realization**
  - Pure extraction, no sentence revision
- **Evaluation**
  - Participated in DUC 2001, 2003
    - Among top 5 systems
  - Solid, straightforward system.
  - Publicly available; will compute/output weights.

# Bayesian Topic Models

- Perspective: Generative story for document topics
- Multiple models of word probability, topics
  - General English
  - Input Document Set
  - Individual documents
- Select summary which minimizes KL-divergence
  - Between document set and summary: $KL(P_D \| P_S)$
- Often by greedily selecting sentences
  - Also global models

# Graph-Based Models

- LexRank (Erkan & Radev, 2004)
- Key ideas:
  - Graph-based model of sentence saliency
    - Draws ideas from PageRank, Hyperlink-Induced Topic Search (HITS)
    - Contrasts with direct term-weighting models
    - Beats tf · idf centroid

# Graph View

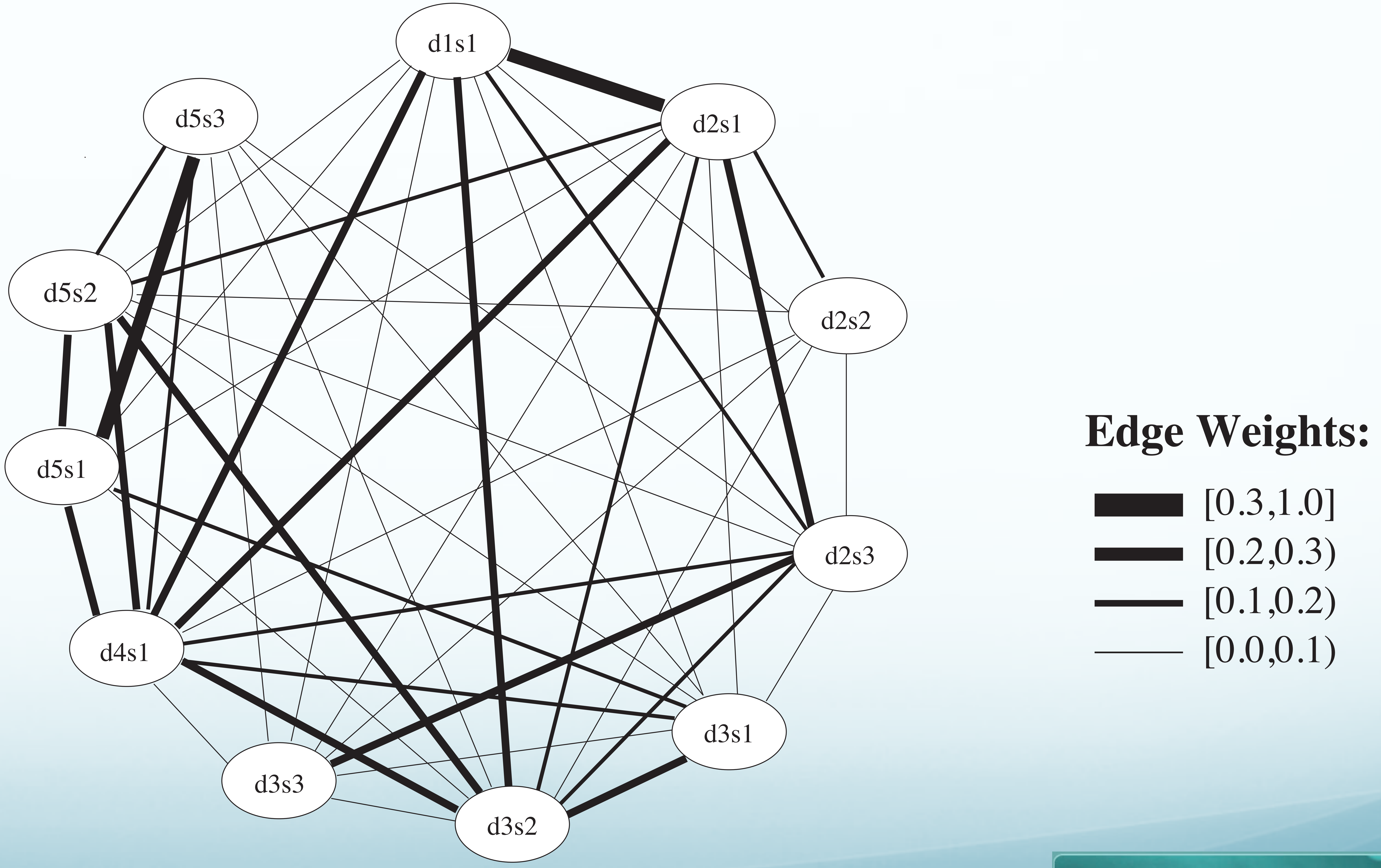
- Centroid Approach:
  - Central pseudo-document of key words in cluster
- Graph-based approach
  - Sentences (or other units) in cluster link to each other
  - Salient if similar to many others
    - More central or relevant to the cluster
  - Low similarity with most others, not central

