You're Mr. Lebowski, I'm the Dude
Inducing Address Term Formality in Signed Social Networks

Vinodh Krishnan and Jacob Eisenstein

Georgia Institute of Technology

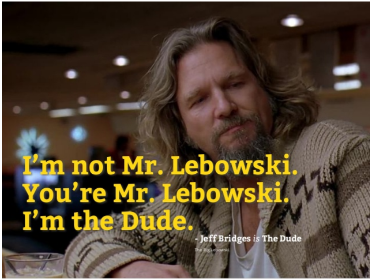
Putting language in a social context

Lillian Lee: "Putting the ACL in Computational Social Science"

- **Social network analysis** captures structural phenomena like information diffusion and networked bargaining.
- **Natural language processing** captures meaning in communication: entities, topics, sentiment, argumentation, etc.

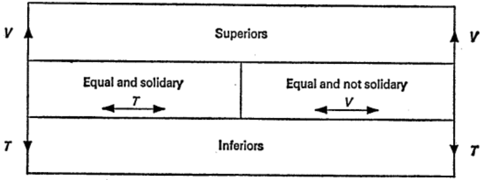
Formality of address

- **Address terms** reflect social structure in language, indicating level of formality.



Prior work on formality of address

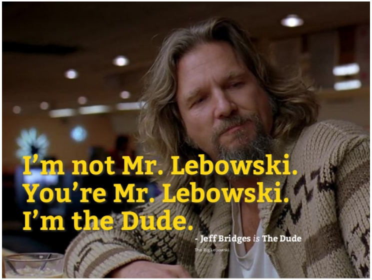
- **T/V** system of pronouns
(R. Brown and Gilman, 1960)



- Forms of address in politeness theory
(P. Brown and Levinson, 1987)
- Project formality from German, where it is morphologically marked
(Faruqui and Padó, 2011, 2012)

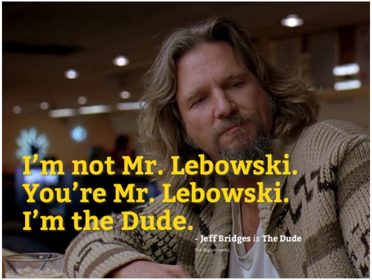
Formality of address

- **Address terms** reflect social structure in language, indicating level of formality.



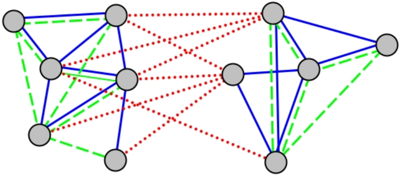
Formality of address

- **Address terms** reflect social structure in language, indicating level of formality.
- Formality can be modeled as a **signed social network**.



Prior work on signed social networks

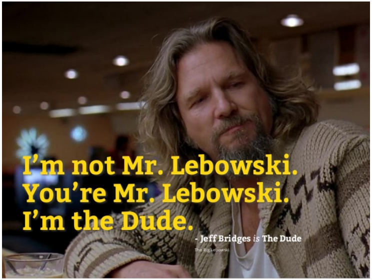
- **Structural balance theory**: stability of signed networks is characterized by triads. (Cartright and Harary 1956)



- Structural balance in social media (Leskovec et al 2010)
- Linguistic sentiment in signed social networks (Hassan et al 2012; West et al 2014)

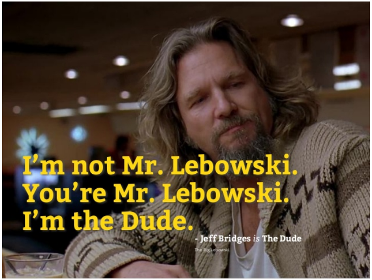
Formality of address

- **Address terms** reflect social structure in language, indicating level of formality.
- Formality can be modeled as a **signed social network**.



Formality of address

- **Address terms** reflect social structure in language, indicating level of formality.
- Formality can be modeled as a **signed social network**.



