Lecture 10. Exploring Pattern Mining Applications

Lecture 10. Exploring Pattern Mining Applications

- Frequent Pattern Mining for Text Data—Phrase Mining and Topic Modeling
  - Strategy 1: Simultaneously Inferring Phrases and Topics
  - Strategy 2: Post Topic Modeling Phrase Construction
  - Strategy 3: First Phrase Mining then Topic Modeling (ToPMine)

Thanks to Ahmed El-Kishky@UIUC, Chi Wang@MSR and Marina Danilevsky@IBM for their contributions

Note: Only one application is discussed here—Other applications will be discussed in Lecture 11 or have already been scattered in other Lectures

Session 1. Frequent Pattern Mining for Text Data

Frequent Pattern Mining for Text Data: Phrase Mining and Topic Modeling

- Motivation: Unigrams (single words) can be difficult to interpret
- Ex.: The topic that represents the area of Machine Learning

learning
reinforcement
support
machine
vector
selection
feature
random
:

versus

learning
support vector machines
reinforcement learning
feature selection
conditional random fields
classification
decision trees
:

Various Strategies: Phrase-Based Topic Modeling

- Strategy 1: Generate bag-of-words → generate sequence of tokens
  - Bigram topical model [Wallach'06], topical n-gram model [Wang, et al.'07], phrase discovering topic model [Lindsey, et al.'12]
- Strategy 2: Post bag-of-words model inference, visualize topics with n-grams
  - Label topic [Mei et al.'07], TurboTopic [Blei & Lafferty'09], KERT [Danilevsky, et al.'14]
- Strategy 3: Prior bag-of-words model inference, mine phrases and impose on the bag-of-words model
  - ToPMine [El-kishky, et al.'15]

Session 2. Strategy 1: Simultaneously Inferring Phrases and Topics

Strategy 1: Simultaneously Inferring Phrases and Topics

- Bigram Topic Model **[Wallach'06]**
  - Probabilistic generative model that conditions on previous word and topic when drawing next word
- Topical N-Grams (TNG) **[Wang, et al.'07]**
  - Probabilistic model that generates words in textual order
  - Create n-grams by concatenating successive bigrams (a generalization of Bigram Topic Model)
- Phrase-Discovering LDA (PDLDA) **[Lindsey, et al.'12]**
  - Viewing each sentence as a time-series of words, PDLDA posits that the generative parameter (topic) changes periodically
  - Each word is drawn based on previous m words (context) and current phrase topic
- High model complexity: Tends to overfitting; High inference cost: Slow

TNG: Experiments on Research Papers

Reinforcement Learning			Human Receptive System		
LDA	n-gram (2+)	n-gram (1)	LDA	n-gram (2+)	n-gram (1)
state	reinforcement learning	action	motion	receptive field	motion
learning	optimal policy	policy	visual	spatial frequency	spatial
policy	dynamic programming	reinforcement	field	temporal frequency	visual
action	optimal control	states	position	visual motion	receptive
reinforcement	function approximator	actions	figure	motion energy	response
states	prioritized sweeping	function	direction	tuning curves	direction
time	finite-state controller	optimal	fields	horizontal cells	cells
optimal	learning system	learning	eye	motion detection	figure
actions	reinforcement learning rl	reward	location	preferred direction	stimulus
function	function approximators	control	retina	visual processing	velocity
algorithm	markov decision problems	agent	receptive	area mt	contrast
reward	markov decision processes	q-learning	velocity	visual cortex	tuning
step	local search	goal	vision	light intensity	moving
dynamic	state-action pair	space	moving	directional selectivity	model
control	markov decision process	step	system	high contrast	temporal
sutton	belief states	environment	flow	motion detectors	responses
rl	stochastic policy	system	edge	spatial phase	orientation
decision	action selection	problem	center	moving stimuli	light
algorithms	upright position	steps	light	decision strategy	stimuli
agent	reinforcement learning methods	transition	local	visual stimuli	cell

TNG: Experiments on Research Papers (2)

Speech Recognition			Support Vector Machines		
LDA	n-gram (2+)	n-gram (1)	LDA	n-gram (2+)	n-gram (1)
recognition	speech recognition	speech	kernel	support vectors	kernel
system	training data	word	linear	test error	training
word	neural network	training	vector	support vector machines	support
face	error rates	system	support	training error	margin
context	neural net	recognition	set	feature space	svm
character	hidden markov model	hmm	nonlinear	training examples	solution
hmm	feature vectors	speaker	data	decision function	kernels
based	continuous speech	performance	algorithm	cost functions	regularization
frame	training procedure	phoneme	space	test inputs	adaboost
segmentation	continuous speech recognition	acoustic	pca	kkt conditions	test
training	gamma filter	words	function	leave-one-out procedure	data
characters	hidden control	context	problem	soft margin	generalization
set	speech production	systems	margin	bayesian transduction	examples
probabilities	neural nets	frame	vectors	training patterns	cost
features	input representation	trained	solution	training points	convex
faces	output layers	sequence	training	maximum margin	algorithm
words	training algorithm	phonetic	svm	strictly convex	working
frames	test set	speakers	kernels	regularization operators	feature
database	speech frames	mlp	matrix	base classifiers	sv
mlp	speaker dependent	hybrid	machines	convex optimization	functions

