Mining Newsgroups Using Networks Arising From Social Behavior by Rakesh Agrawal et al.

Presented by Will Lee
wwlee1@uiuc.edu

September 28, 2004

Motivation

- IR on newsgroups is challenging due to lack of connection among documents
  - Unlike WWW, can not use PageRank to improve the retrieval performance
- An automatically-generated social network within a newsgroup may help IR and text mining applications

Methods Overview

- Classify authors as "for" or "against" a topic

Methods Overview

- Classify authors as "for" or "against" a topic
- Uses graph-theoretic approach to partition the *interaction* graph into two partitions

Methods Overview

- Classify authors as "for" or "against" a topic
- Uses graph-theoretic approach to partition the *interaction* graph into two partitions
  - *graph nodes* = users

Methods Overview

- Classify authors as "for" or "against" a topic
- Uses graph-theoretic approach to partition the *interaction* graph into two partitions
  - *graph nodes* = users
  - *interaction (graph edges)* = an user replying to another

Methods Overview

- Classify authors as "for" or "against" a topic
- Uses graph-theoretic approach to partition the *interaction* graph into two partitions
  - *graph nodes* = users
  - *interaction (graph edges)* = an user replying to another
- Assumptions

Methods Overview

- Classify authors as "for" or "against" a topic
- Uses graph-theoretic approach to partition the *interaction* graph into two partitions
  - *graph nodes* = users
  - *interaction (graph edges)* = an user replying to another
- Assumptions
  - New posts contain opposite comments against parent posts

Methods Overview

- Classify authors as "for" or "against" a topic
- Uses graph-theoretic approach to partition the *interaction* graph into two partitions
  - *graph nodes* = users
  - *interaction (graph edges)* = an user replying to another
- Assumptions
  - New posts contain opposite comments against parent posts
  - There are only two groups of users with roughly the same size

Newsgroup Threads

Re: combining open office spellchecker with Luce...	David Spencer	9/9/2004 11:01...
Re: combining open office spellchecker with Lu...	Andrzej Bialecki	9/9/2004 11:15...
Re: combining open office spellchecker with...	David Spencer	9/9/2004 11:51...
Re: combining open office spellchecker with Lu...	Doug Cutting	9/9/2004 12:03...
Re: combining open office spellchecker with...	David Spencer	9/9/2004 1:10 ...
Re: combining open office spellchecker ...	Doug Cutting	9/9/2004 10:09...
Re: combining open office spellchecke...	eks dev	3:04 AM
Re: combining open office spellche...	David Spencer	10:05 AM
frequent terms - Re: combining open office ...	David Spencer	7:38 PM
Re: MultiFieldQueryParser seems broken... Fix att...	Doug Cutting	9/9/2004 11:52...
Re: MultiFieldQueryParser seems broken... Fix ...	Daniel Naber	9/9/2004 12:28...
Re: MultiFieldQueryParser seems broken... ...	Doug Cutting	12:50 PM
Re: MultiFieldQueryParser seems broken... Fix ...	Bill Janssen	9/9/2004 2:48 ...
Re: MultiFieldQueryParser seems broken... Fix ...	Bill Janssen	9/9/2004 2:53 ...
Lucene working example.	Mr dharmanand ...	9/9/2004 12:44...
Out of memory in lucene 1.4.1 when re-indexing ...	Daniel Taurat	9/9/2004 12:47...
Re: Out of memory in lucene 1.4.1 when re-ind...	Daniel Naber	9/9/2004 2:30 ...
Re: Out of memory in lucene 1.4.1 when re-i...	Daniel Taurat	7:10 AM

Graph Partitioning

- Define a graph $G(V, E)$

Graph Partitioning

- Define a graph $G(V, E)$
- V = newsgroup participants

Graph Partitioning

- Define a graph $G(V, E)$
- V = newsgroup participants
- $e \in E$ where $e = (v_i, v_j)$ and $v_i, v_j \in V$ such that v_i has responded to a post by v_j

Graph Partitioning

- Define a graph $G(V, E)$
- V = newsgroup participants
- $e \in E$ where $e = (v_i, v_j)$ and $v_i, v_j \in V$ such that v_i has responded to a post by v_j
- Goal is to find set of verticies F (for) and A (against)

