Syntactic Dependencies (II)

CS 690N, Spring 2017

Advanced Natural Language Processing

http://people.cs.umass.edu/~brenocon/anlp2017/

Brendan O'Connor

College of Information and Computer Sciences

University of Massachusetts Amherst

Dependency applications

- Dependencies can be used as less sparse alternative to n-grams
  - Sometimes helps, sometimes doesn’t
- Dependency relations can be selected for semantic relationships
- Today: large-scale applications
  - Ad-hoc historical analysis
  - Inference rules via dist. sim.
  - Movie personas, international relations

https://books.google.com/ngrams/

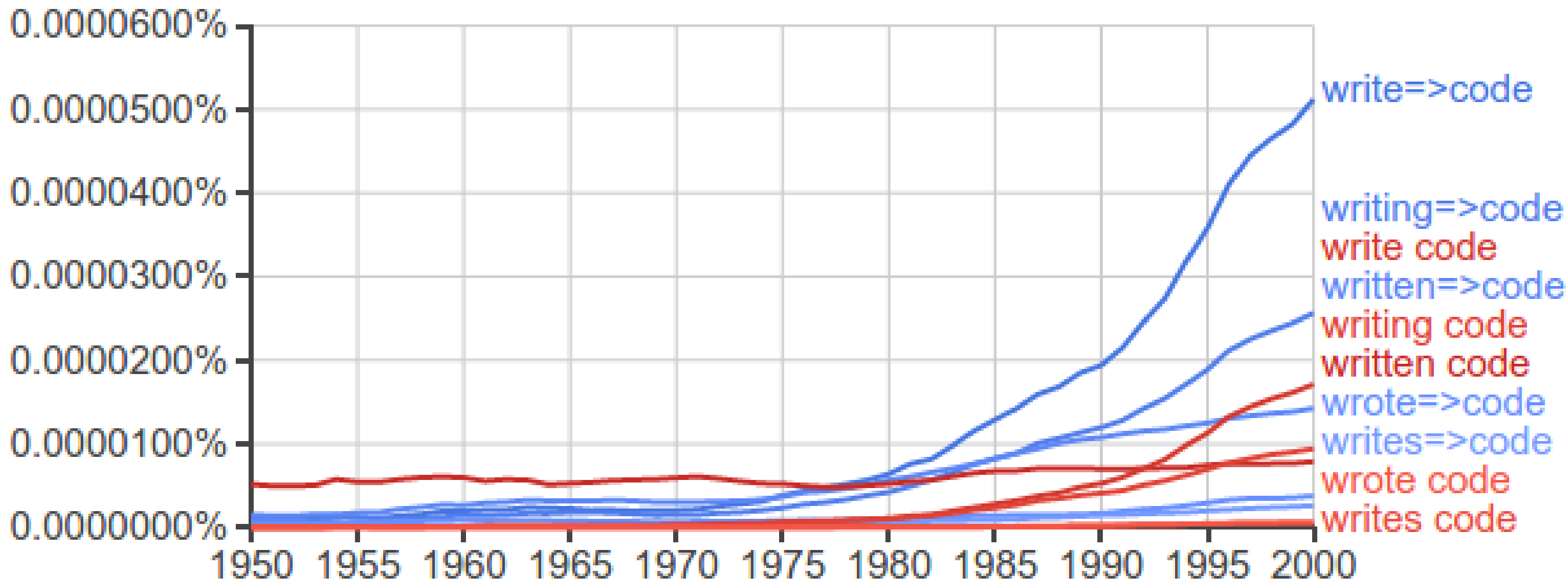


Figure 12.8: Google n-grams results for the bigram *write code* and the dependency arc *write => code* (and their morphological variants)

- History of "writing code"?
- *Goldberg & Orwant 2013*: historical dependencies from google books
- Downloadable! Counts by year.

DIRT (Discovering inference rules from text)

- *[Lin and Pantel 2001]*
- Goal: learn "inference" (paraphrase) rules
  - X is author of Y = X wrote Y
  - X solved Y = X found a solution to Y
  - X caused Y = Y is triggered by X

For example, consider the query to an information retrieval system: "*Who is the author of the 'Star Spangled Banner'?*" Unless the system recognizes the relationship between "*X wrote Y*" and "*X is the author of Y*", it would not necessarily rank the sentence

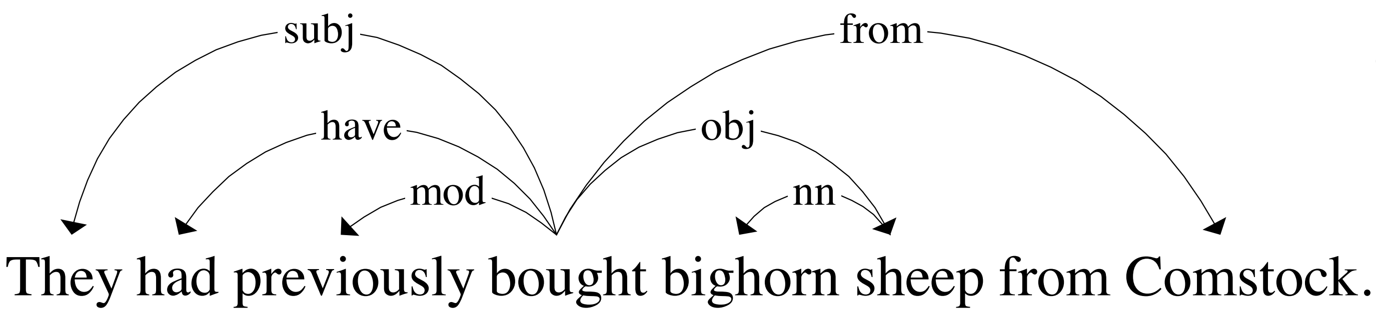
> ... Francis Scott Key wrote the "Star Spangled Banner" in 1814.

higher than the sentence

> …comedian-actress Roseanne Barr sang her famous shrieking rendition of the "Star Spangled Banner" before a San Diego Padres-Cincinnati Reds game.

- Approach
  - Representation: Dependency paths
  - Learning: Distributional similarity

Dependency paths



The paths extracted from this sentence and their meanings are:

(a) N:subj:V←buy→V:from:N
≡ *X* buys something from *Y*

(b) N:subj:V←buy→V:obj:N
≡ *X* buys *Y*

(c) N:subj:V←buy→V:obj:N→sheep→N:nn:N
≡ *X* buys *Y* sheep

(d) N:nn:N←sheep←N:obj:V←buy→V:from:N
≡ *X* sheep is bought from *Y*

(e) N:obj:V←buy→V:from:N
≡ *X* is bought from *Y*

An inverse path is also added for each one above.

- Dep path corresponds to a lexico-syntactic pattern
- Dep path is a chain of relation conjunctions, leaving further modifications unspecified
- Which dep paths to get? Heuristics to alleviate sparsity (L&P require content words, limit path length, etc.)

Distributional similarity

- "You shall know a word by the company it keeps" *[Firth, 1957]*

- Simple single-word (lexical semantics) exmaple:
  "duty" vs "responsibility"
  adj. modification, verbs they're arguments of?

Distributional similarity

- "You shall know a word by the company it keeps" *[Firth, 1957]*

- Simple single-word (lexical semantics) exmaple:
  "duty" vs "responsibility"
  adj. modification, verbs they're arguments of?

  - *duty* can be modified by adjectives such as *additional*, *administrative*, *assigned*, *assumed*, *collective*, *congressional*, *constitutional*, …, so can *responsibility*;

  - *duty* can be the object of verbs such as *accept*, *articulate*, *assert*, *assign*, *assume*, *attend to*, *avoid*, *become*, *breach*, …, so can *responsibility*.

Dist. sim. for dep. paths

Extended Distributional Hypothesis:

If two paths tend to occur in similar contexts, the meanings of the paths tend to be similar.

Table 2. Sample slot fillers for two paths extracted from a newspaper corpus.

"X finds a solution to Y"		*"X solves Y"*	
SLOTX	SLOTY	SLOTX	SLOTY
commission	strike	committee	problem
committee	civil war	clout	crisis
committee	crisis	government	problem
government	crisis	he	mystery
government	problem	she	problem
he	problem	petition	woe
legislator	budget deficit	researcher	mystery
sheriff	dispute	sheriff	murder

Dist. sim. for dep. paths

- Similarity between paths: if they tend to have same words in SlotX and same words in SlotY
- (This paper uses an averaged PMI score for similarity; most work in this area uses cosine similarity)
- Data: "1 GB" news text, 6M paths

Table 3. The top-20 most similar paths to "X solves Y".

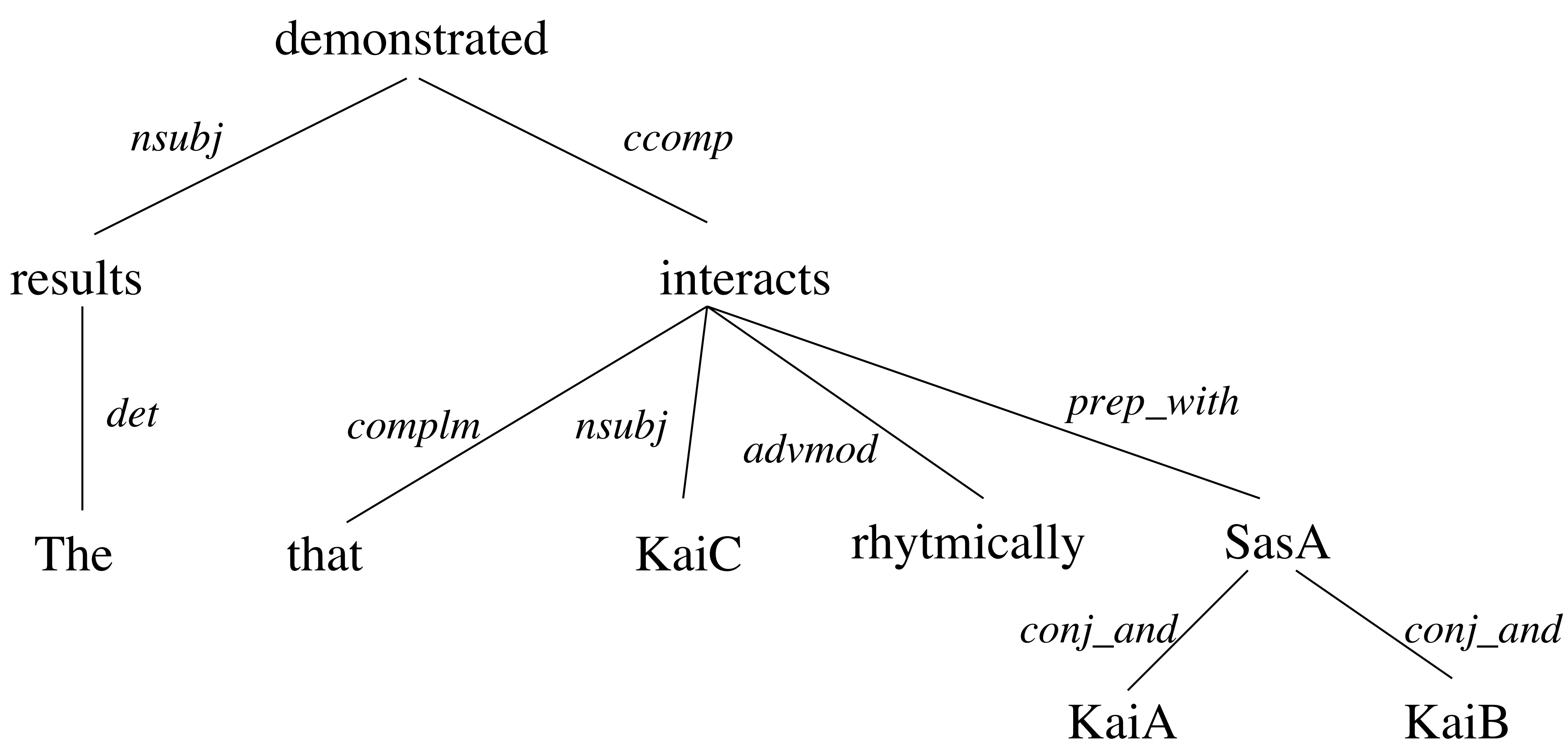
Y is solved by *X*	*Y* is resolved in *X*
X resolves *Y*	*Y* is solved through *X*
X finds a solution to *Y*	*X* rectifies *Y*
X tries to solve *Y*	*X* copes with *Y*
X deals with *Y*	*X* overcomes *Y*
Y is resolved by *X*	*X* eases *Y*
X addresses *Y*	*X* tackles *Y*
X seeks a solution to *Y*	*X* alleviates *Y*
X do something about *Y*	*X* corrects *Y*
X solution to *Y*	*X* is a solution to *Y*

Table 5. Evaluation of Top-40 most similar paths.