Session 3. Strategy 2: Post Topic Modeling Phrase Construction

Strategy 2: Post Topic Modeling Phrase Construction

- **TurboTopics** [Blei & Lafferty'09] – Phrase construction as a post-processing step to Latent Dirichlet Allocation
  - Perform Latent Dirichlet Allocation on corpus to assign each token a topic label
  - Merge adjacent unigrams with the same topic label by a distribution-free permutation test on arbitrary-length back-off model
  - End recursive merging when all significant adjacent unigrams have been merged
- **KERT** [Danilevsky et al.'14] – Phrase construction as a post-processing step to Latent Dirichlet Allocation
  - Perform **frequent pattern mining** on each topic
  - Perform **phrase ranking** based on four different criteria

Example of TurboTopics

Annotated documents

What is phase$_{11}$ transition$_{11}$? Why is there phase$_{11}$ transitions$_{11}$? These is are old$_{127}$ questions$_{127}$ people$_{170}$ have been asking$_{195}$ for many years$_{127}$ but get$_{153}$ few answers$_{127}$ We established$_{127}$ one general$_{11}$ theory$_{127}$ based$_{153}$ on game$_{153}$ theory$_{127}$ and topology$_{85}$ it provides$_{11}$ a basic$_{127}$ understanding$_{127}$ to phase$_{11}$ transitions$_{11}$ We proposed$_{11}$ a modern$_{127}$ definition$_{117}$ of phase$_{11}$ transition$_{11}$ based$_{153}$ on game$_{153}$ theory$_{127}$ and topology$_{85}$ of symmetry$_{11}$ group$_{184}$ which unified$_{135}$ Ehrenfests definition$_{117}$ A spontaneous$_{11}$ result$_{68}$ of this topological$_{85}$ phase$_{11}$ transition$_{11}$ theory$_{127}$ is the universal$_{14}$ equation$_{117}$ of coexistence$_{195}$ curve$_{195}$ in phase$_{11}$ diagram$_{11}$ it holds$_{153}$ both for classical$_{122}$ and quantum$_{11}$ phase$_{11}$ transition$_{11}$ This

LDA topic #11

phase, transitions, phases, transition, quantum, critical, symmetry, field, point, model, order, diagram, systems, two, theory, system, study, breaking, spin, first

Turbo topic #11

phase transitions, model, symmetry, point, quantum, systems, phase transition, phase diagram, system, order, field, order, parameter, critical, two, transitions in, models, different, symmetry breaking, first order, phenomena

- Perform LDA on corpus to assign each token a topic label
    - E.g., … phase$_{11}$ transition$_{11}$ …. game$_{153}$ theory$_{127}$ …
- Then merge adjacent unigrams with same topic label

KERT: Topical Keyphrase Extraction & Ranking
[Danilevsky, et al. 2014]

knowledge discovery using least squares support vector machine classifiers

support vectors for reinforcement learning

a hybrid approach to feature selection

pseudo conditional random fields

automatic web page classification in a dynamic and hierarchical way

inverse time dependency in convex regularized learning

postprocessing decision trees to extract actionable knowledge

variance minimization least squares support vector machines

…

Unigram topic assignment: Topic 1 & Topic 2

learning
support vector machines
reinforcement learning
feature selection
conditional random fields
classification
decision trees
:

Topical keyphrase extraction & ranking

Framework of KERT

1. Run bag-of-words model inference and assign topic label to each token
2. Extract candidate keyphrases within each topic

Frequent pattern mining

3. Rank the keyphrases in each topic
   - Popularity: 'information retrieval' vs. 'cross-language information retrieval'
   - Discriminativeness: only frequent in documents about topic t
   - Concordance: 'active learning' vs. 'learning classification'
   - Completeness: 'vector machine' vs. 'support vector machine'

Comparability property: directly compare phrases of mixed lengths

KERT: Topical Phrases on Machine Learning

Top-Ranked Phrases by Mining Paper Titles in DBLP

kpRel [Zhao et al. 11]	KERT (-popularity)	KERT (-discriminativeness)	KERT (-concordance)	KERT [Danilevsky et al. 14]
learning	effective	support vector machines	learning	learning
classification	text	feature selection	classification	support vector machines
selection	probabilistic	reinforcement learning	selection	reinforcement learning
models	identification	conditional random fields	feature	feature selection
algorithm	mapping	constraint satisfaction	decision	conditional random fields
features	task	decision trees	bayesian	classification
decision	planning	dimensionality reduction	trees	decision trees
:	:	:	:	:

The topic that represents the area of Machine Learning

Session 4. Strategy 3: First Phrase Mining then Topic Modeling

Strategy 3: First Phrase Mining then Topic Modeling

- **ToPMine** [El-Kishky et al. VLDB'15]
  - First phrase construction, then topic mining
  - Contrast with KERT: topic modeling, then phrase mining
- **The ToPMine Framework:**
  - Perform frequent ***contiguous pattern*** mining to extract candidate phrases and their counts
  - Perform agglomerative merging of adjacent unigrams as guided by a significance score—This segments each document into a ***"bag-of-phrases"***
  - The newly formed bag-of-phrases are passed as input to PhraseLDA, an extension of LDA, that constrains all words in a phrase to each sharing the same latent topic

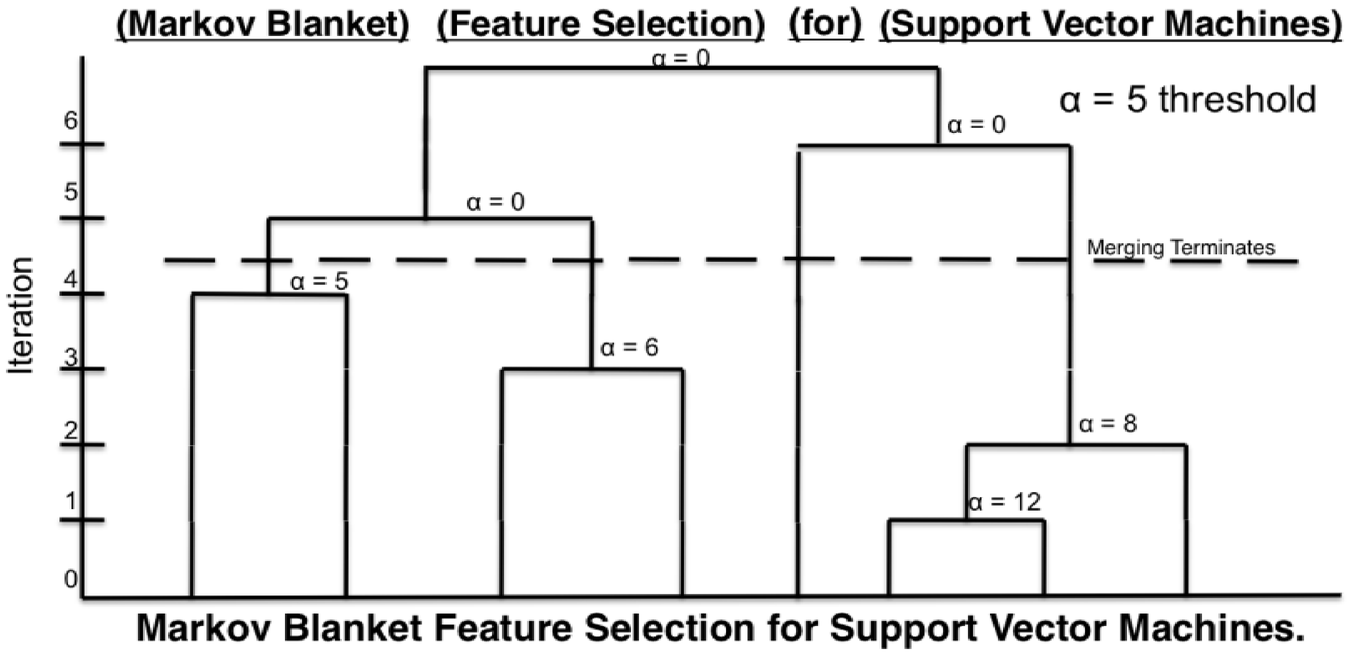
Why First Phrase Mining then Topic Modeling ?