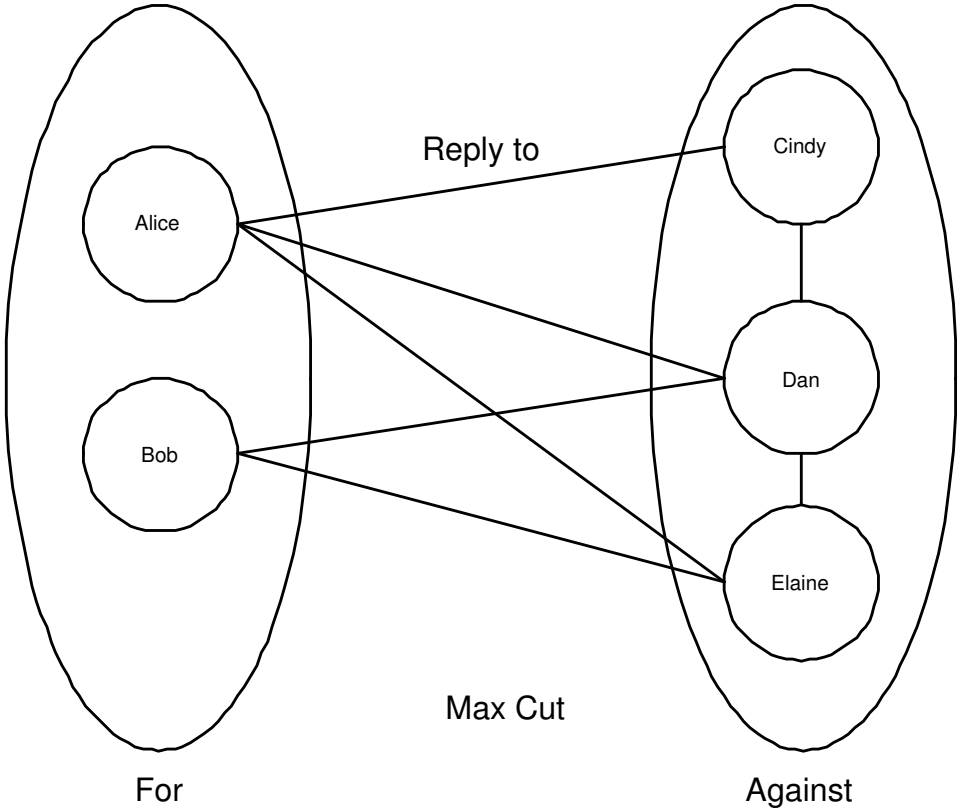
Graph Partitioning

- Define a graph $G(V, E)$
- V = newsgroup participants
- $e \in E$ where $e = (v_i, v_j)$ and $v_i, v_j \in V$ such that v_i has responded to a post by v_j
- Goal is to find set of verticies F (for) and A (against)
- Maximize the cut function $f(F, A) = |E \cap (F \times A)|$ (NP-complete problem)

Graph Partitioning

- Define a graph $G(V, E)$
- V = newsgroup participants
- $e \in E$ where $e = (v_i, v_j)$ and $v_i, v_j \in V$ such that v_i has responded to a post by v_j
- Goal is to find set of verticies F (for) and A (against)
- Maximize the cut function $f(F, A) = |E \cap (F \times A)|$ (NP-complete problem)
- Uses spectral partitioning for efficiency

Turning Social Behavior Into Graph Problem



Graph Partitioning Methods

1. EV Algorithm

   (a) Co-citation matrix $D = GG^T$ with weighted edge $w = \#$ of people "co-cited" by author u_1 and u_2. Think of D as a similarity matrix for author u_i and u_j.
   (b) Second eigenvector of D is a good approximation of G's bipartition

Graph Partitioning Methods

1. EV Algorithm

   (a) Co-citation matrix $D = GG^T$ with weighted edge $w = \#$ of people "co-cited" by author u_1 and u_2. Think of D as a similarity matrix for author u_i and u_j.
   (b) Second eigenvector of D is a good approximation of G's bipartition