# Constructing a Graph

- Graph:
  - Nodes — Sentences
  - Edges — measure of similarity between sentences
- How do we compute similarity between nodes?
  - Here: tf · idf modified cosine, but could use other schemes
  - (*tf* = word count within a **sentence**, *df* = within this document, **not** docset)

$$\text{idf-modified-cosine}(x,y) = \frac{\sum_{w \in x,y} \text{tf}_{w,x}\text{tf}_{\text{w,y}}(\text{idf}_w)^2}{\sqrt{\sum_{x_i \in x}(\text{tf}_{x_i,x}\text{idf}_{x_i})^2} \times \sqrt{\sum_{y_i \in y}(\text{tf}_{y_i,y}\text{idf}_{y_i})^2}}$$

# Example Graph




(Erkan & Radev, 2004)

# Constructing a Graph

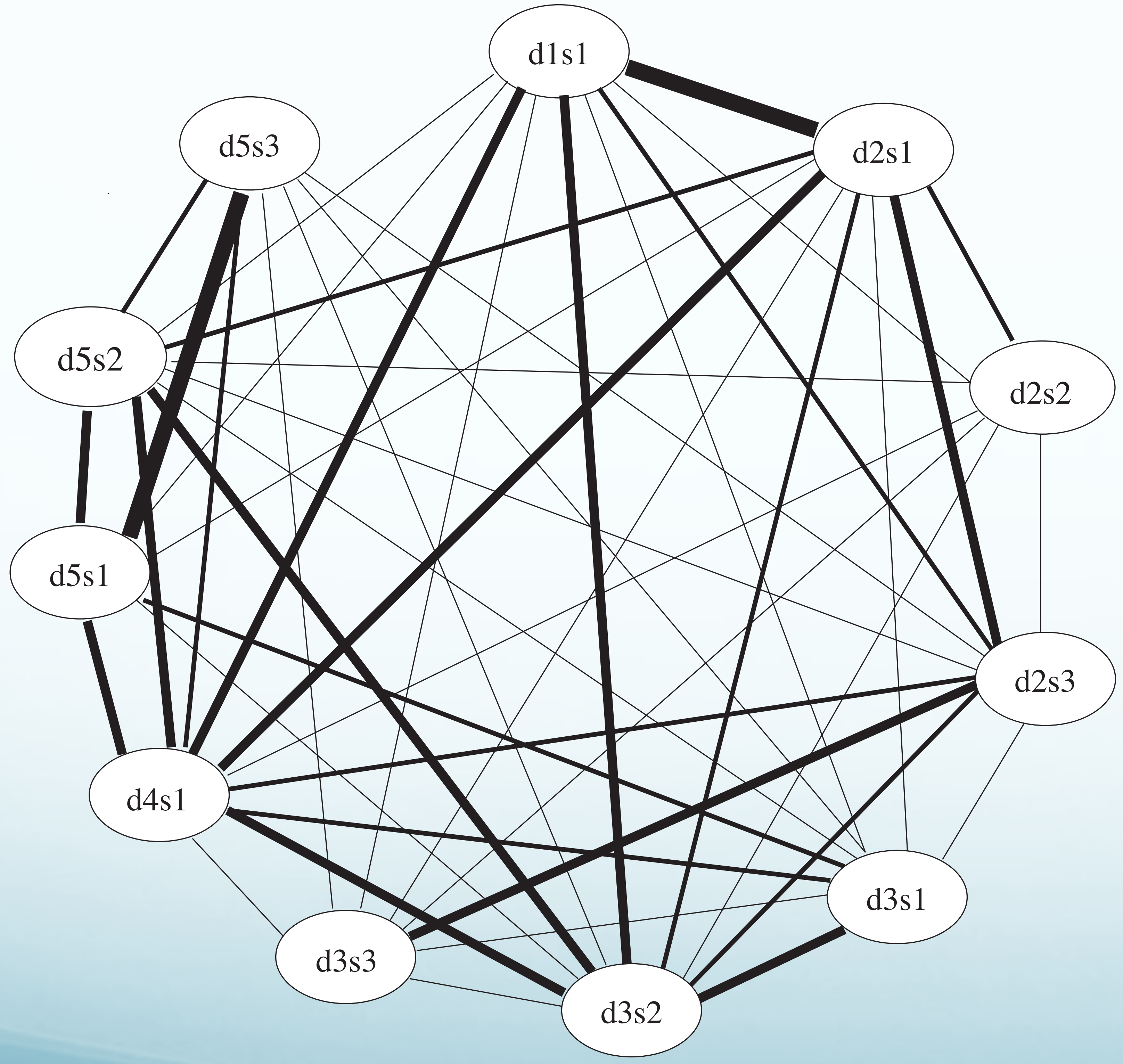
- How do we compute overall sentence saliency?
  - Degree centrality
  - LexRank