Q#	PATHS	MAN.	DIRT	INT.	ACC.
Q_1	X is author of Y	7	21	2	52.5%
Q_2	X is monetary value of Y	6	0	0	N/A
Q_3	X manufactures Y	13	37	4	92.5%
Q_4	X spend Y	7	16	2	40.0%
	spend X on Y	8	15	3	37.5%
Q_5	X is managing director of Y	5	14	1	35.0%
Q_6	X asks Y	2	23	0	57.5%
	asks X for Y	2	14	0	35.0%
	X asks for Y	3	21	3	52.5%

(Manual judgments...)

- Erkan et al.: protein-protein interactions



- Next: movie personas and international relations
- Approach
  - Representation: Dependency paths
  - Learning: Topic models over dep. paths (Bayesian admixtures)

Learning Latent Personas of Film Characters

David Bamman, Brendan O'Connor and Noah Smith

School of Computer Science
Carnegie Mellon University

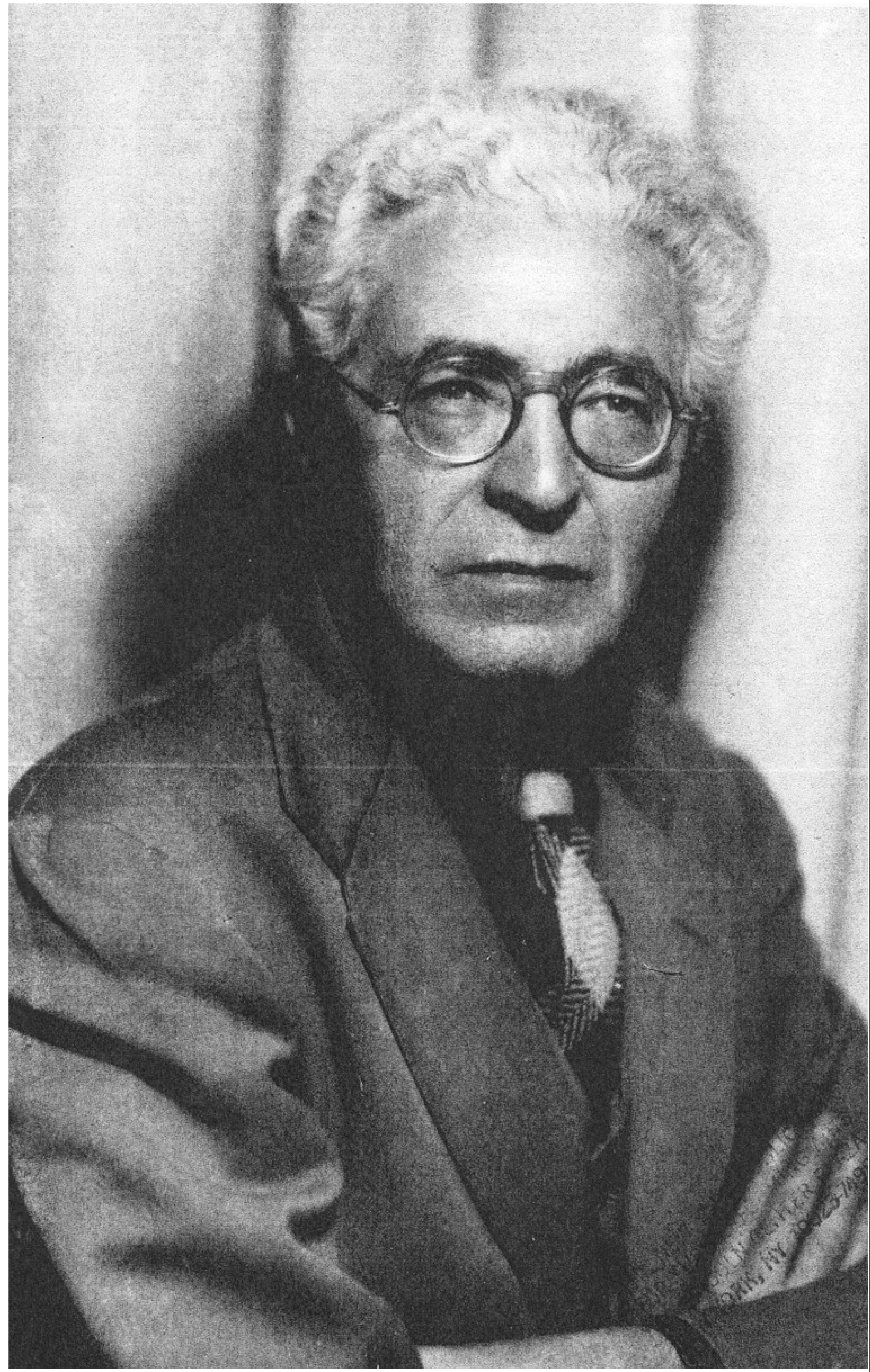
Association for Computational Linguistics, 2013

"The Plot, then, is the first principle, and, as it were, the soul of a tragedy: Character holds the second place."

Poetics I.VI
Aristotle, 335 BCE

"Aristotle was mistaken … Character was a great factor in Aristotle's time, and no fine play ever was or ever will be written without it"

The Art of Dramatic Writing
Lajos Egri, 1946



PLOT

- Procedural scripts
  - Schank and Abelson 1977, Regneri et al. 2010
- Narrative chains
  - Chambers and Jurafsky 2008
- Plot structure
  - Finlayson 2011, Elsner 2012, McIntyre and Lapata 2010, Goyal et al. 2010

CHARACTER

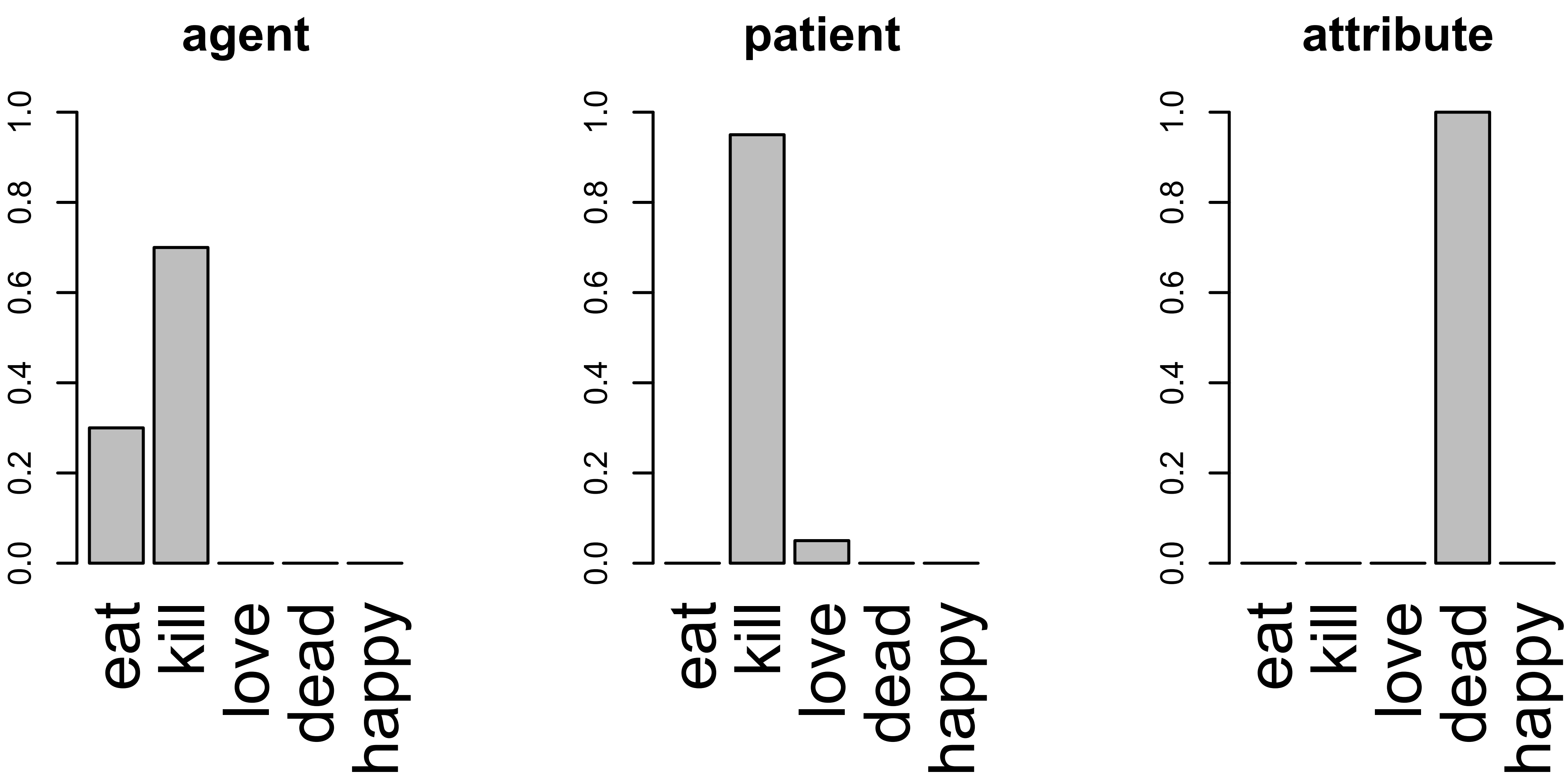
- Chambers and Jurafsky (2009), Regneri et al. (2011)
- Entity-centric coreference (Haghighi and Klein 2010)
- Semantic role induction (Titov and Klementiev 2012)

THE VILLAIN

Text features:

- Does: kill, hunt, severs, chokes
- Has done to him: fights, defeats, refuses
- Is described as: evil, frustrated, lord

PERSONA



agent
patient
attribute
0.0
0.2
0.4
0.6
0.8
1.0
eat
kill
love
dead
happy

DATA

43,959 plot summaries extracted from English-language Wikipedia

- Stanford CoreNLP to tag, parse, extract named entities, resolve coref
- Linguistic features extracted from the typed dependency tuples:
    - Agent = nsubj or agent
    - Patient = dobj, nsubjpass or iobj
    - Attribute = nsubj/appos governors, nsubj, appos, amod, nn dependents of entity mentions

Freebase metadata

- Detailed genre (365 non-mutually exclusive categories)
- Character/actor alignments
    - Gender
    - Age at time of movie's release

NLP Pipeline

Step 1: noun phrase coreference => entities

The young **[Luke Skywalker]1** is a farmer ...

[Luke]1 watches as [**Vader]2** kills [**Kenobi]3**.

Then [**Luke]1** runs away, while soldiers shoot at **[him]1**.