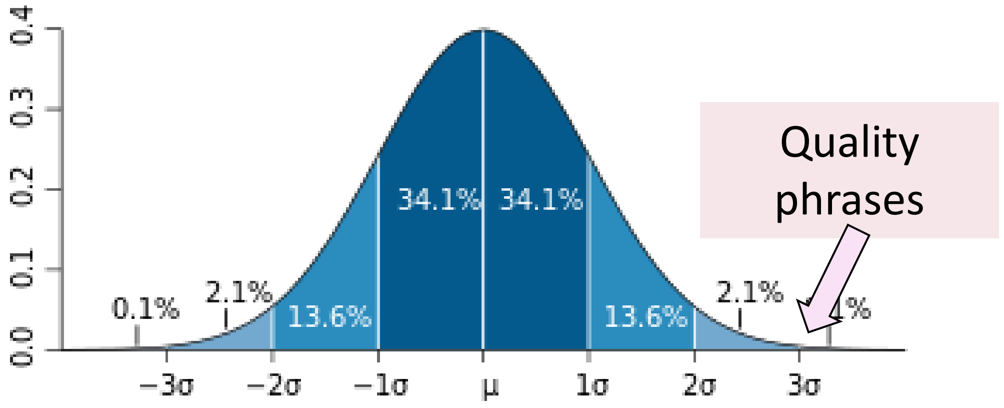
- With Strategy 2, tokens in the same phrase may be assigned to different topics
  - Ex. knowledge discovery using least squares support vector machine classifiers…
  - *Knowledge discovery* and *support vector machine* should have coherent topic labels
- Solution: switch the order of phrase mining and topic model inference

[knowledge discovery] using [least squares] [support vector machine] [classifiers] … → [knowledge discovery] using [least squares] [support vector machine] [classifiers] …

- Techniques
  - Phrase mining and document segmentation
  - Topic model inference with phrase constraint

Phrase Mining: Frequent Pattern Mining + Statistical Analysis





Based on significance score [Church et al.'91]:

$$\alpha(P_1, P_2) \approx (f(P_1 \bullet P_2) - \mu_0(P_1,P_2))/\sqrt{f(P_1 \bullet P_2)}$$

[Markov blanket] [feature selection] for [support vector machines]
[knowledge discovery] using [least squares] [support vector machine] [classifiers]
…[support vector] for [machine learning]…

Phrase	Raw freq.	True freq.
[support vector machine]	90	80
[vector machine]	95	0
[support vector]	100	20

Collocation Mining

- Collocation: A sequence of words that occur more frequently than expected
  - Often "interesting" and due to their non-compositionality, often relay information not portrayed by their constituent terms (e.g., "made an exception", "strong tea")
- Many different measures used to extract collocations from a corpus [Dunning 93, Pederson 96]
  - E.g., mutual information, t-test, z-test, chi-squared test, likelihood ratio

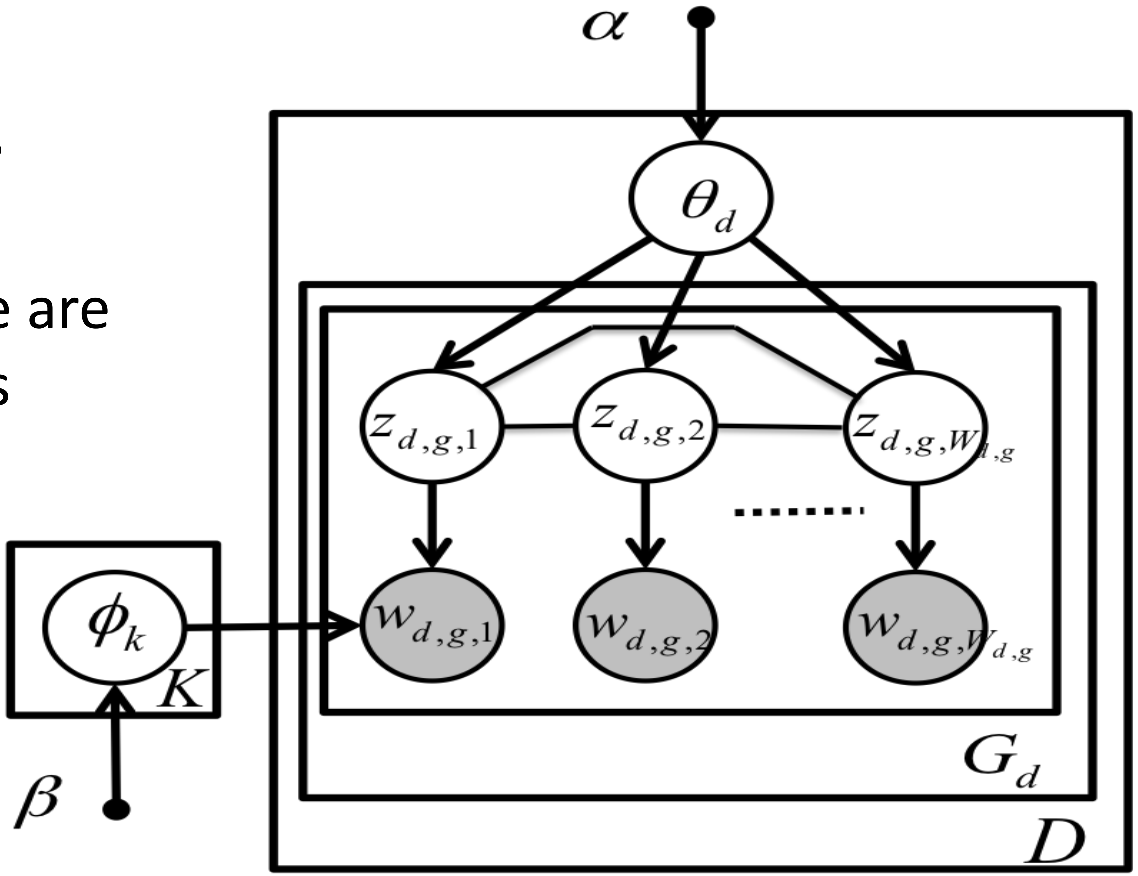
$$\mathrm{PMI}(x,y) = \log \frac{p(x,y)}{p(x)p(y)} \quad sig = \frac{count(phr_{x+y}) - E[count(phr_{x+y})]}{\sqrt{count(phr_{x+y})}} \quad \chi^2 = \sum \frac{(O-E)^2}{E}$$