This research: model formality as a latent variable, unifying linguistics and social network analysis.

Modeling formality of address

- What is the nature of the relationships between agents in a social network?
- Are there regular structures that emerge across signed networks?
- How does language reflect and reproduce social relationships?

Modeling formality of address

- What is the nature of the relationships between agents in a social network?

  Induce a clustering of edges by formality.

- Are there regular structures that emerge across signed networks?

- How does language reflect and reproduce social relationships?

Modeling formality of address

- What is the nature of the relationships between agents in a social network?

  Induce a clustering of edges by formality.

- Are there regular structures that emerge across signed networks?

  Estimate a prior distribution over signed network structures.

- How does language reflect and reproduce social relationships?

Modeling formality of address

- What is the nature of the relationships between agents in a social network?

  Induce a clustering of edges by formality.

- Are there regular structures that emerge across signed networks?

  Estimate a prior distribution over signed network structures.

- How does language reflect and reproduce social relationships?

  Estimate a likelihood distribution over address terms given formality.

A generative model of networked content

Basic notation:

- Undirected graph $G = \{\langle i, j \rangle\}$, $i < j$
- All edges $\langle i, j \rangle \in G$ have labels $y_{ij} \in \mathcal{Y}$
- Edges have **content** $\boldsymbol{x}_{i \rightarrow j}, \boldsymbol{x}_{i \leftarrow j} \in \mathbb{N}^V$, where V is the size of the vocabulary

A generative model of networked content

Basic notation:

- Undirected graph $G = \{\langle i, j \rangle\}$, $i < j$
- All edges $\langle i, j \rangle \in G$ have labels $y_{ij} \in \mathcal{Y}$
- Edges have **content** $\boldsymbol{x}_{i \to j}, \boldsymbol{x}_{i \leftarrow j} \in \mathbb{N}^V$, where V is the size of the vocabulary

$$P(\boldsymbol{y}, \boldsymbol{x} \mid G; \Theta, \boldsymbol{\beta}, \boldsymbol{\eta}) = P(\boldsymbol{x} \mid \boldsymbol{y}; \Theta) P(\boldsymbol{y} \mid G; \boldsymbol{\beta}, \boldsymbol{\eta})$$

- The likelihood factors across dyads;
- The prior factors across dyads and triads.

A generative model of networked content

Basic notation:

- Undirected graph $G = \{\langle i, j \rangle\}$, $i < j$
- All edges $\langle i, j \rangle \in G$ have labels $y_{ij} \in \mathcal{Y}$
- Edges have **content** $\boldsymbol{x}_{i \to j}, \boldsymbol{x}_{i \leftarrow j} \in \mathbb{N}^V$, where V is the size of the vocabulary

$$P(\boldsymbol{y}, \boldsymbol{x} \mid G; \Theta, \boldsymbol{\beta}, \boldsymbol{\eta}) = P(\boldsymbol{x} \mid \boldsymbol{y}; \Theta) P(\boldsymbol{y} \mid G; \boldsymbol{\beta}, \boldsymbol{\eta})$$

- The **likelihood** factors across dyads;
- The prior factors across dyads and triads.

A generative model of networked content

Basic notation:

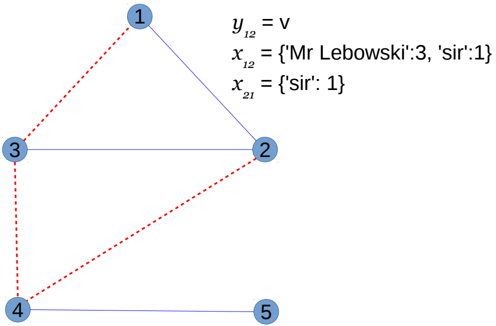
- Undirected graph $G = \{\langle i, j \rangle\}$, $i < j$
- All edges $\langle i, j \rangle \in G$ have labels $y_{ij} \in \mathcal{Y}$
- Edges have **content** $\boldsymbol{x}_{i \to j}, \boldsymbol{x}_{i \leftarrow j} \in \mathbb{N}^V$, where V is the size of the vocabulary

$$P(\boldsymbol{y}, \boldsymbol{x} \mid G; \Theta, \boldsymbol{\beta}, \boldsymbol{\eta}) = P(\boldsymbol{x} \mid \boldsymbol{y}; \Theta) P(\boldsymbol{y} \mid G; \boldsymbol{\beta}, \boldsymbol{\eta})$$