While saving [**Leia]4** [**he]1** figures out [...], and **[she]4** says [...].

Step 2: dependency path extraction

attribute attribute

The *young* **[Luke Skywalker]1** is a *farmer* ...

agent-of-verb agent-of-verb patient-of-verb

[Luke]1 *watches* as [**Vader]2** *kills* [**Kenobi]3**.

agent-of-verb patient-of-verb

Then [**Luke]1** *runs* away, while soldiers *shoot* at **[him]1**.

agent-of-verb agent-of-verb agent-of-verb

While *saving* [**Leia]4** [**he]1** *figures* out ..., and **[she]4** *wants* ....

patient-of-verb

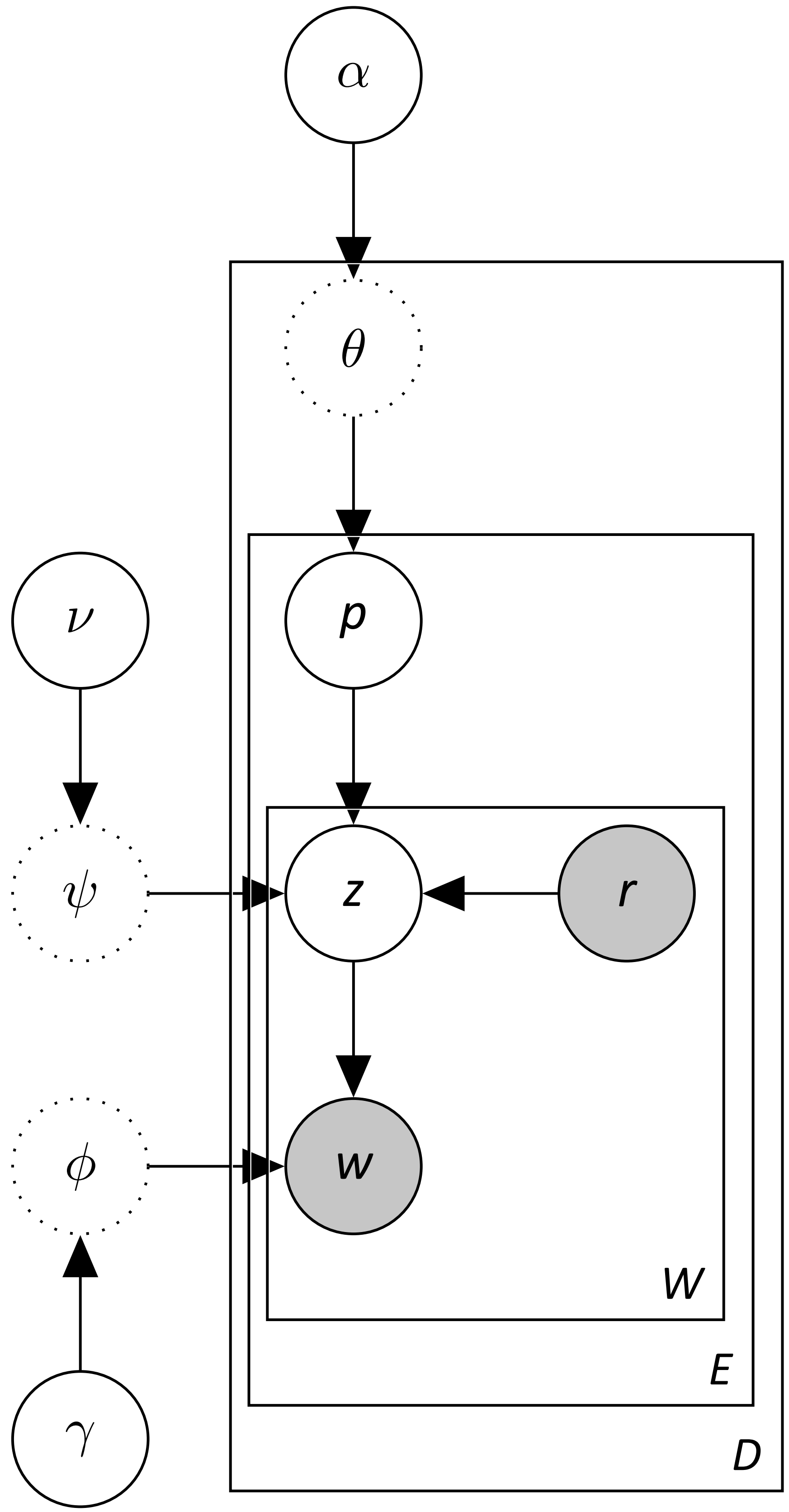
Want: for every entity,
bag of (rel, word) pairs
(Unary, not binary, relations)

Rule-based semantic relation normalization from syntactic deps

- **Agent verbs.** Verbs for which the entity is an agent argument (*nsubj* or *agent*).
- **Patient verbs.** Verbs for which the entity is the patient, theme or other argument (*dobj*, *nsubjpass*, *iobj*, or any prepositional argument *prep_**).
- **Attributes.** Adjectives and common noun words that relate to the mention as adjectival modifiers, noun-noun compounds, appositives, or copulas (*nsubj* or *appos* governors, or *nsubj, appos, amod, nn* dependents of an entity mention).

more recent, open-source rule-based postprocessors: PropS, PredPatt

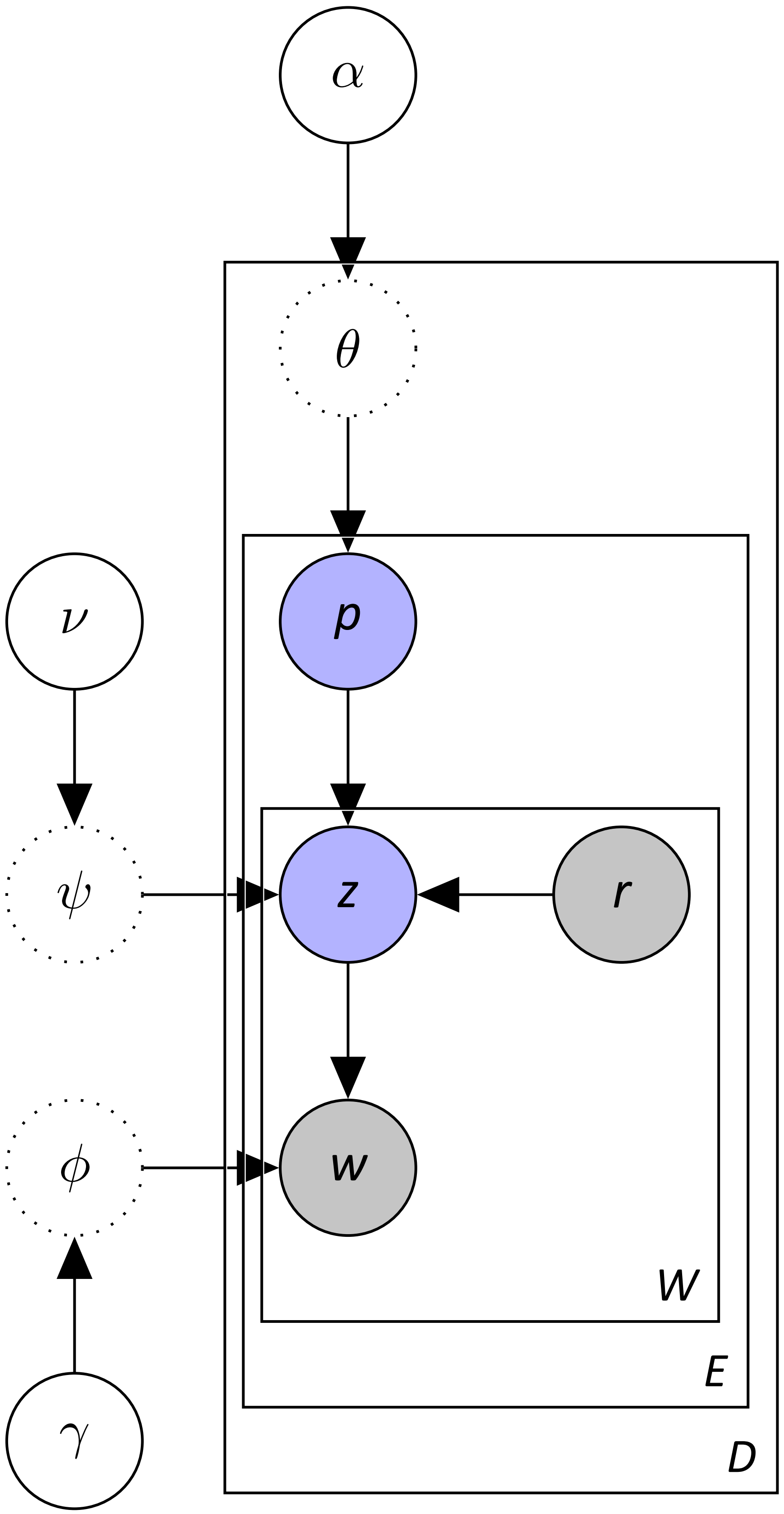
DIRICHLET PERSONA MODEL



- θ document-persona mixture: $\sim \text{Dir}(\alpha)$
- *p* persona: $\sim \text{Cat}(\theta)$
- $\psi_{p,r}$ persona-topic mixture: $\sim \text{Dir}(\nu_r)$
- *r* observed word type
- *z* word class: $\sim \text{Cat}(\psi_{p,r})$
- ϕ_z topic-word mixture: $\sim \text{Dir}(\gamma)$
- *w* word token: $\sim \text{Cat}(\phi_z)$
- *W* plate: multiple words
- *E* plate: multiple entities
- *D* plate: multiple plot summaries

Inference: CGS with slice sampling on the priors.

DIRICHLET PERSONA MODEL



- θ document-persona mixture: $\sim \text{Dir}(\alpha)$
- *p* persona: $\sim \text{Cat}(\theta)$
- $\psi_{p,r}$ persona-topic mixture: $\sim \text{Dir}(\nu_r)$
- *r* observed word type
- *z* word class: $\sim \text{Cat}(\psi_{p,r})$
- ϕ_z topic-word mixture: $\sim \text{Dir}(\gamma)$
- *w* word token: $\sim \text{Cat}(\phi_z)$
- *W* plate: multiple words
- *E* plate: multiple entities
- *D* plate: multiple plot summaries

Inference: CGS with slice sampling on the priors.

EVALUATION I: NAMES

Gold clusters: characters with the same name

- Sequels
- Remakes

Noise: “Street thug”

970 unique character names used twice in data; n=2,666.