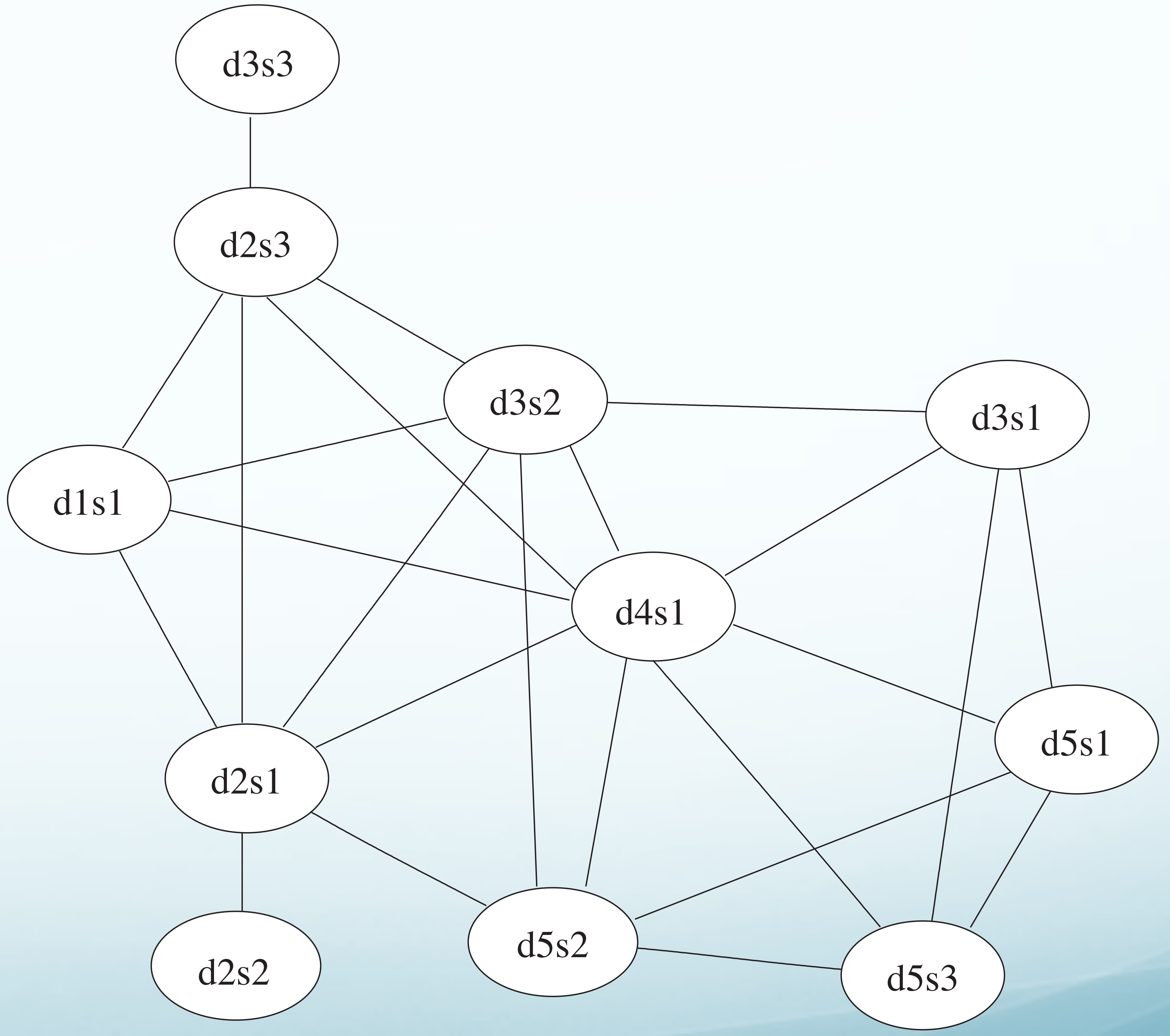
# Degree Centrality

- Centrality — # of neighbors in graph
  - Draw $\mathrm{Edge}(a,b)$ if $\mathrm{sim}(a,b) \geq \mathit{threshold}$

# Threshold = 0

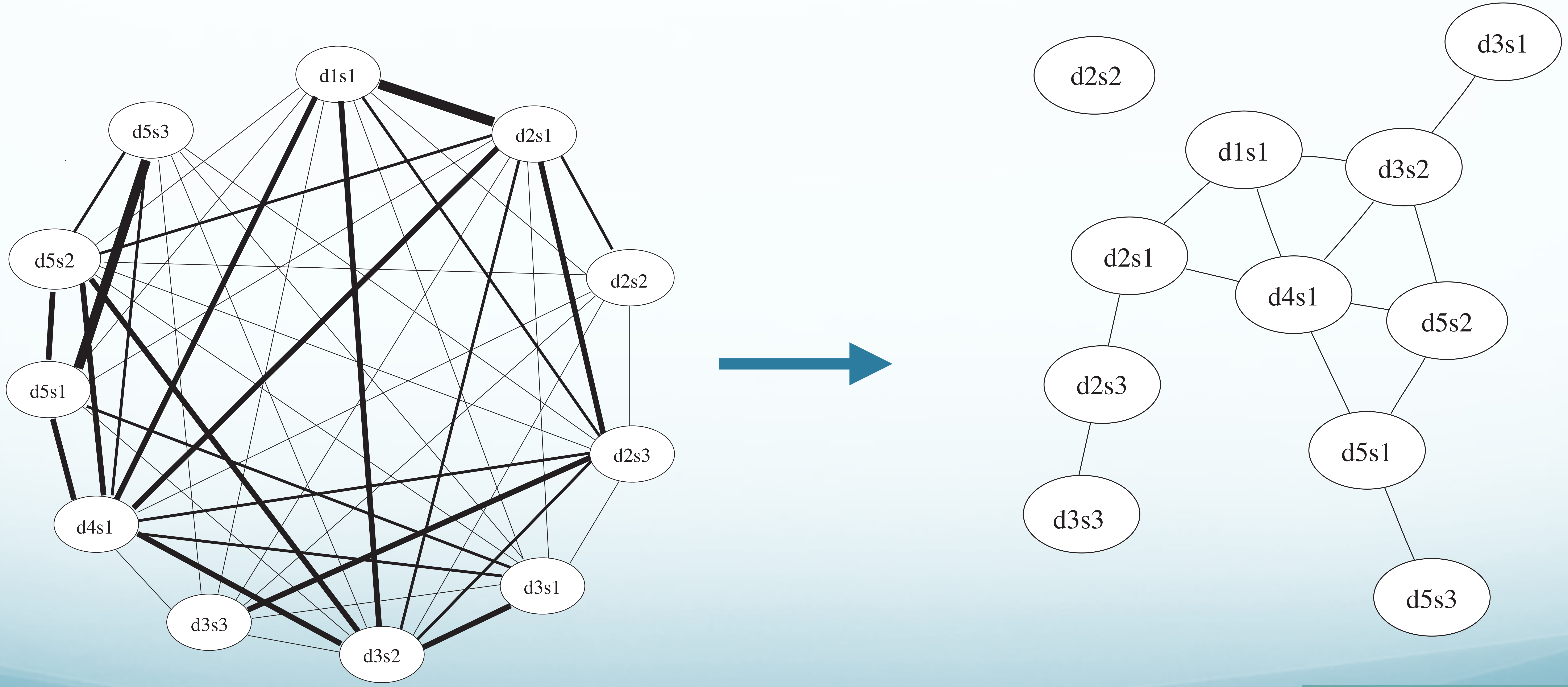
- Almost fully connected
  - Not particularly informative