2. EV + KL

   (a) Uses the Kernighan-Lin heuristic to improve the partitioning

Graph Partitioning Methods

1. EV Algorithm
   (a) Co-citation matrix $D = GG^T$ with weighted edge $w = \#$ of people "co-cited" by author u_1 and u_2. Think of D as a similarity matrix for author u_i and u_j.
   (b) Second eigenvector of D is a good approximation of G's bipartition
2. EV + KL
   (a) Uses the Kernighan-Lin heuristic to improve the partitioning
3. EV (Constrained) and EV + KL (Constrained)
   (a) Identify some "for" and "against" authors, group them as one node

Graph Partitioning Methods

1. EV Algorithm
   (a) Co-citation matrix $D = GG^T$ with weighted edge $w = \#$ of people "co-cited" by author u_1 and u_2. Think of D as a similarity matrix for author u_i and u_j.
   (b) Second eigenvector of D is a good approximation of G's bipartition
2. EV + KL
   (a) Uses the Kernighan-Lin heuristic to improve the partitioning
3. EV (Constrained) and EV + KL (Constrained)
   (a) Identify some "for" and "against" authors, group them as one node
4. Iterative Classification

Graph Partitioning Methods

1. EV Algorithm
   (a) Co-citation matrix $D = GG^T$ with weighted edge $w = \#$ of people "co-cited" by author u_1 and u_2. Think of D as a similarity matrix for author u_i and u_j.
   (b) Second eigenvector of D is a good approximation of G's bipartition
2. EV + KL
   (a) Uses the Kernighan-Lin heuristic to improve the partitioning
3. EV (Constrained) and EV + KL (Constrained)
   (a) Identify some "for" and "against" authors, group them as one node
4. Iterative Classification
   (a) Initialize: Label "for" and "against" for a small number of people in the newsgroup

(b) Iterate m times:

(b) Iterate m times:

i. Calculate the $s(v_i)$ for each node v_i . The weight w_{ij} is the weight between node v_j and v_i):

$$s(v_i) = \frac{\sum_j -s(v_j) \times w_{ij}}{\sum_j w_{ij}}$$

(b) Iterate m times:

i. Calculate the $s(v_i)$ for each node v_i . The weight w_{ij} is the weight between node v_j and v_i):

$$s(v_i) = \frac{\sum_j -s(v_j) \times w_{ij}}{\sum_j w_{ij}}$$

ii. Sort the labels (sign of $s(v_i)$) by confidence ($|s(v_i)|$)

(b) Iterate m times:

i. Calculate the $s(v_i)$ for each node v_i . The weight w_{ij} is the weight between node v_j and v_i):

$$s(v_i) \quad = \quad \frac{\sum_j -s(v_j) \times w_{ij}}{\sum_j w_{ij}}$$

ii. Sort the labels (sign of $s(v_i)$) by confidence ($|s(v_i)|$)

iii. Accept $k = N \times \frac{i}{m}$ labels where i = iteration, m = total iterations, and N = number of instances in test data

Evaluation

- Uses three newsgroups – Abortion, Gun Control, and Immigration
- Manually tag 50 random people in the "for" or "against" categories
- Comparing with classic classification algorithms (Naive Bayes & SVM) that work on message content

	Abortion	Gun Control	Immigration
Majority	57%	72%	54%
SVM	55%	42%	55%
Naive Bayes	50%	72%	54%
Iterative	67%	80%	83%
EV/EV+KL	73%/75%	78%/74%	50%/52%
Constrained EV/EV+KL	73%/73%	84%/82%	88%/88%

- Also, sensitivity experiments show more posts = more bias posts = higher accuracy

Contributions / Limitations

- Contributions
  - Apply graph-theoretic algorithms to a new domain
  - Sensitivity analysis on simulated newsgroup data
- Limitations
  - Assume users post against each other, may not be true in some newsgroups (technical ones)
  - Constrained and iterative method still need training data
  - Should justify why the constrained methods perform much better than the unconstrained ones

Discussion Questions

- How does user partitioning help IR?
- In a complex web of discussions within a newsgroup, users may not belong to the same "for" or "against" group for all topics. How can this system be applied on such newsgroup?
- How is this system similar to the PageRank algorithm? Is there any other way to draw connection among the newsgroup postings?