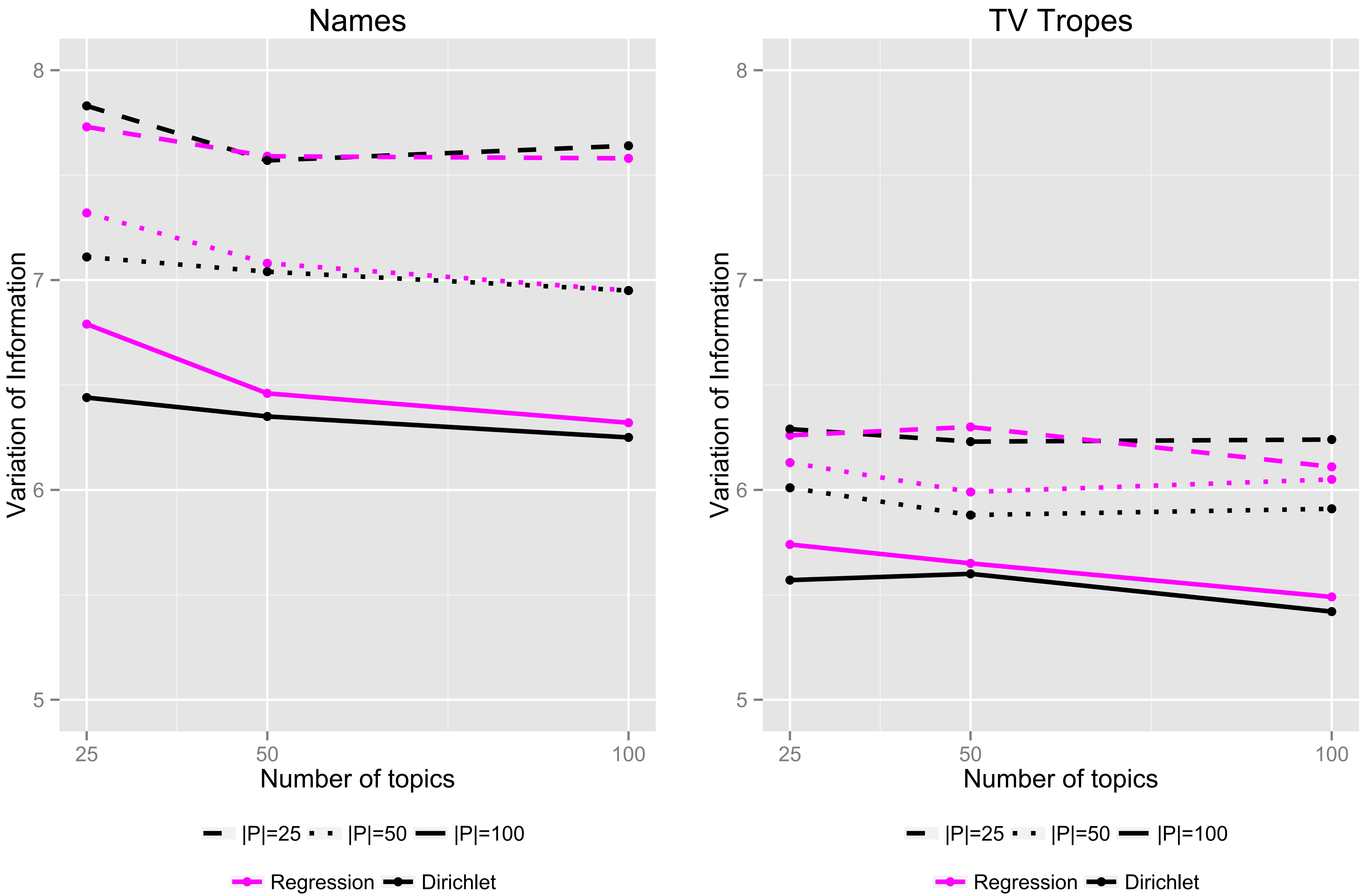
EVALUATION II: TV TROPES

Gold clusters: manually clustered characters from `www.tvtropes.com`

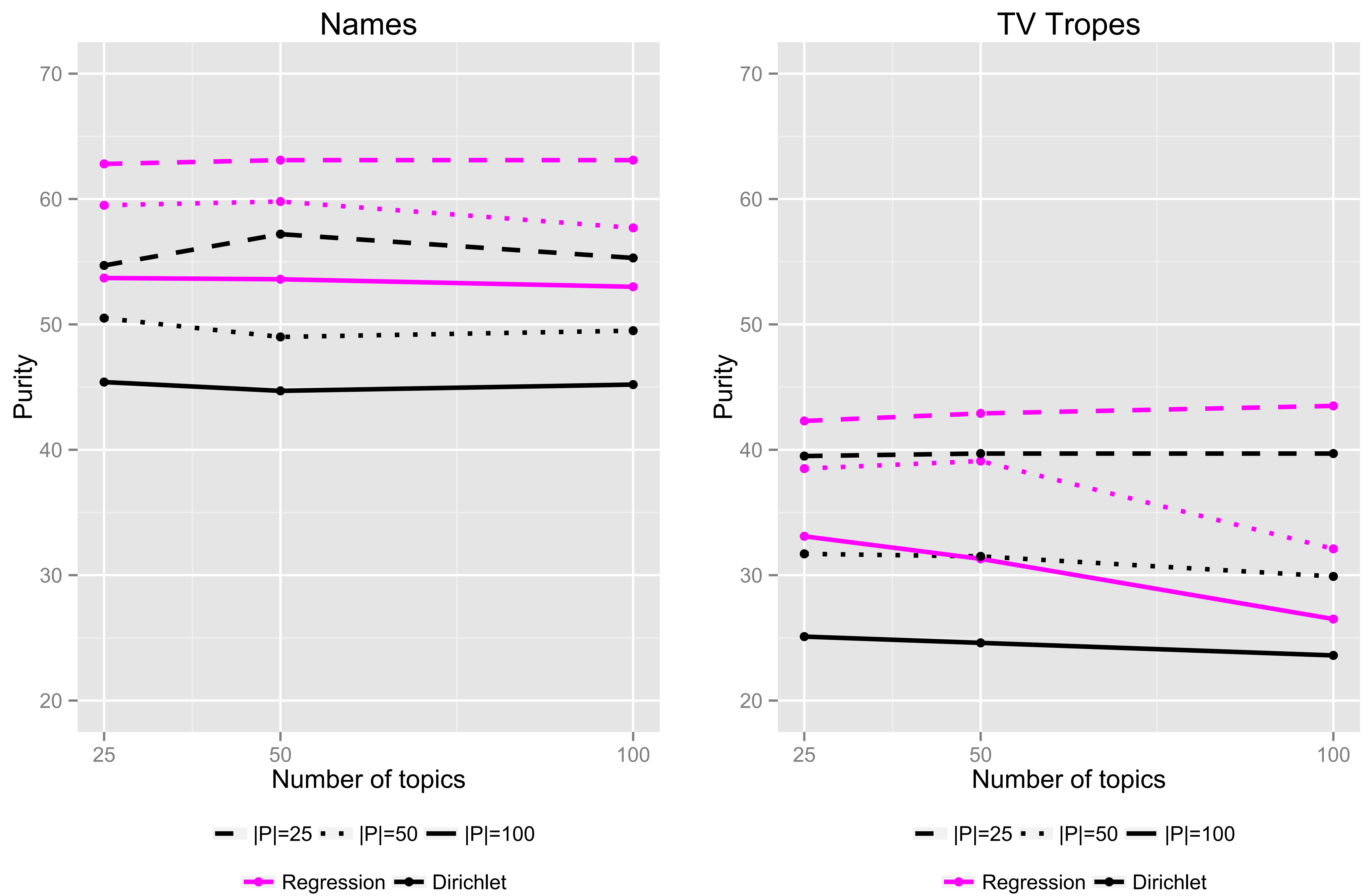
- “The Surfer Dude”
- “Arrogant Kung-Fu Guy”
- “Hardboiled Detective”
- “The Klutz”
- “The Valley Girl

72 character tropes containing 501 characters

VARIATION OF INFORMATION

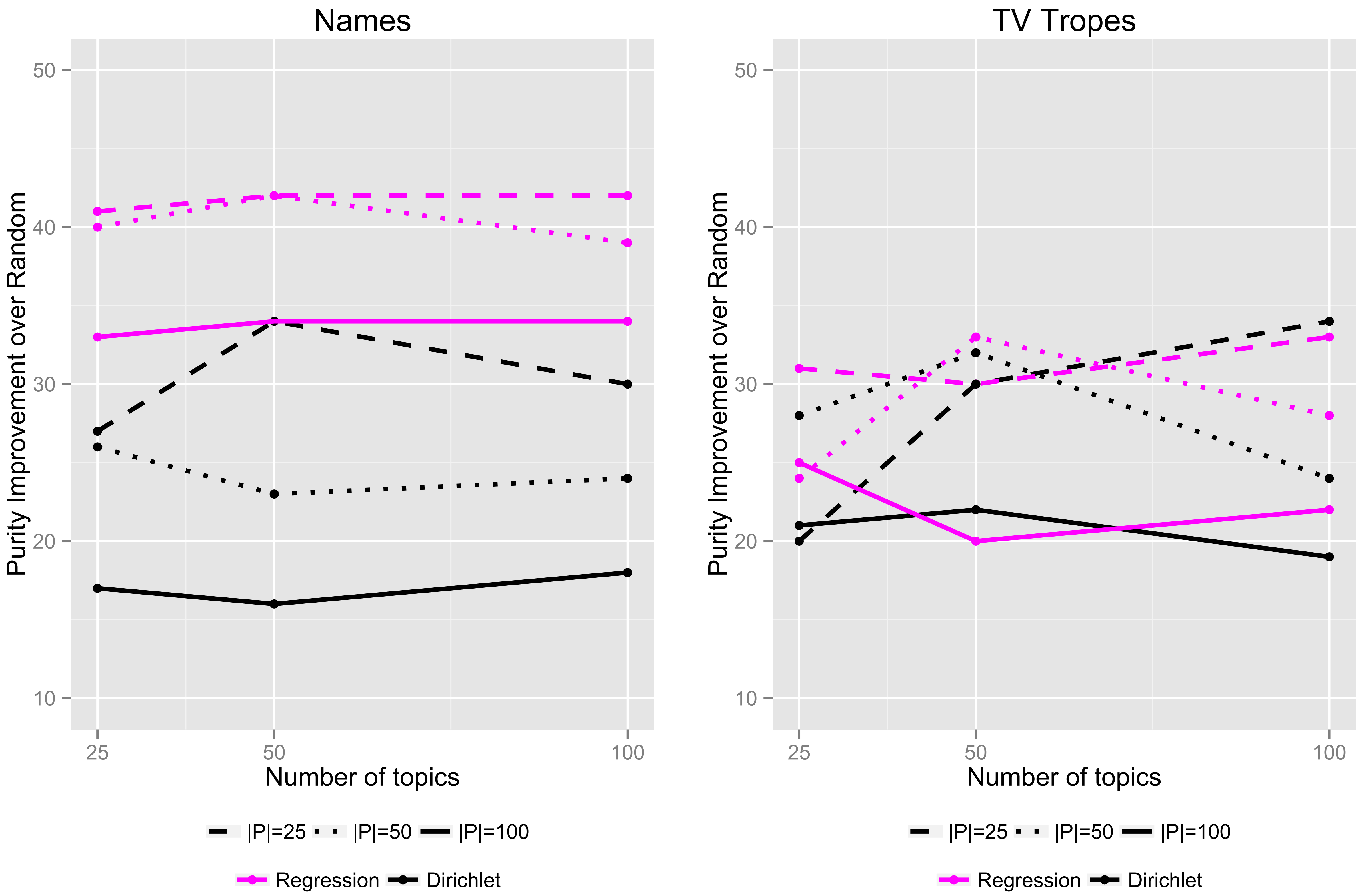


PURITY



Names
TV Tropes
Purity
70
60
50
40
30
20
25
50
100
Number of topics
|P|=25
|P|=50
|P|=100
Regression
Dirichlet

PURITY IMPROVEMENT



TOPICS ϕ

- unite marry woo elope court
- purchase sign sell owe buy
- shoot aim overpower interrogate kill
- explore investigate uncover deduce
- woman friend wife sister husband
- witch villager kid boy mom
- reply say mention answer shout
- pop lift crawl laugh shake
- sing perform cast produce dance
- approve die suffer forbid collapse
- werewolf mother parent killer father
- decapitate bite impale strangle stalk
- invade sail travel land explore