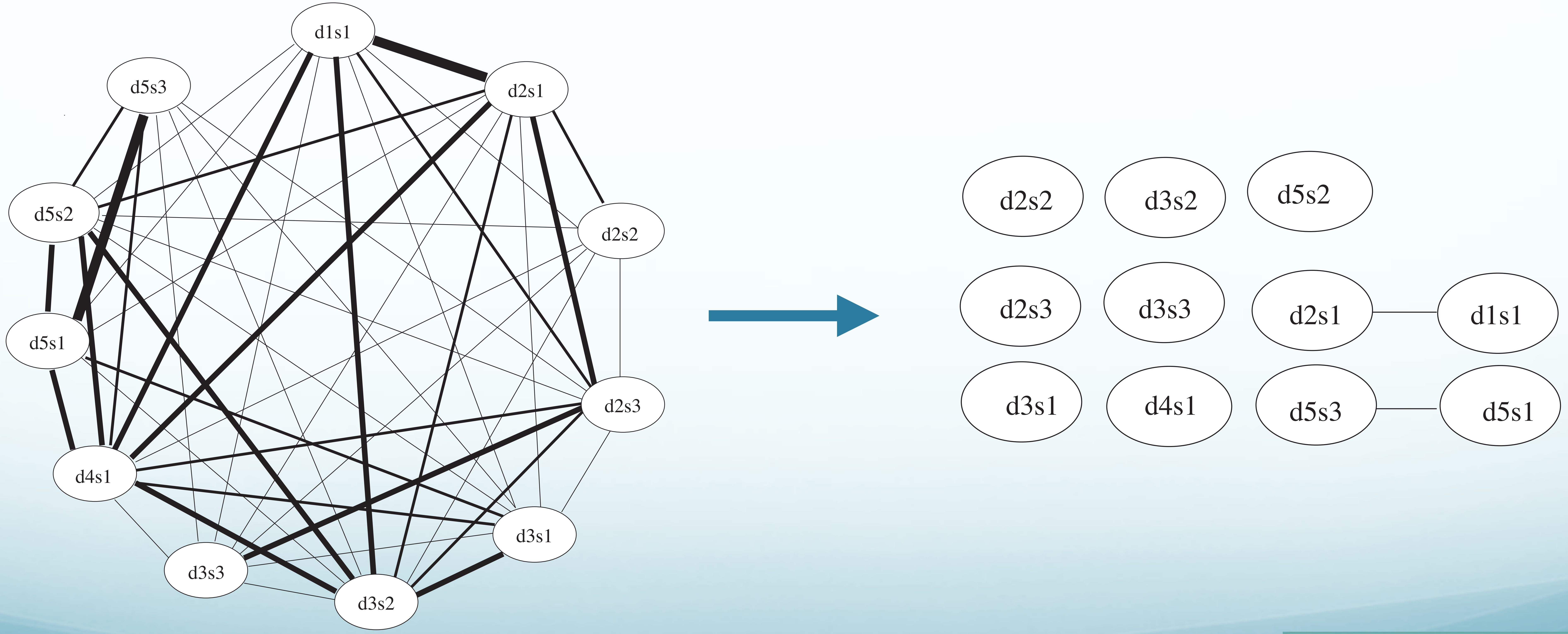
# Threshold = 0.1–0.2

- Some filtering, can be useful



# Threshold = 0.3

- Only two connections remain, also uninformative



# LexRank

- Degree centrality: 1 edge = 1 vote
  - Possibly problematic
    - e.g. erroneous document in cluster, some sentence may score high
- LexRank idea:
  - Node can have high(er) score via high scoring neighbors
    - Same idea as PageRank, HITS
      - Page ranked high b/c incoming link from high ranking pages

$p(u)$ is centrality
$adj(u)$ = adjacent nodes
$deg(v)$ = degree

$$p(u) = \sum_{v \in adj(u)} \frac{p(v)}{deg(v)}$$

# Power Method: Computing the Weights

```
Input: Adjacency matrix M
Initialize p0 (uniform distribution)
t=0;
repeat
   t=t+1
   pt=MTpt-1
until convergence
Return pt
```

Computes the stationary distribution of a Markov chain

# LexRank: Computing the Weights

- Can think of matrix $X$ as transition matrix of Markov chain
  - $\mathrm{X(i,j)}$ is probability of transition from state *i* to state *j*
- Will converge to a stationary distribution (r)
  - Given certain properties (aperiodic, irreducible)
  - Probability of ending up in each state via random walk
- Can compute iteratively to converge via PageRank formula ($d$ is “damping factor”):

$$p(u) = \frac{d}{N} + (1-d) \sum_{v \in adj(u)} \frac{p(v)}{deg(v)}$$

# LexRank Score Example

- For earlier graph:

| ID | LR(0.1) | LR(0.2) | LR(0.3) | Centroid |
|---|---|---|---|---|
| d1s1 | 0.6007 | 0.6944 | 1.0000 | 0.7209 |
| d2s1 | 0.8466 | 0.7317 | 1.0000 | 0.7249 |
| d2s2 | 0.3491 | 0.6773 | 1.0000 | 0.1356 |
| d2s3 | 0.7520 | 0.6550 | 1.0000 | 0.5694 |
| d3s1 | 0.5907 | 0.4344 | 1.0000 | 0.6331 |