- Many of these measures can be used to guide the agglomerative phrase-segmentation algorithm

ToPMine: Phrase LDA (Constrained Topic Modeling)

- The generative model for PhraseLDA is the same as LDA
- Difference: the model incorporates constraints obtained from the **"bag-of-phrases"** input
  - Chain-graph shows that all words in a phrase are constrained to take on the same topic values

[knowledge discovery] using [least squares] [support vector machine] [classifiers] …



Topic model inference with phrase constraints

Example Topical Phrases: A Comparison

social networks	information retrieval
web search	text classification
time series	machine learning
search engine	support vector machines
management system	information extraction
real time	neural networks
decision trees	text categorization
:	:
Topic 1	Topic 2

information retrieval	feature selection
social networks	machine learning
web search	semi supervised
search engine	large scale
information extraction	support vector machines
question answering	active learning
web pages	face recognition
:	:
Topic 1	Topic 2

PDLDA [Lindsey et al. 12] Strategy 1 (3.72 hours)

ToPMine [El-kishky et al. 14] Strategy 3 (67 seconds)

ToPMine: Experiments on DBLP Abstracts

	Topic 1	*Topic 2*	*Topic 3*	*Topic 4*	*Topic 5*
unigrams	problem	word	data	programming	data
	algorithm	language	method	language	patterns
	optimal	text	algorithm	code	mining
	solution	speech	learning	type	rules
	search	system	clustering	object	set
	solve	recognition	classification	implementation	event
	constraints	character	based	system	time
	programming	translation	features	compiler	association
	heuristic	sentences	proposed	java	stream
	genetic	grammar	classifier	data	large
n-grams	genetic algorithm	natural language	data sets	programming language	data mining
	optimization problem	speech recognition	support vector machine	source code	data sets
	solve this problem	language model	learning algorithm	object oriented	data streams
	optimal solution	natural language processing	machine learning	type system	association rules
	evolutionary algorithm	machine translation	feature selection	data structure	data collection
	local search	recognition system	paper we propose	program execution	time series
	search space	context free grammars	clustering algorithm	run time	data analysis
	optimization algorithm	sign language	decision tree	code generation	mining algorithms
	search algorithm	recognition rate	proposed method	object oriented programming	spatio temporal
	objective function	character recognition	training data	java programs	frequent itemsets

ToPMine: Topics on Associate Press News (1989)

	Topic 1	*Topic 2*	*Topic 3*	*Topic 4*	*Topic 5*
unigrams	plant	church	palestinian	bush	drug
	nuclear	catholic	israeli	house	aid
	environmental	religious	israel	senate	health
	energy	bishop	arab	year	hospital
	year	pope	plo	bill	medical
	waste	roman	army	president	patients
	department	jewish	reported	congress	research
	power	rev	west	tax	test
	state	john	bank	budget	study
	chemical	christian	state	committee	disease
n-grams	energy department	roman catholic	gaza strip	president bush	health care
	environmental protection agency	pope john paul	west bank	white house	medical center
	nuclear weapons	john paul	palestine liberation prganization	bush administration	united states
	acid rain	catholic church	united states	house and senate	aids virus
	nuclear power plant	anti semitism	arab reports	members of congress	drug abuse
	hazardous waste	baptist church	prime minister	defense secretary	food and drug administration
	savannah river	united states	yitzhak shamir	capital gains tax	aids patient
	rocky flats	lutheran church	israel radio	pay raise	centers for disease control
	nuclear power	episcopal church	occupied territories	house members	heart disease
	natural gas	church members	occupied west bank	committee chairman	drug testing