PERSONAS

- dark major henchman warrior sergeant
- shoot aim overpower interrogate kill

- Action
- Male
- War Film

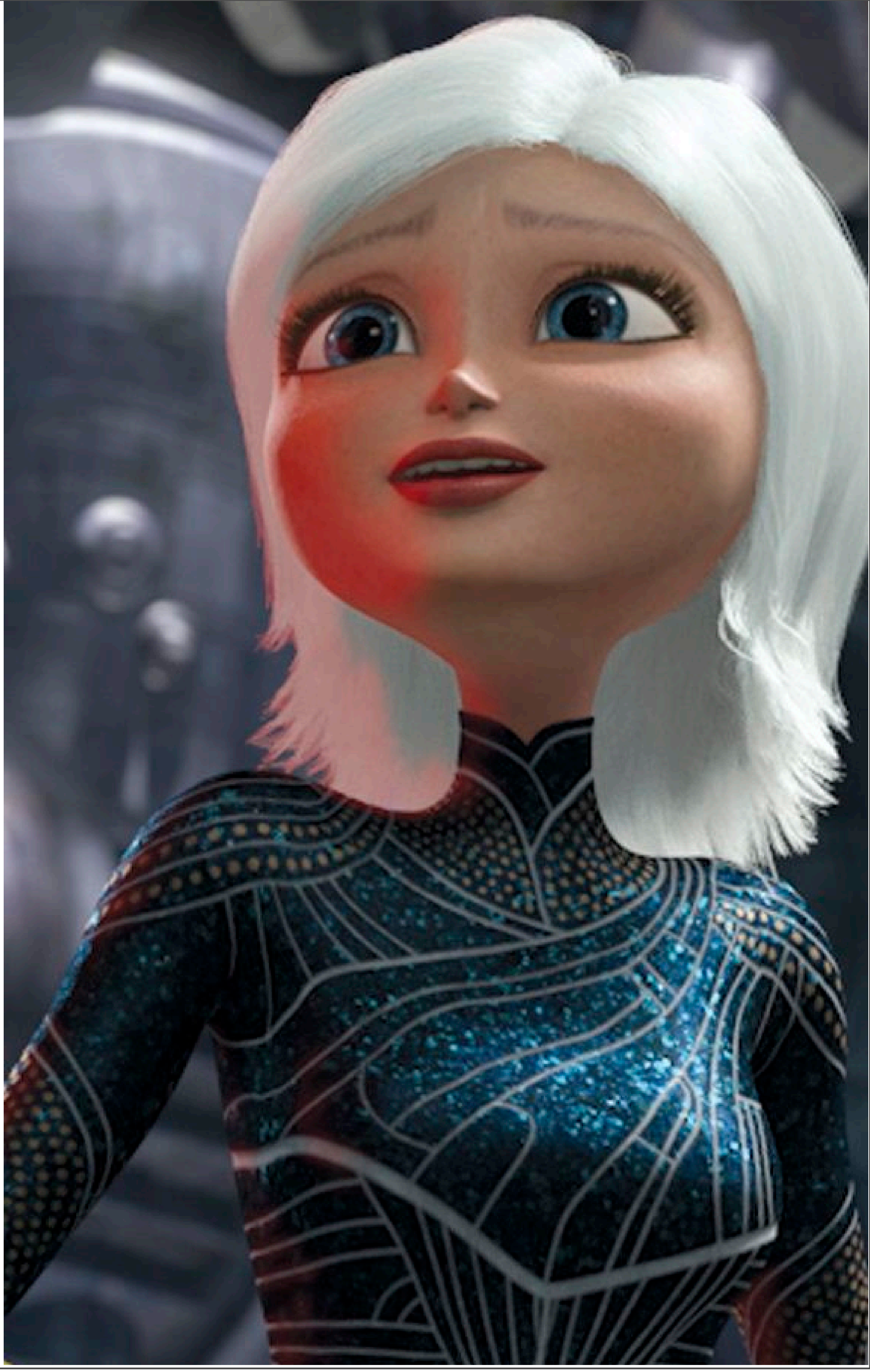
- Jason Bourne (*Bourne Supremacy*)
- Jack Traven (*Speed*)
- Jean-Claude (*Taken*)

PERSONAS

- capture corner transport imprison trap
- infiltrate deduce leap evade obtain
- flee escape swim hide manage

- Female
- Action
- Adventure

- Ginormica (*Monsters vs. Aliens*)
- Aang (*The Last Airbender*),
- Carly (*Transformers*)



PERSONAS

- reply say mention answer shout
- talk tell reassure assure calm
- flirt reconcile date dance forgive

- Female
- Comedy
- Romance Film

- Graham (*The Holiday*)
- Abby Richter (*The Ugly Truth*)
- Anna Scott (*Notting Hill*)

Personas

- http://www.cs.cmu.edu/~ark/personas/
- Unsupervised learning, discovery, and validation

- stopped here 3/23

Event data through knowledge engineering

[Schrodt 1994, Leetaru and Schrodt 2013]

Event classes
(~200)

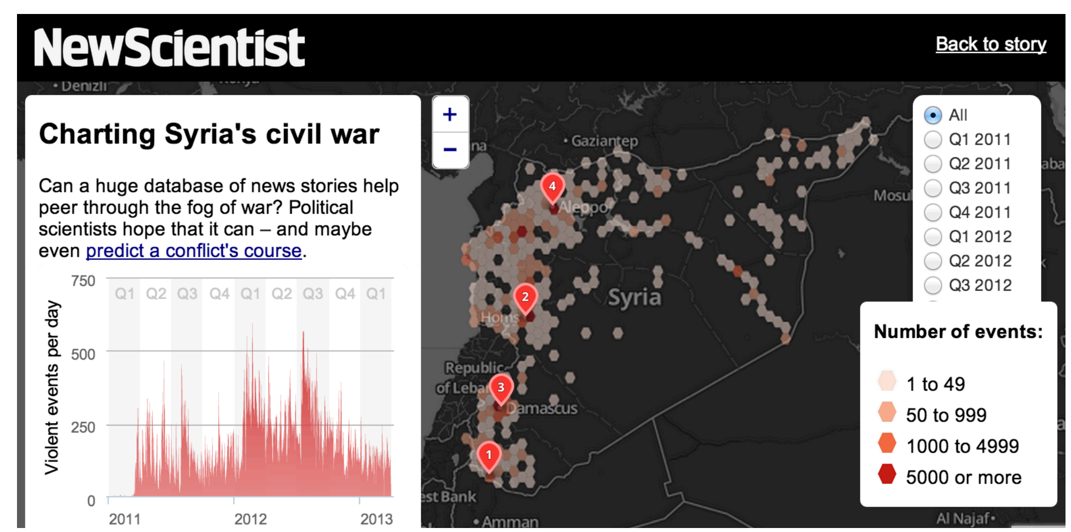
03 - EXPRESS INTENT TO COOPERATE
07 - PROVIDE AID
15 - EXHIBIT MILITARY POSTURE

Dictionary:
Verb patterns per event class
(~15000)

191 - Impose blockade, restrict movement

not_ allow to_ enter ;mj 02 aug 2006
barred travel
block traffic from ;ab 17 nov 2005
block road ;hux 1/7/98

Extract events from news text



Issue: Hard to maintain and adapt to new domains

Our approach

[O'Connor, Stewart, and Smith, *Proc. of ACL*, 2013]

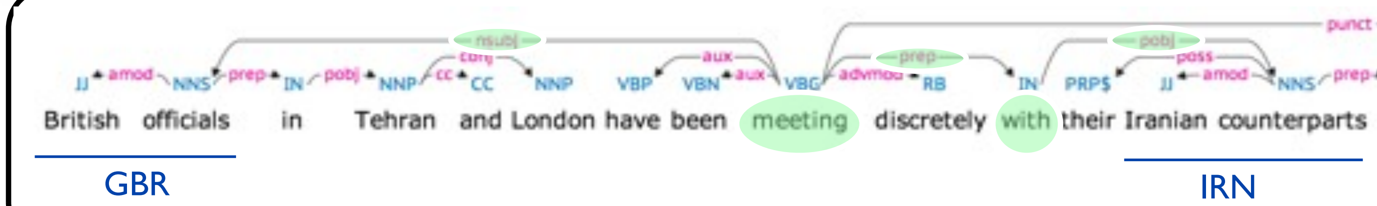
Data: twenty years of news articles

Our approach

[O'Connor, Stewart, and Smith, *Proc. of ACL*, 2013]

Data: twenty years of news articles

Natural Language Processing



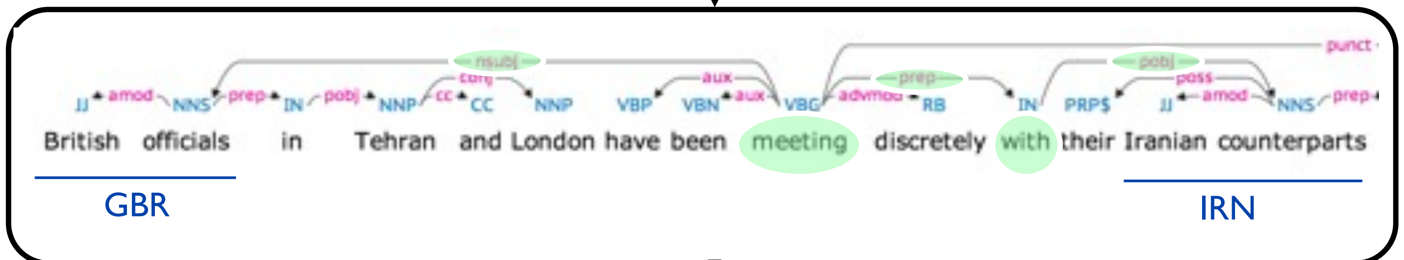
Event phrases of actor interactions

Our approach

[O'Connor, Stewart, and Smith, *Proc. of ACL*, 2013]

Data: twenty years of news articles

Natural Language Processing



Event phrases of actor interactions

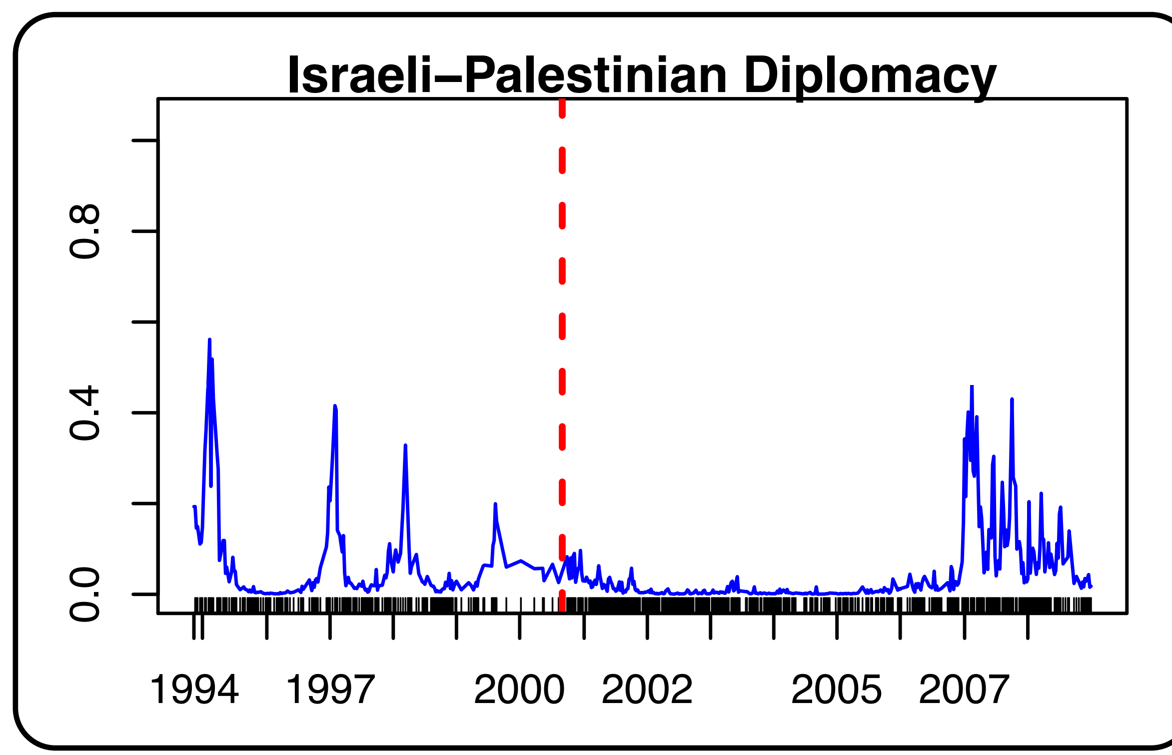
Probabilistic Graphical Model
Purely from textual data, jointly learns both