- The likelihood factors across dyads;
- The **prior** factors across dyads and triads.

Example

At inference time, we observe x but not y.



Performing statistical inference over y gives a labeling over edges.

Likelihood

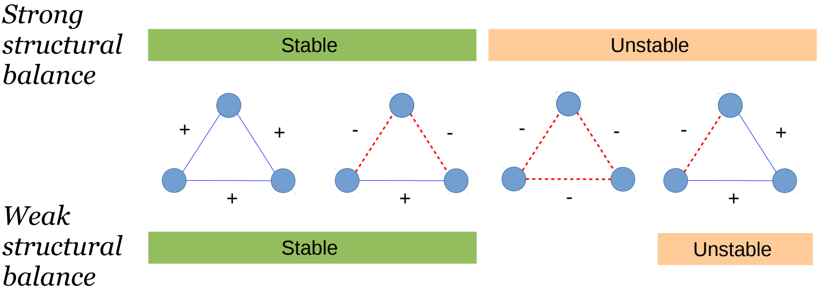
- The likelihood $P(\boldsymbol{x} \mid \boldsymbol{y})$ captures how formality is expressed in language.
- Intuitively,

$$P(\text{dude} \mid y = \text{FORMAL}) < P(\text{dude} \mid y = \text{INFORMAL})$$
$$P(\text{sir} \mid y = \text{FORMAL}) > P(\text{sir} \mid y = \text{INFORMAL})$$

- These probabilities are expressed in the parameter $\boldsymbol{\theta}$, which is learned from data.
- We can generalize to directed signs by distinguishing $\boldsymbol{\theta}^{\rightarrow}_{y_{ij}}$ and $\boldsymbol{\theta}^{\leftarrow}_{y_{ij}}$.

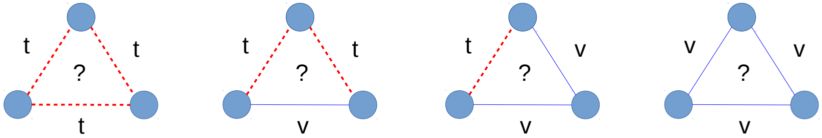
Adding social theory

- So far, this is just a mixture model over dyads.
- But social theory may tell us that not all label configurations are equally likely.
- Ex: **structural balance theory** describes networks of friend/enemy links, where signed triads may be stable or unstable:



Social theories with unknown parameters

- What if the magnitude, and even the direction of the effect of each triad type is a priori unknown?



- We assume a triadic form, but make no assumptions about the specifics.

Triadic stability in a prior distribution

Assume the prior factors over dyads and triads.

$$P(\boldsymbol{y}; G, \boldsymbol{\eta}, \boldsymbol{\beta}) = \frac{1}{Z(\boldsymbol{\eta}, \boldsymbol{\beta}; G)} \times \exp \sum_{\langle i,j \rangle \in G} \boldsymbol{\eta}^\top \boldsymbol{f}(y_{ij}, i, j, G)$$
$$\times \exp \sum_{\langle i,j,k \rangle \in \mathcal{T}(G)} \beta_{y_{ij}, y_{jk}, y_{ik}},$$

where,

- $Z(\boldsymbol{\eta}, \boldsymbol{\beta}; G)$ is a normalizing constant;
- $\boldsymbol{f}(y_{ij}, i, j, G)$ is a set of dyad features, with associated weights $\boldsymbol{\eta}$;
- $\mathcal{T}(G)$ is the set of triads in the graph G;
- $\beta_{y_{ij}, y_{jk}, y_{ik}}$ scores the stability of a triad type.