ToPMine: Experiments on Yelp Reviews

	Topic 1	*Topic 2*	*Topic 3*	*Topic 4*	*Topic 5*
unigrams	coffee	food	room	store	good
	ice	good	parking	shop	food
	cream	place	hotel	prices	place
	flavor	ordered	stay	find	burger
	egg	chicken	time	place	ordered
	chocolate	roll	nice	buy	fries
	breakfast	sushi	place	selection	chicken
	tea	restaurant	great	items	tacos
	cake	dish	area	love	cheese
	sweet	rice	pool	great	time
n-grams	ice cream	spring rolls	parking lot	grocery store	mexican food
	iced tea	food was good	front desk	great selection	chips and salsa
	french toast	fried rice	spring training	farmer's market	food was good
	hash browns	egg rolls	staying at the hotel	great prices	hot dog
	frozen yogurt	chinese food	dog park	parking lot	rice and beans
	eggs benedict	pad thai	room was clean	wal mart	sweet potato fries
	peanut butter	dim sum	pool area	shopping center	pretty good
	cup of coffee	thai food	great place	great place	carne asada
	iced coffee	pretty good	staff is friendly	prices are reasonable	mac and cheese
	scrambled eggs	lunch specials	free wifi	love this place	fish tacos

Session 5. A Comparative Study of Three Strategies

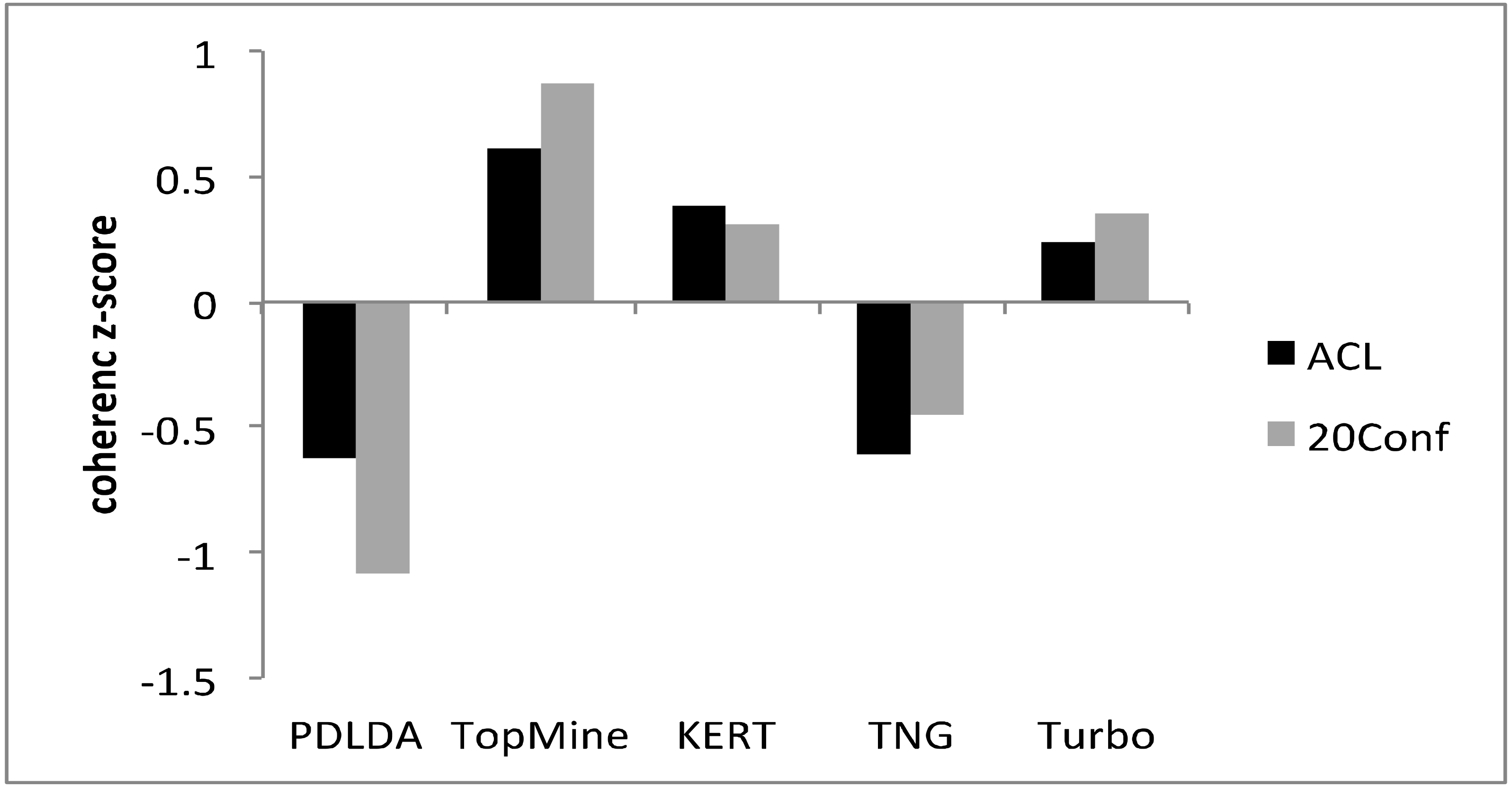
Efficiency: Running Time of Different Strategies