(1) **Event class dictionaries**

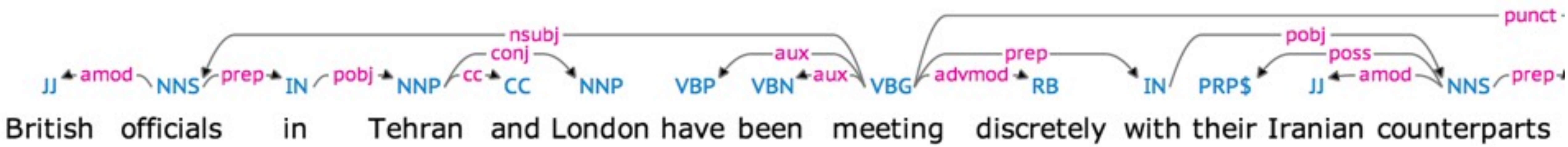
(2) **Political dynamics**

"diplomacy"	arrive in, visit, meet with, travel to, leave, hold with, meet, meet in, fly to, be in, arrive for talk with, say in, arrive with, head to, hold in, due in, leave for, make to, arrive to,
"verbal conflict"	accuse, blame, say, break with, sever with, blame on, warn, call, attack, rule with, charge, say←ccomp come from, say ←ccomp, suspect, slam, accuse government ←poss,
"material conflict"	kill in, have troops in, die in, be in, wound in, have soldier in, hold in, kill in attack in, remain in, detain in, have in, capture in, stay in, about ←pobj troops in, kill, have troops

↔



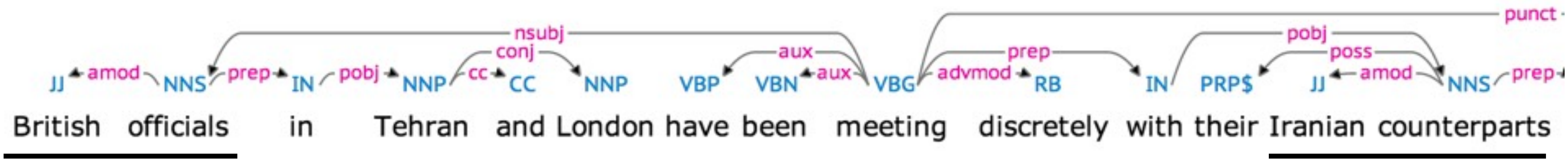
Event Extraction:
Who did what to whom?



Source *(s)*:
Recipient *(r)*:
Event phrase *(w)*:

[e.g. *Dowty 1991*]

Event Extraction:
Who did what to whom?



GBR

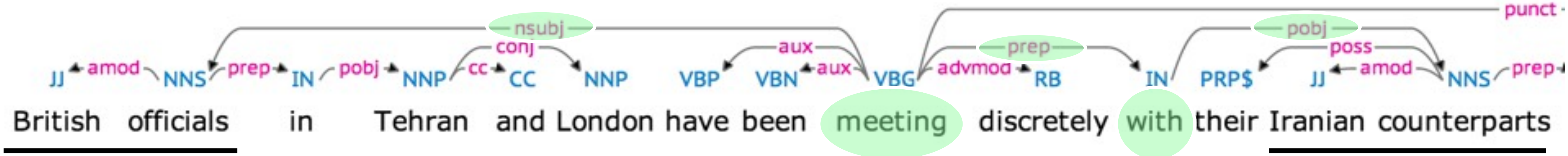
IRN

Match
country name list

Source *(s)*:
Recipient *(r)*:
Event phrase *(w)*:

[e.g. *Dowty 1991*]

Event Extraction:
Who did what to whom?



GBR

IRN

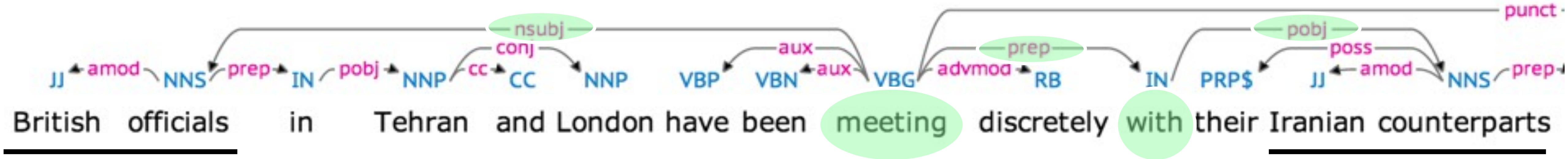
Match
country name list

Extract
event phrase

Source *(s)*:
Recipient *(r)*:
Event phrase *(w)*:

[e.g. *Dowty 1991*]

Event Extraction:
Who did what to whom?



GBR

IRN

Match
country name list

Extract
event phrase

Source (*s*): GBR

Recipient (*r*): IRN

Event phrase (*w*): *<--nsubj--* meet *--prep-->* with *--pobj-->*

[e.g. *Dowty 1991*]

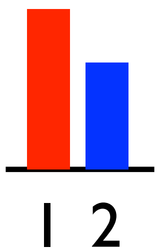
"*X* meets with *Y*"

Event Extraction:
Who did what to whom?

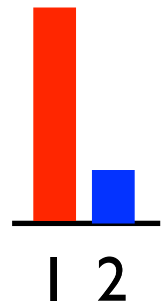
- Structured linguistic analysis pipeline
  - Document classifier
  - Part-of-speech tagging
  - Syntactic parsing (rare in text-as-data) (*CoreNLP*)
  - POS and parse filtering rules
    - Factivity, verb paths, and parse quality

Contextual event class probabilities

s=ISR, *r*=PSE

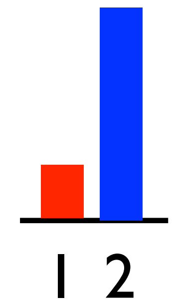
$\theta_{s,r,t} =$ 

<u>*t*= Jul 15-21, 2002</u>
say <-ccomp be to
release to
take control of
occupy
wound in
scuffle with
be <-xcomp meet
meet with
meet with
arrest

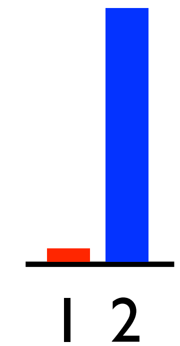
$\theta_{s,r,t} =$ 

<u>*t*= Jul 3-9, 2006</u>
commit to
strike
carry in
continue in
reject
fire at target in
start around
ratchet pressure on
shell
hit

s=USA, *r*=FRA

$\theta_{s,r,t} =$ 

<u>*t*= Feb 2-8, 1998</u>
travel <-xcomp meet with
consider
meet with
meet with
meet with

$\theta_{s,r,t} =$ 

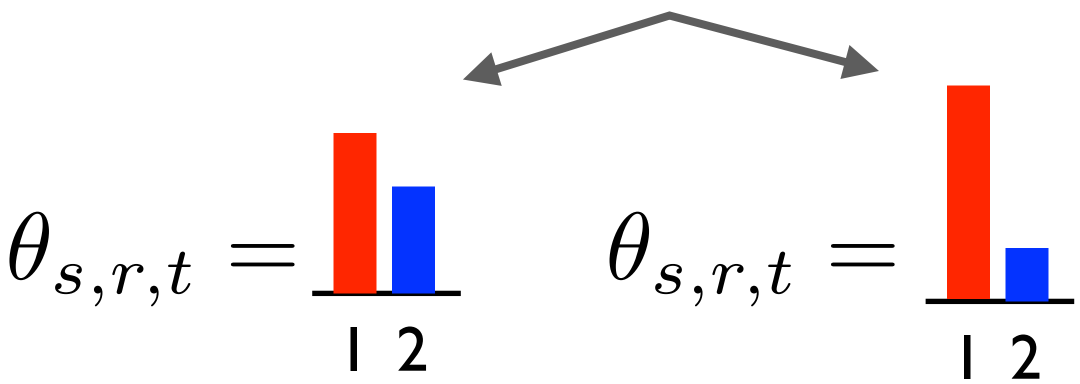
<u>*t*= Dec 22-28, 2003</u>
release with
welcome
welcome by
win
agree with
indict
win from
concern over
win
indict

Event class dictionaries ϕ_1 ϕ_2

agree with, arrest, be <-xcomp meet, carry in, commit to, concern over, consider, continue in, fire at target in, hit, indict, meet with, occupy, ratchet pressure on, reject, release to, release with, say <-ccomp be to, scuffle with, shell, start around, strike, take control of, travel <-xcomp meet with, welcome, welcome by, win, win from, wound in

Contextual event class probabilities

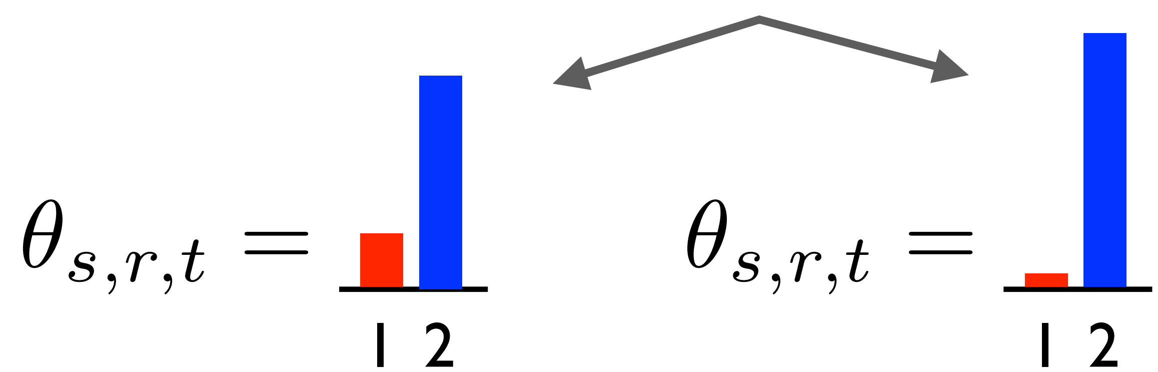
s=ISR, r=PSE



t= Jul 15-21, 2002
say <-ccomp be to
release to
take control of
occupy
wound in
scuffle with
be <-xcomp meet
meet with
meet with
arrest

t= Jul 3-9, 2006
commit to
strike
carry in
continue in
reject
fire at target in
start around
ratchet pressure on
shell
hit