Triadic stability in a prior distribution

Assume the prior factors over dyads and triads.

$$P(\boldsymbol{y}; G, \boldsymbol{\eta}, \boldsymbol{\beta}) = \frac{1}{Z(\boldsymbol{\eta}, \boldsymbol{\beta}; G)} \times \exp \sum_{\langle i,j \rangle \in G} \boldsymbol{\eta}^\top \boldsymbol{f}(y_{ij}, i, j, G) \\ \times \exp \sum_{\langle i,j,k \rangle \in \mathcal{T}(G)} \beta_{y_{ij}, y_{jk}, y_{ik}},$$

where,

- $Z(\boldsymbol{\eta}, \boldsymbol{\beta}; G)$ is a normalizing constant;
- $\boldsymbol{f}(y_{ij}, i, j, G)$ is a set of dyad features, with associated weights $\boldsymbol{\eta}$;
- $\mathcal{T}(G)$ is the set of triads in the graph G;
- $\beta_{y_{ij}, y_{jk}, y_{ik}}$ scores the stability of a triad type.

Triadic stability in a prior distribution

Assume the prior factors over dyads and triads.

$$P(\boldsymbol{y}; G, \boldsymbol{\eta}, \boldsymbol{\beta}) = \frac{1}{Z(\boldsymbol{\eta}, \boldsymbol{\beta}; G)} \times \exp \sum_{\langle i,j \rangle \in G} \boldsymbol{\eta}^\top \boldsymbol{f}(y_{ij}, i, j, G)$$
$$\times \exp \sum_{\langle i,j,k \rangle \in \mathcal{T}(G)} \beta_{y_{ij}, y_{jk}, y_{ik}},$$

where,

- $Z(\boldsymbol{\eta}, \boldsymbol{\beta}; G)$ is a normalizing constant;
- $\boldsymbol{f}(y_{ij}, i, j, G)$ is a set of dyad features, with associated weights $\boldsymbol{\eta}$;
- $\mathcal{T}(G)$ is the set of triads in the graph G;
- $\beta_{y_{ij}, y_{jk}, y_{ik}}$ scores the stability of a triad type.

Triadic stability in a prior distribution

Assume the prior factors over dyads and triads.

$$P(\boldsymbol{y}; G, \boldsymbol{\eta}, \boldsymbol{\beta}) = \frac{1}{Z(\boldsymbol{\eta}, \boldsymbol{\beta}; G)} \times \exp \sum_{\langle i,j\rangle \in G} \boldsymbol{\eta}^\top \boldsymbol{f}(y_{ij}, i, j, G)$$
$$\times \exp \sum_{\langle i,j,k\rangle \in \mathcal{T}(G)} \beta_{y_{ij}, y_{jk}, y_{ik}},$$

where,

- $Z(\boldsymbol{\eta}, \boldsymbol{\beta}; G)$ is a normalizing constant;
- $\boldsymbol{f}(y_{ij}, i, j, G)$ is a set of dyad features, with associated weights $\boldsymbol{\eta}$;
- $\mathcal{T}(G)$ is the set of triads in the graph G;
- $\beta_{y_{ij}, y_{jk}, y_{ik}}$ scores the stability of a triad type.

Triadic stability in a prior distribution

Assume the prior factors over dyads and triads.

$$P(\boldsymbol{y}; G, \boldsymbol{\eta}, \boldsymbol{\beta}) = \frac{1}{Z(\boldsymbol{\eta}, \boldsymbol{\beta}; G)} \times \exp \sum_{\langle i,j \rangle \in G} \boldsymbol{\eta}^\top \boldsymbol{f}(y_{ij}, i, j, G)$$
$$\times \exp \sum_{\langle i,j,k \rangle \in \mathcal{T}(G)} \beta_{y_{ij}, y_{jk}, y_{ik}},$$

where,

- $Z(\boldsymbol{\eta}, \boldsymbol{\beta}; G)$ is a normalizing constant;
- $\boldsymbol{f}(y_{ij}, i, j, G)$ is a set of dyad features, with associated weights $\boldsymbol{\eta}$;
- $\mathcal{T}(G)$ is the set of triads in the graph G;
- $\beta_{y_{ij}, y_{jk}, y_{ik}}$ scores the stability of a triad type.