| d3s2 | 0.7993 | 0.8718 | 1.0000 | 0.7972 |
| d3s3 | 0.3548 | 0.4993 | 1.0000 | 0.3328 |
| d4s1 | 1.0000 | 1.0000 | 1.0000 | 0.9414 |
| d5s1 | 0.5921 | 0.7399 | 1.0000 | 0.9580 |
| d5s2 | 0.6910 | 0.6967 | 1.0000 | 1.0000 |
| d5s3 | 0.5921 | 0.4501 | 1.0000 | 0.7902 |

# Continuous LexRank

- Basic LexRank ignores similarity scores
  - Except for initial thresholding of adjacency
- Could just use weights directly (rather than degree)

$$p(u) = \frac{d}{N} + (1-d) \sum_{v \in adj(u)} \frac{\cos(u,v)}{\sum_{z \in adj(v)} \cos(z,v)} p(v)$$

# Advantages vs. Centroid

- Captures information subsumption
  - Highly ranked sentences have greatest overlap w/adjacent
  - Will promote those sentences
- Reduces impact of spurious high-IDF terms
  - Rare terms get very high weight (reduce TF)
  - Lead to selection of sentences w/high IDF terms
  - Effect minimized in LexRank

# Example Results

- Beat official DUC 2004 entrants
  - All versions beat baselines and centroid

| | 2004 Task 2 | | |
|---|---|---|---|
| | min | max | average |
| Centroid | 0.3580 | 0.3767 | 0.3670 |
| Degree (t=0.1) | 0.3590 | 0.3830 | 0.3707 |
| LexRank (t=0.1) | 0.3646 | 0.3808 | 0.3736 |
| Cont. LexRank | 0.3617 | 0.3826 | 0.3758 |

baselines: random: 0.3238
lead-based: 0.3686

# Content Selection: Supervised

# Supervised Word Selection

- RegSumm
  - Improving the Estimation of Word Importance for news Multi-Document Summarization (Hong & Nenkova, '14)
- Key ideas:
  - Supervised method for word selection
  - Diverse, rich feature set
    - Unsupervised measures, POS, NER, position, etc
  - Identification of common "important" words via side corpus of news articles and human summaries