s=USA, r=FRA



t= Feb 2-8, 1998
travel <-xcomp meet with
consider
meet with
meet with
meet with

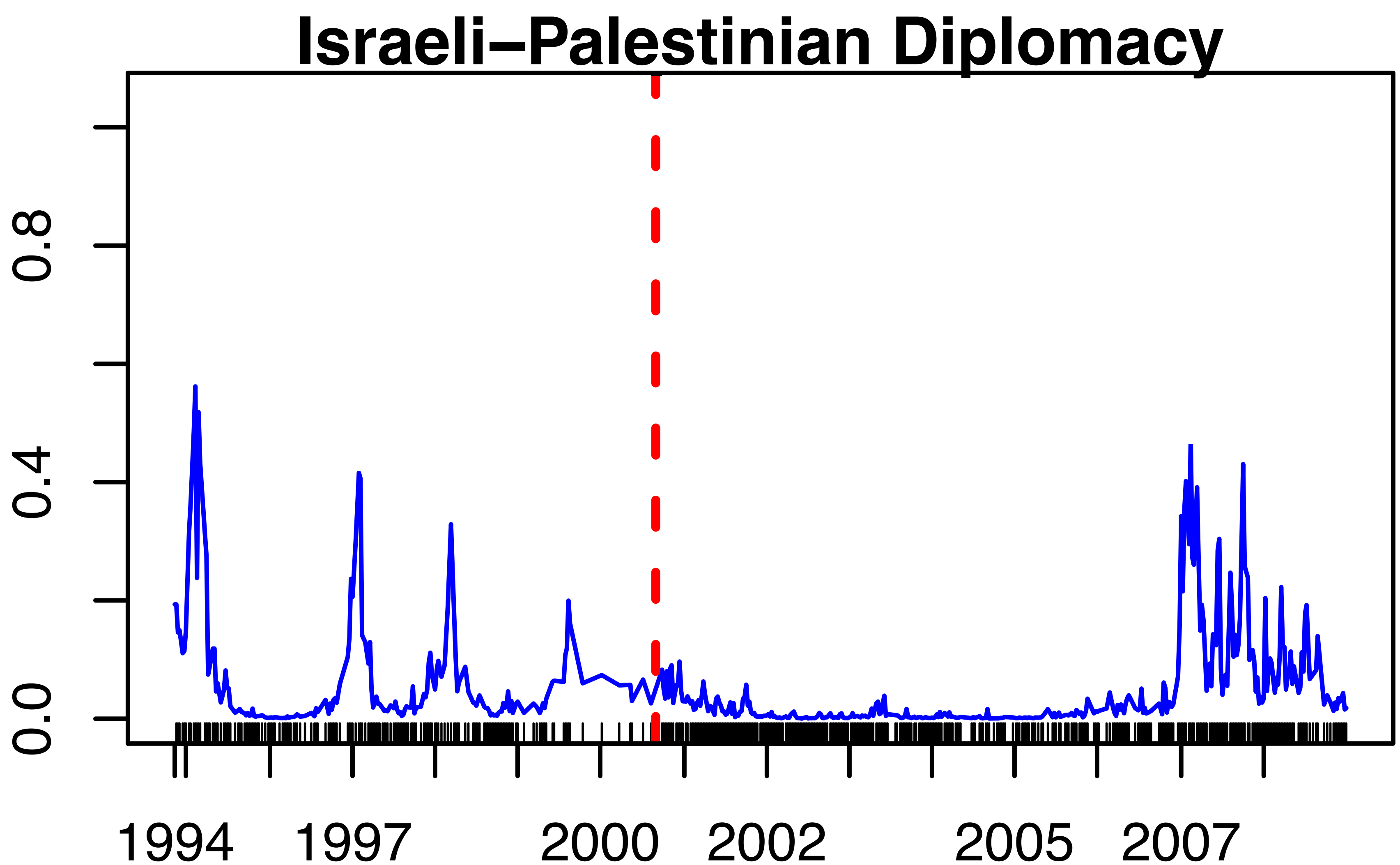
t= Dec 22-28, 2003
release with
welcome
welcome by
win
agree with
indict
win from
concern over
win
indict

Event class dictionaries ϕ_1 ϕ_2

agree with, arrest, be <-xcomp meet, carry in, commit to, concern over, consider, continue in, fire at target in, hit, indict, meet with, occupy, ratchet pressure on, reject, release to, release with, say <-ccomp be to, scuffle with, shell, start around, strike, take control of, travel <-xcomp meet with, welcome, welcome by, win, win from, wound in

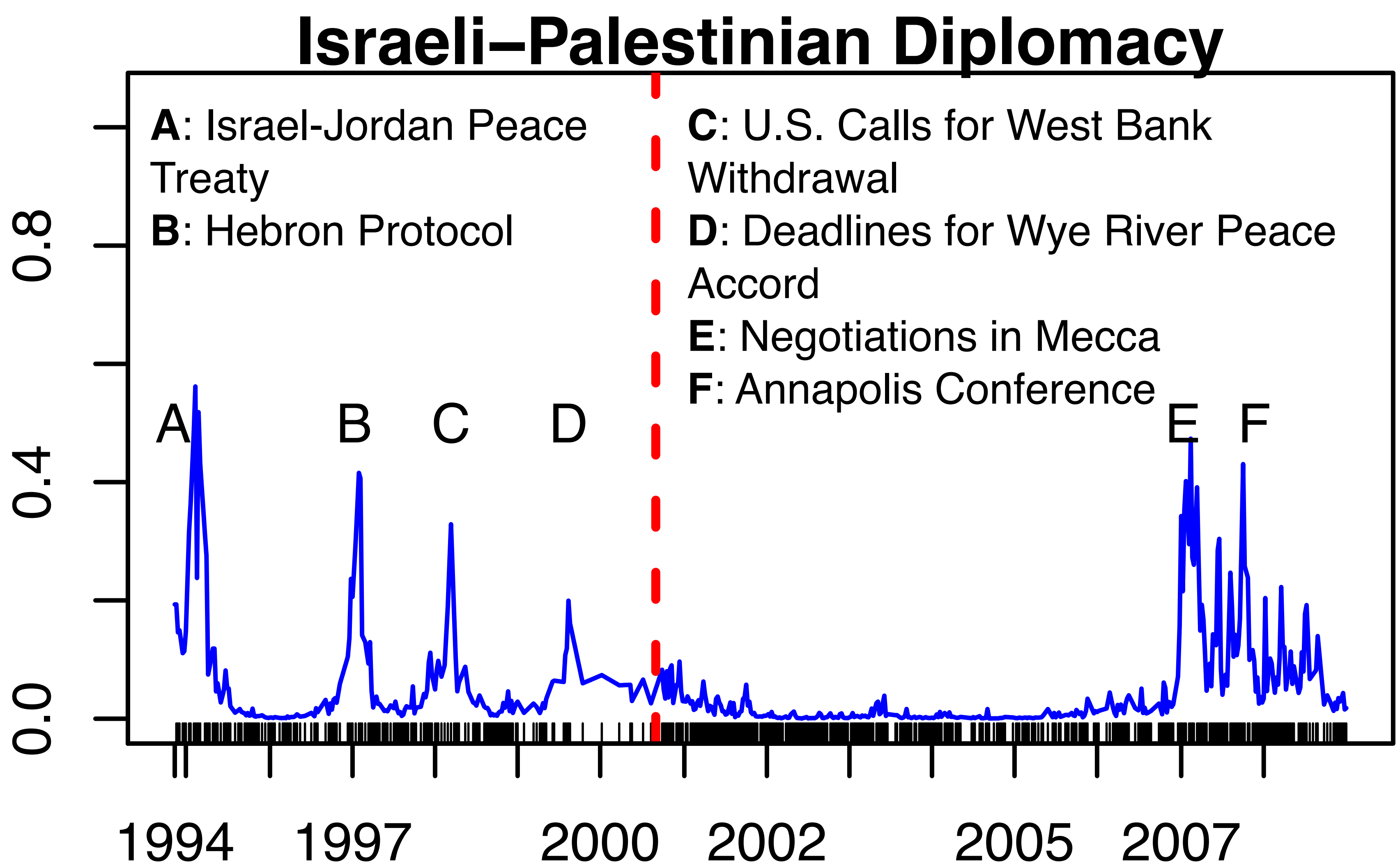
Case study

meet with, sign with, praise, say with, arrive in, host, tell, welcome, join, thank, meet, travel to, criticize, leave, take to, begin to, begin with, summon, reach with, hold with

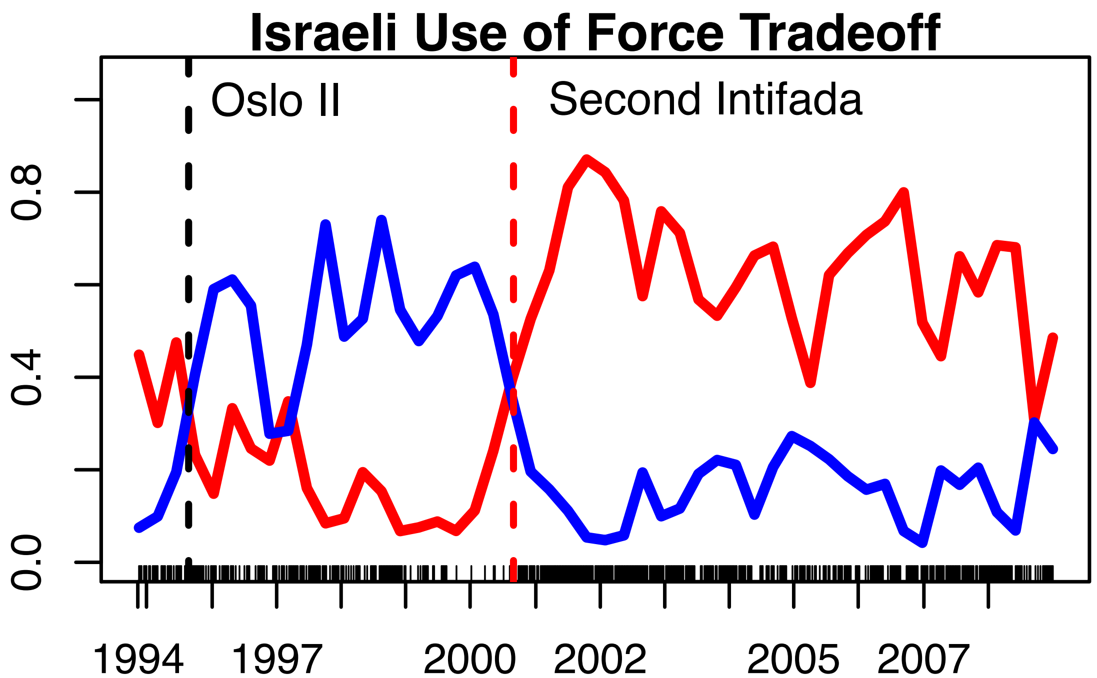


Case study

meet with, sign with, praise, say with, arrive in, host, tell, welcome, join, thank, meet, travel to, criticize, leave, take to, begin to, begin with, summon, reach with, hold with



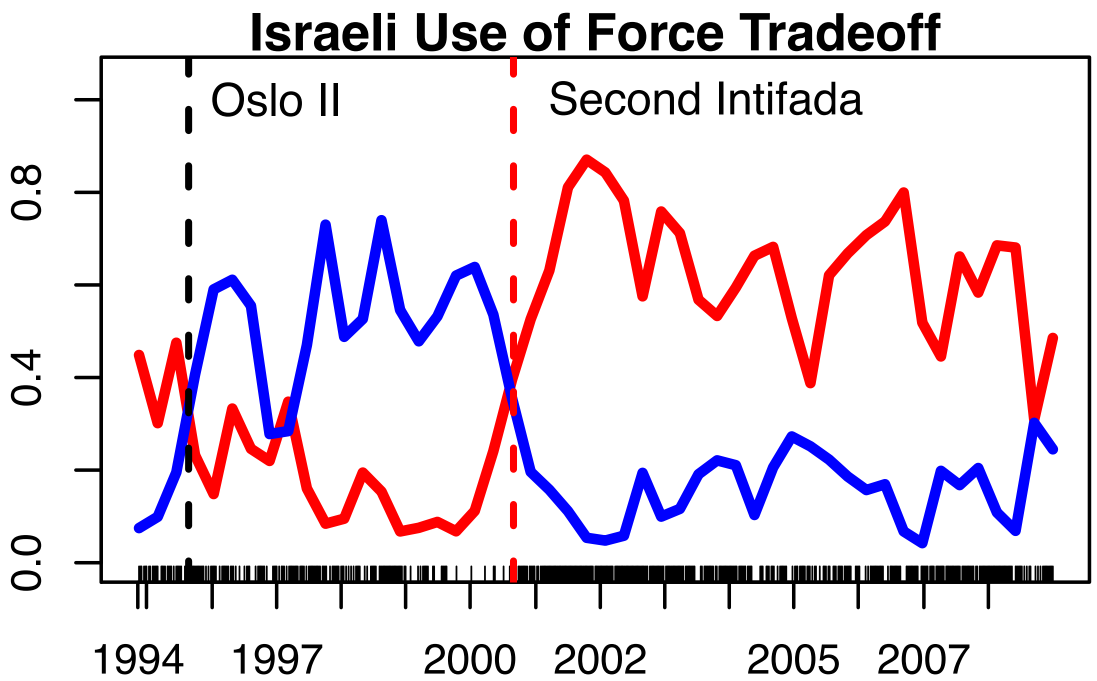
Validation of unsupervised models...



impose on, seal, capture from, seize from, arrest, ease closure of, close, deport, close with, release

kill, fire at, enter, kill in, attack, raid, strike in, move into, pound, bomb

Validation of unsupervised models...