Complete model specification

$$P(\boldsymbol{y}, \boldsymbol{x} \mid G; \Theta, \boldsymbol{\beta}, \boldsymbol{\eta}) = P(\boldsymbol{x} \mid \boldsymbol{y}; \Theta) P(\boldsymbol{y} \mid G; \boldsymbol{\beta}, \boldsymbol{\eta})$$

Bayesian inference answers several questions:

1. What is the label (formality) of each dyad?
2. How is formality expressed in language?
3. Are there structural regularities across signed networks?

Complete model specification

$$P(\boldsymbol{y}, \boldsymbol{x} \mid G; \Theta, \boldsymbol{\beta}, \boldsymbol{\eta}) = P(\boldsymbol{x} \mid \boldsymbol{y}; \Theta) P(\boldsymbol{y} \mid G; \boldsymbol{\beta}, \boldsymbol{\eta})$$

Bayesian inference answers several questions:

1. What is the label (formality) of each dyad?
2. How is formality expressed in language?
3. Are there structural regularities across signed networks?

Complete model specification

$$P(\boldsymbol{y}, \boldsymbol{x} \mid G; \Theta, \boldsymbol{\beta}, \boldsymbol{\eta}) = P(\boldsymbol{x} \mid \boldsymbol{y}; \Theta) P(\boldsymbol{y} \mid G; \boldsymbol{\beta}, \boldsymbol{\eta})$$

Bayesian inference answers several questions:

1. What is the label (formality) of each dyad?
2. How is formality expressed in language?
3. Are there structural regularities across signed networks?

Variational expectation-maximization

Iterate until convergence:

1. **E-step**: update each edge label in closed form.
2. **M-step: content**: Update $\boldsymbol{\theta}$ in closed form.
3. **M-step: structure** Update $\boldsymbol{\beta}$ and $\boldsymbol{\eta}$ by gradient descent on the expected likelihood.

Variational expectation-maximization

Iterate until convergence:

1. **E-step**: update each edge label in closed form.
2. **M-step: content**: Update $\boldsymbol{\theta}$ in closed form.
3. **M-step: structure** Update $\boldsymbol{\beta}$ and $\boldsymbol{\eta}$ by gradient descent on the expected likelihood.

But! Computing the gradient requires summing over all possible labelings.

- West et al (2014) show that this is NP-Hard.
- We instead optimize $\boldsymbol{\beta}$ and $\boldsymbol{\eta}$ by noise-contrastive estimation.

Let's do it!

- The **Cornell Movie Dialogue corpus** offers 300K conversational turns between 10K dyads, in 617 movies.
- All we need are the address terms...
- But no such resource appears to exist!

Forms of address

Names such as Barack, Barack Hussein Obama.

Titles such as Ms., Dr., Private, Reverend.
Titles can be used for address either by preceding a name (e.g., Colonel Kurtz), or in isolation (e.g., Yes, Colonel.).

Placeholder names such as dude, bro, brother, sweetie, cousin, and asshole.
These terms can be used for address only in isolation.

Subtasks

- Build a vocabulary of titles.
- Build a vocabulary of placeholder names.
- Distinguish address **tokens**:

  His/O name/O is/O **Lebowski/O** *?/O*
  That's/O your/O name/O, **Dude/ADDR**

- Surprisingly little prior work on these problems!