Method	*sampled dblp titles (k=5)*	*dblp titles (k=30)*	*sampled dblp abstracts*	*dblp abstracts*
PDLDA	3.72(hrs)	~20.44(days)	1.12(days)	~95.9(days)
Turbo Topics	6.68(hrs)	>30(days)*	>10(days)*	>50(days)*
TNG	146(s)	5.57 (hrs)	853(s)	NA†
LDA	**65(s)**	3.04 (hrs)	353(s)	13.84(hours)
KERT	68(s)	3.08(hrs)	1215(s)	NA†
ToP-Mine	67(s)	**2.45(hrs)**	**340(s)**	**10.88(hrs)**

Running time: strategy 3 > strategy 2 > strategy 1 (“>” means outperforms)

- Strategy 1: Generate bag-of-words → generate sequence of tokens
- Strategy 2: Post bag-of-words model inference, visualize topics with n-grams
- Strategy 3: Prior bag-of-words model inference, mine phrases and impose to the bag-of-words model

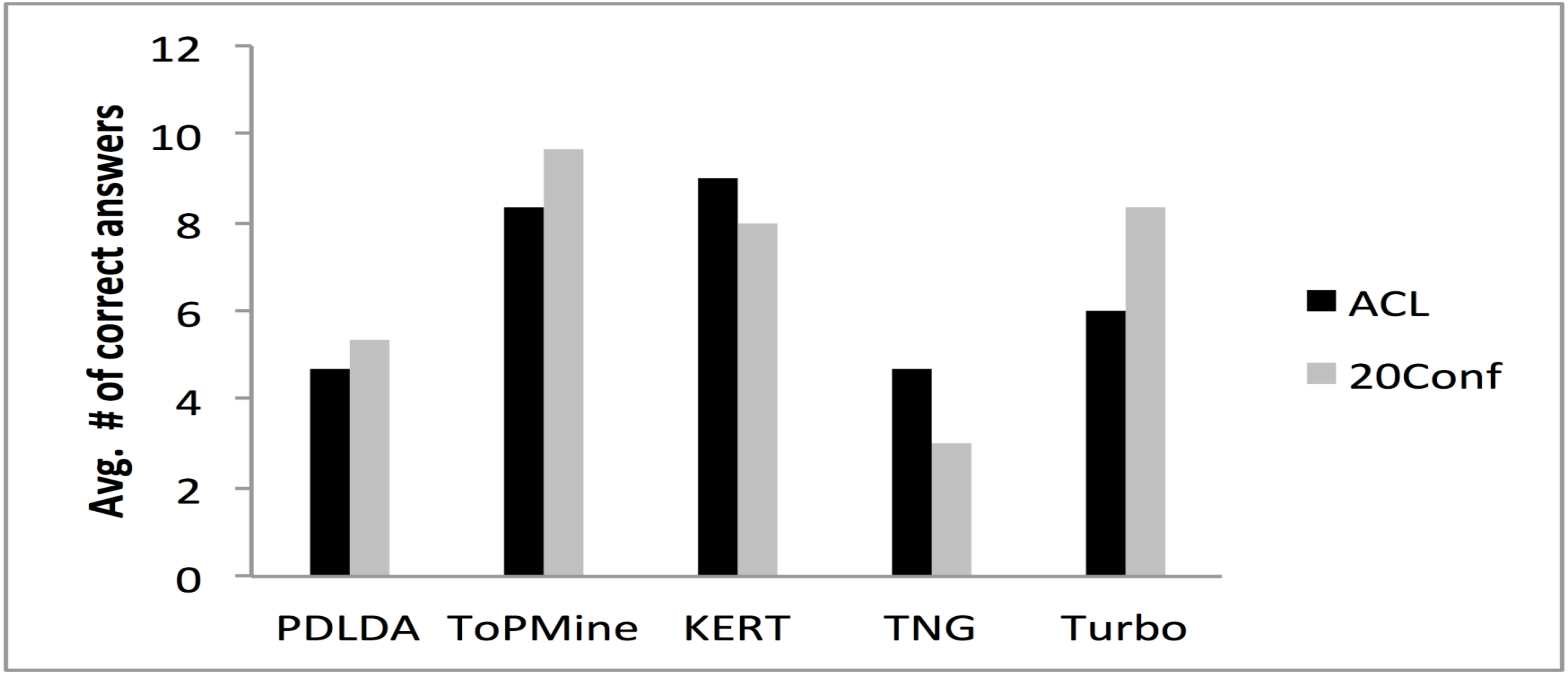
Coherence of Topics: Comparison of Strategies