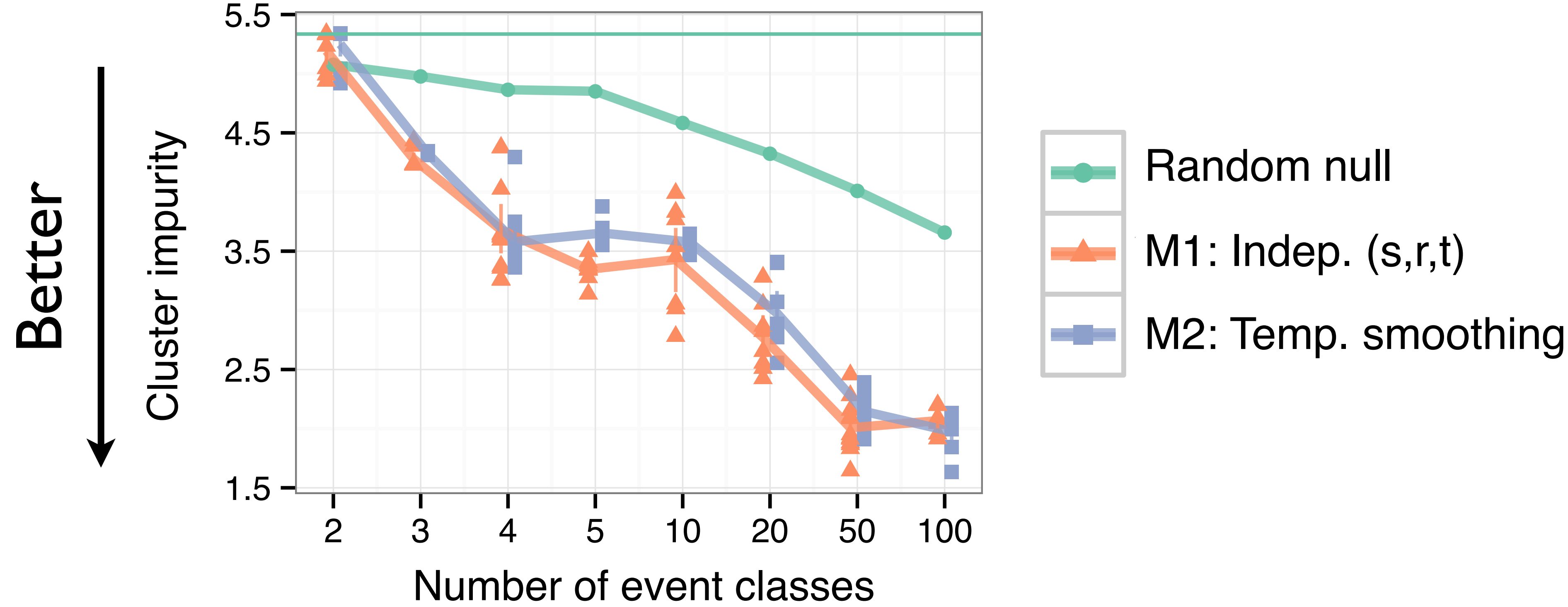
impose on, seal, capture from, seize from, arrest, ease closure of, close, deport, close with, release

kill, fire at, enter, kill in, attack, raid, strike in, move into, pound, bomb

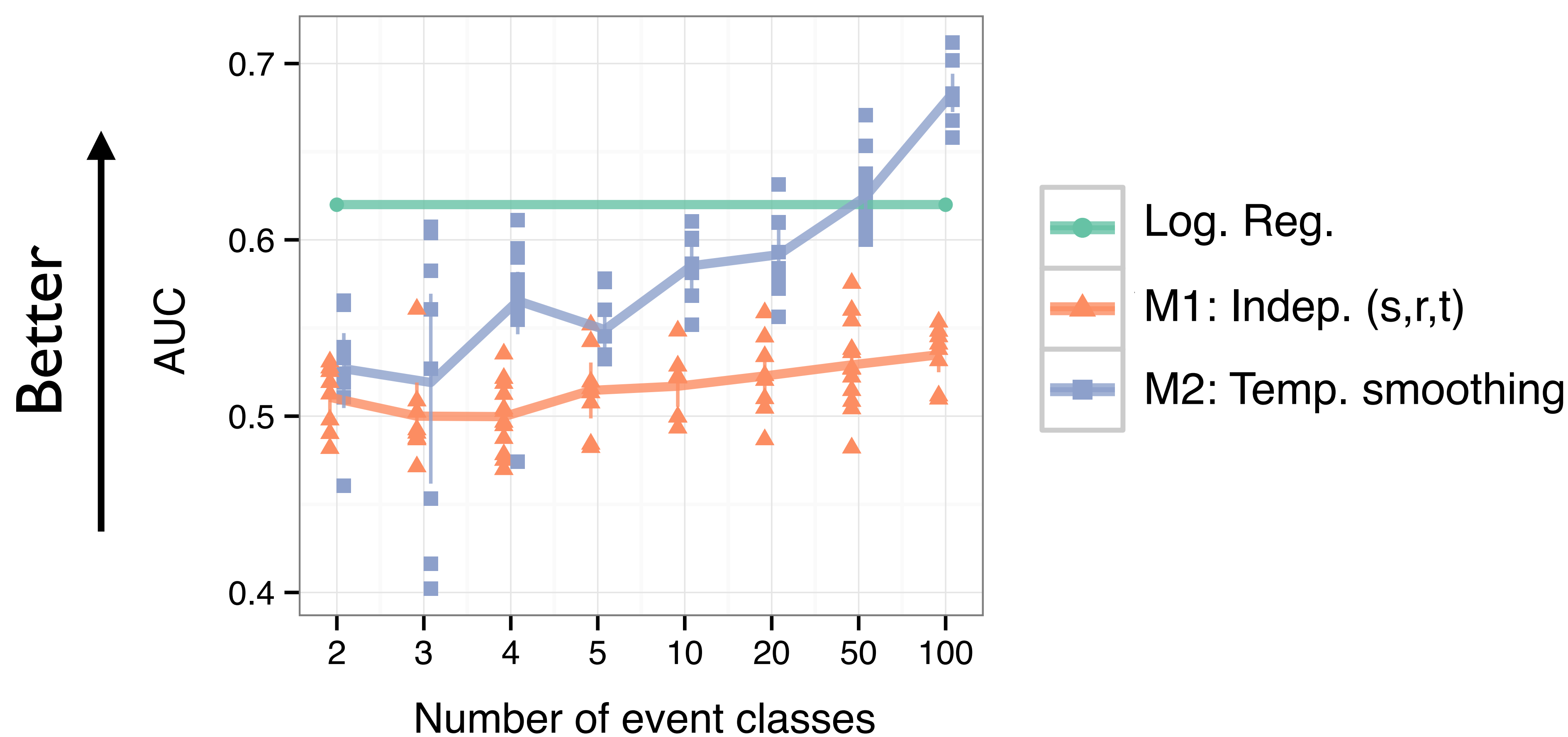
Correlates to conflict?

Semantic coherence?

Evaluations



Lexicon / Ontology reconstruction



Real-world conflict reconstruction

Feature type	Left window	NE1	Middle	NE2	Right window
Lexical	[]	PER	[was/VERB born/VERB in/CLOSED]	LOC	[]
Lexical	[Astronomer]	PER	[was/VERB born/VERB in/CLOSED]	LOC	[,]
Lexical	[#PAD#, Astronomer]	PER	[was/VERB born/VERB in/CLOSED]	LOC	[, Missouri]
Syntactic	[]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[]
Syntactic	[Edwin Hubble $\Downarrow_{lex-mod}$]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[]
Syntactic	[Astronomer $\Downarrow_{lex-mod}$]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[]
Syntactic	[]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[$\Downarrow_{lex-mod}$,]
Syntactic	[Edwin Hubble $\Downarrow_{lex-mod}$]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[$\Downarrow_{lex-mod}$,]
Syntactic	[Astronomer $\Downarrow_{lex-mod}$]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[$\Downarrow_{lex-mod}$,]
Syntactic	[]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[$\Downarrow_{inside}$ Missouri]
Syntactic	[Edwin Hubble $\Downarrow_{lex-mod}$]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[$\Downarrow_{inside}$ Missouri]
Syntactic	[Astronomer $\Downarrow_{lex-mod}$]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[$\Downarrow_{inside}$ Missouri]

Table 3: Features for 'Astronomer Edwin Hubble was born in Marshfield, Missouri'.

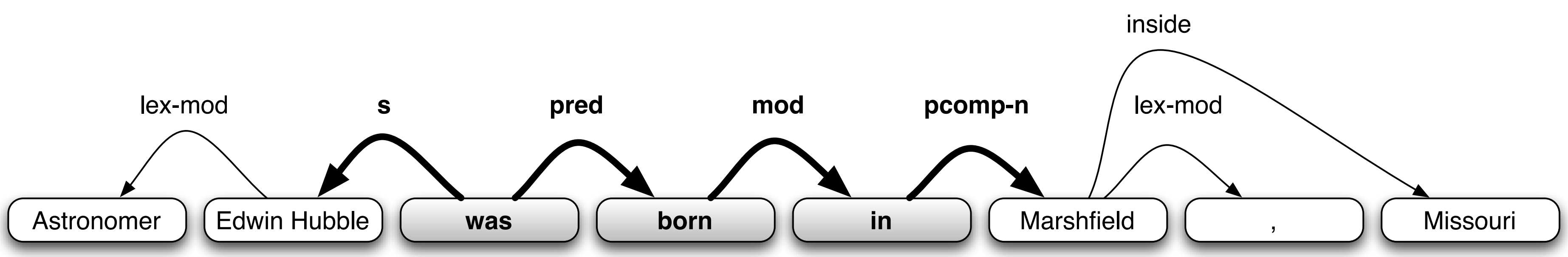


Figure 1: Dependency parse with dependency path from 'Edwin Hubble' to 'Marshfield' highlighted in boldface.

http://web.stanford.edu/~jurafsky/mintz.pdf

- Adjective and adverb modifiers in reviews for economic analysis

http://crowdsourcing-class.org/readings/downloads/nlp/opinion-mining-using-econometrics.pdf

http://pages.stern.nyu.edu/~aghose/kdd2007.pdf