Automatic address annotations

Text:	You	're	Mr.	Lebowski	.
POS:	PRP	VBP	NNP	NNP	.
Address:	O	O	B-ADDR	L-ADDR	O

1. Look for listener's name (mined from rotten tomatoes).
2. Identify NNP tag sequences including those names.
3. Automatically label those sequences as address spans.

Automatic address annotations

Text:	You	're	Mr.	Lebowski	.
POS:	PRP	VBP	NNP	NNP	.
Address:	O	O	B-ADDR	L-ADDR	O

1. Look for listener's name (mined from rotten tomatoes).
2. Identify NNP tag sequences including those names.
3. Automatically label those sequences as address spans.

Automatic address annotations

Text:	You	're	Mr.	Lebowski	.
POS:	PRP	VBP	NNP	NNP	.
Address:	O	O	B-ADDR	L-ADDR	O

1. Look for listener's name (mined from rotten tomatoes).
2. Identify NNP tag sequences including those names.
3. Automatically label those sequences as address spans.

Automatic address annotations

Text:	You	're	Mr.	Lebowski	.
POS:	PRP	VBP	NNP	NNP	.
Address:	O	O	B-ADDR	L-ADDR	O

1. Look for listener's name (mined from rotten tomatoes).
2. Identify NNP tag sequences including those names.
3. Automatically label those sequences as address spans.

Address tagger: features

Feature	Description
Lexical	The word to be tagged, and its two predecessors and successors, $w_{i-2:i+2}$
POS	The part-of-speech of the token to be tagged, and its neighbors
Case	The case of the word to be tagged, and its neighbors.
Constituency parse	First non-NNP ancestor node of the word w_i in the constituent parse tree, and all leaf node siblings in the tree.
Dependency	All dependency relations involving w_i.
Location	Distance of w_i from the start and the end of the sentence or turn.
Punctuation	All punctuation symbols occurring before and after w_i.
Second person pronoun	All forms of the second person pronoun within the sentence.

Address tagger: accuracy

Class	F-measure	Total Instances
B-ADDR	0.800	483
I-ADDR	0.58	53
L-ADDR	0.813	535
U-ADDR	0.987	1864
O-ADDR	0.993	35975

Lexicon induction: titles

- Run the tagger, find terms that frequently appear at the beginning of address spans containing the character's name.
- We then manually filter out 17 of 34 candidates, obtaining:

agent, aunt, captain, colonel, commander, cousin, deputy, detective, dr, herr, inspector, judge, lord, master, mayor, miss, mister, miz, monsieur, mr, mrs, ms, professor, queen, reverend, sergeant, uncle

Lexicon induction: placeholder names

- Remove the CURRENT-WORD feature from the tagger model, then find terms that are frequently tagged as the unique element in an address span.
- After manually filter out 41 of 96 candidates, we obtain:

asshole, babe, baby, boss, boy, bro, bud, buddy, cocksucker, convict, cousin, cowboy, cunt, dad, darling, dear, detective, doll, dude, dummy, father, fella, gal, ho, hon, honey, kid, lad, lady, lover, ma, madam, madame, man, mate, mister, mon, moron, motherfucker, pal, papa, partner, peanut, pet, pilgrim, pop, president, punk, shithead, sir, sire, son, sonny, sport, sucker, sugar, sweetheart, sweetie, tiger

Feature vector construction

Content features

- Addressee name, including any title in the lexicon (e.g., You're **Mr. Lebowski**)
- Any element in the placeholder name lexicon, if tagged as the unique element in an address span (e.g., Thanks, **dude**)

Dyad feature: Adamic-Adar metric for each dyad

Model comparison

	Text	Dyad Feature	Signed triads	Predictive Log-likelihood
M1	✓			
M2	✓		✓	
M3	✓	✓		
M4	✓	✓	✓	

Model comparison

	Text	Dyad Feature	Signed triads	Predictive Log-likelihood
M1	✓			-2133.28
M2	✓		✓	-2018.21
M3	✓	✓		-1884.02
M4	✓	✓	✓	-1582.43

Predictive likelihood is evaluated on held-out address terms for a 10% test fold.