# Basic Approach

- Learn Keyword Imporatnce
  - Contrast with unsupervised selection, learning sentences
- Train regression over large number of possible features
  - Supervision over *words*
    - Did document word appear in summary or not?
- Greedy sentence selection
  - Highest scoring sentences: average word weight
  - Do not add if $\geq$ 0.5 cosine similarity with current sentences

# Features I

- Unsupervised measures — (binary features given a threshold)
- Word probability: $\frac{count(w)}{N}$
  - computed over input cluster
- Log likelihood ratio: Gigaword used for background corpus
- Markov Random Walk (MRW)
  - Graphical model approach like LexRank
  - Nodes = words
  - Edges = # of syntactic dependencies between words in sentence
  - Weights via PageRank algorithm

# Features II

- "Global" word importance
  - Are there words which are intrinsically likely to show up in (news) summaries?
- Approach:
  - Build language models on NYT corpus of articles & summaries
    - One model on articles, one model on summaries
    - Measures: $Pr_A(w)$, $Pr_A(w) \cdot Pr_G(w)$, $Pr_A(w)/Pr_G(w)$
    - $KL(A\|G) = Pr_A(w) \cdot \ln\left(\frac{Pr_A(w)}{Pr_G(w)}\right)$ $KL(G\|A) = Pr_G(w) \cdot \ln\left(\frac{Pr_G(w)}{Pr_A(w)}\right)$
  - Binary features: top-k or bottom-k features

# Features III

- Adaptations of Common Features
  - Word position as proportion of document [0,1]
    - Earliest first, latest last, average, average first
  - Word type: POS, NER
    - Emphasizes NNS, NN, capitalization; ORG, PERS, LOC
  - **Sentiment (MPQA + LIWC)**
    - **MPQA: Multi-Perspective Question Answering** — (sentiment, subjectivity terms)
      - Strong sentiment likely or not? NOT
    - LIWC: **Linguistic Inquiry and Word Count**
      - **words** for 69 categories
      - {positive feature: death, anger, money} {negative feature: pron, neg, function words, swear, adverbs, etc}

# Assessment: Words

- Select N highest ranked keywords via regression
- Compute F-measure over words in summaries
  - $G_i$: i = # of summaries in which word appears

| $G_i$ | #words | PROB | LLR | MRW | REGBASIC | REGSUM |
|---|---|---|---|---|---|---|
| $G_1$ | 80 | 43.6 | 37.9 | 38.9 | 39.9 | 45.7 |
| $G_1$ | 100 | 44.3 | 38.7 | 39.2 | 41.0 | 46.5 |
| $G_1$ | 120 | 44.6 | 38.5 | 39.2 | 40.9 | 46.4 |
| $G_2$ | 30 | 47.8 | 44.0 | 42.4 | 47.4 | 50.2 |
| $G_2$ | 35 | 47.1 | 43.3 | 42.1 | 47.0 | 49.5 |
| $G_2$ | 40 | 46.5 | 42.4 | 41.8 | 46.4 | 49.2 |

# Assessment: Summaries

- Summarization with ROUGE-1,2,4

| | System | R-1 | R-2 | R-4 |
|---|---|---|---|---|
| Basic Systems | PROB | 35.14 | 8.17 | 1.06 |
| | LLR | 34.60 | 7.56 | 0.83 |
| | MRW | 35.78 | 8.15 | 0.99 |
| | REGBASIC | 37.56 | 9.28 | 1.49 |
| SotA Systems | KL | 37.97 | 8.53 | 1.26 |
| | PEER -65 | 37.62 | 8.96 | 1.51 |
| | SUBMOD | 39.18 | 9.35 | 1.39 |
| | DPP | 39.79 | 9.62 | 1.57 |
| | **REGSUM** | **38.57** | **9.75** | **1.60** |