Cluster coherence

"V-cluster"	"T-cluster"
sir	FIRSTNAME
mr+LASTNAME	man
mr+FIRSTNAME	baby
mr	honey
miss+LASTNAME	darling
son	sweetheart
mister+FIRSTNAME	buddy
mrs	sweetie

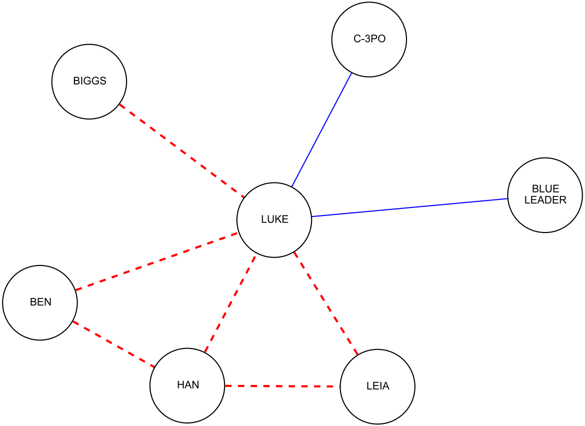
Cluster coherence

"V-cluster"	"T-cluster"
sir	FIRSTNAME
mr+LASTNAME	man
mr+FIRSTNAME	baby
mr	honey
miss+LASTNAME	darling
son	sweetheart
mister+FIRSTNAME	buddy
mrs	sweetie

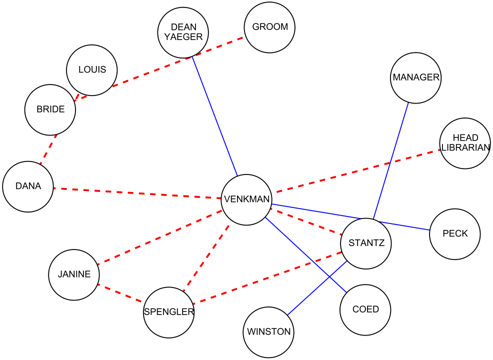
On an intrusion task

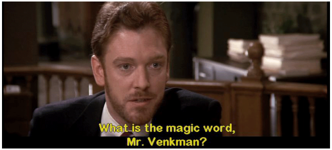
- Raters found the intruder in 73% of cases for the full model (M4).
- ... versus 52% in the text-only model (M1).

Star Wars

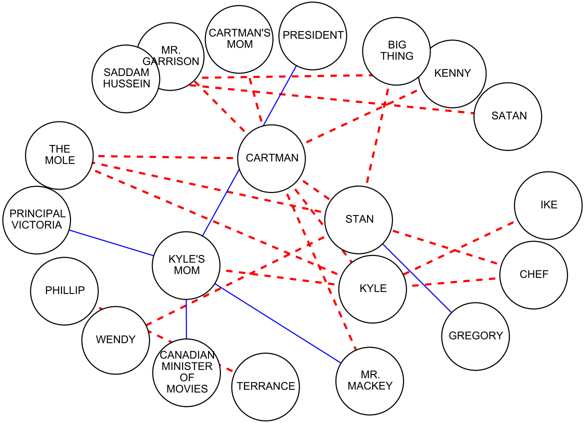


Ghostbusters





South Park

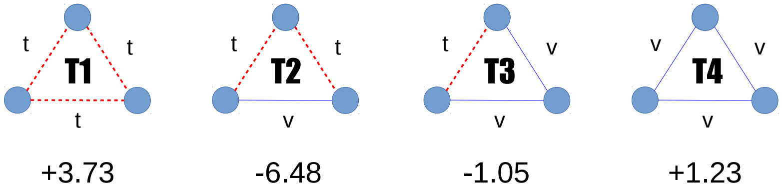


MR. GARRISON
CARTMAN'S MOM
PRESIDENT
BIG THING
KENNY
SADDAM HUSSEIN
SATAN
THE MOLE
CARTMAN
PRINCIPAL VICTORIA
STAN
IKE
KYLE'S MOM
CHEF
PHILLIP
KYLE
WENDY
GREGORY
CANADIAN MINISTER OF MOVIES
TERRANCE
MR. MACKEY

RESPECT MY
AUTHORITY

Network features

- Structural balance theory **stipulates** which triads are stable and unstable.
- In contrast, we are able to induce triadic stability as a parameter of the model:



Summary

Probabilistic sociolinguistic models reveal:

- how social phenomena like politeness and solidarity are conveyed in language;
- which social structures are likely and unlikely.

Summary

Probabilistic sociolinguistic models reveal:

- how social phenomena like politeness and solidarity are conveyed in language;
- which social structures are likely and unlikely.

We apply unsupervised machine learning, but...

- linguistic theory identifies the class of interesting features (address terms);
- social theory identifies the form of the distribution over social structures (triads).

Intractability of Inference

- The normalizing constant z requires summing across all labelings.

  The number of labelings is $\mathcal{O}(\#|\mathcal{Y}|^N)$.

- West et al (2014) show that optimizing an objective over dyads and triads is NP-hard.

  Even exact posterior decoding of $\boldsymbol{y}$ is not tractable, given point estimates for parameters $\boldsymbol{\theta}$, $\boldsymbol{\eta}$, and $\boldsymbol{\beta}$.

- We therefore apply a mean-field approximation.

Variational inference

We iteratively maximize a variational lower bound on the expected likelihood:

$$\begin{aligned}\mathcal{L}_Q = & E_Q[\log P(\boldsymbol{x} \mid \boldsymbol{y}; \boldsymbol{\theta})] \\ & + E_Q[\log P(\boldsymbol{y}; G, \boldsymbol{\beta}, \boldsymbol{\eta})] \\ & - E_Q[\log Q(\boldsymbol{y})].\end{aligned}$$

Variational inference

We iteratively maximize a variational lower bound on the expected likelihood:

$$\begin{aligned}\mathcal{L}_Q = & E_Q[\log P(\boldsymbol{x} \mid \boldsymbol{y}; \boldsymbol{\theta})] \\ & + E_Q[\log P(\boldsymbol{y}; G, \boldsymbol{\beta}, \boldsymbol{\eta})] \\ & - E_Q[\log Q(\boldsymbol{y})].\end{aligned}$$

- The first and third terms factor across dyads.

  This means we have the "obvious" closed form updates for q_{ij} (E-step) and $\boldsymbol{\theta}$ (M-step).

Variational inference

We iteratively maximize a variational lower bound on the expected likelihood:

$$\begin{aligned}\mathcal{L}_Q = & E_Q[\log P(\boldsymbol{x} \mid \boldsymbol{y}; \boldsymbol{\theta})] \\ & + E_Q[\log P(\boldsymbol{y}; G, \boldsymbol{\beta}, \boldsymbol{\eta})] \\ & - E_Q[\log Q(\boldsymbol{y})].\end{aligned}$$

- The first and third terms factor across dyads.

  This means we have the "obvious" closed form updates for q_{ij} (E-step) and $\boldsymbol{\theta}$ (M-step).

- The second term requires an intractable sum over all dyad labelings.

Estimating the prior

- Optimization w.r.t. $\boldsymbol{\beta}$ and $\boldsymbol{\eta}$ requires computing the partition function $Z(\boldsymbol{\eta}, \boldsymbol{\beta}; G)$.
- In **noise-contrastive estimation**, this constant is an additional parameter, to be learned (Gutmann and Hyvärinen, 2012)
- We draw "noise examples", and design an alternative objective, distinguishing the true examples from the noise examples.
- This gives a gradient on $\boldsymbol{\beta}$ and $\boldsymbol{\eta}$, which we optimize through L-BFGS.