Coherence measured by z-score: strategy 3 > strategy 2 > strategy 1

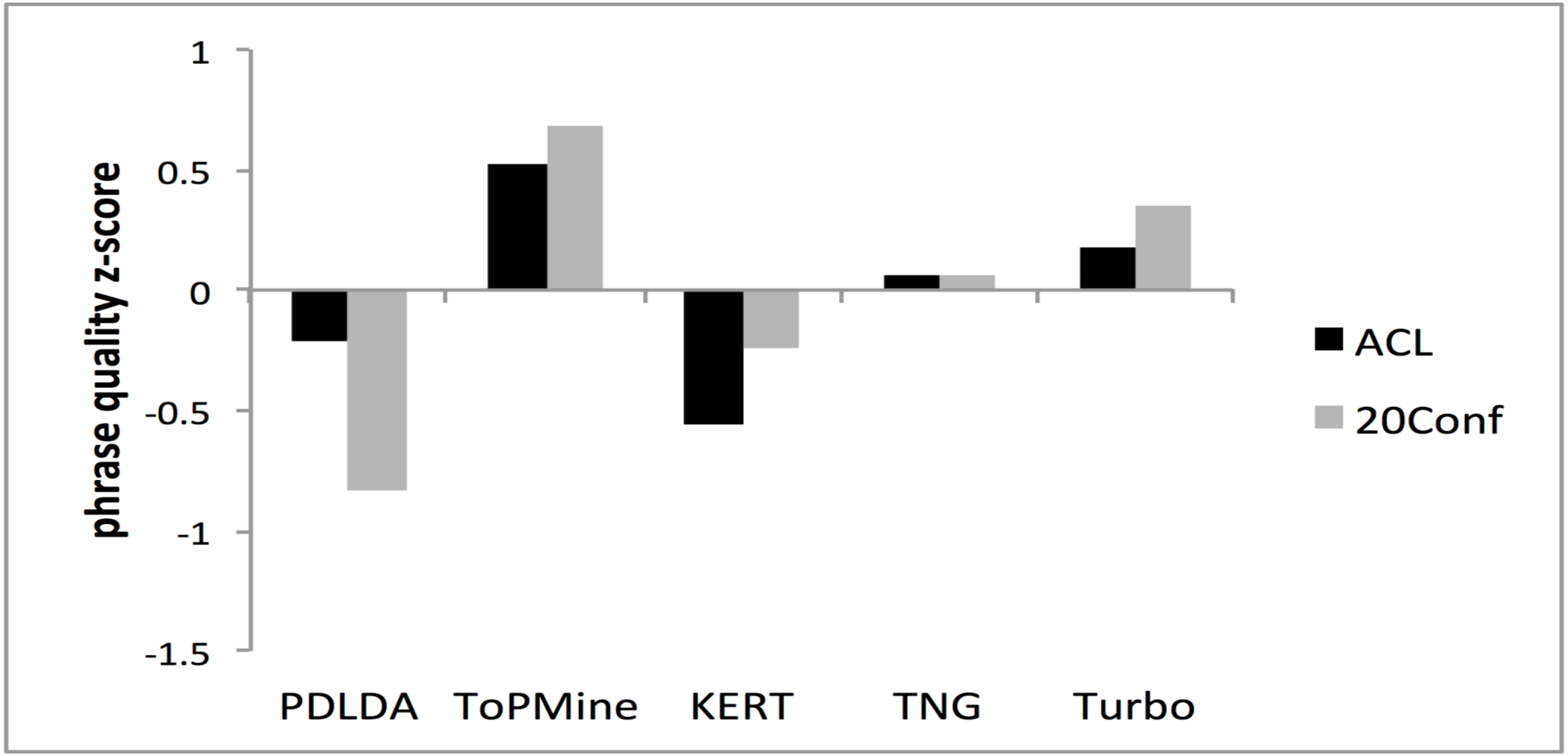
- Strategy 1: Generate bag-of-words → generate sequence of tokens
- Strategy 2: Post bag-of-words model inference, visualize topics with n-grams
- Strategy 3: Prior bag-of-words model inference, mine phrases and impose to the bag-of-words model

Phrase Intrusion: Comparison of Strategies



Phrase intrusion measured by average number of correct answers:
strategy 3 > strategy 2 > strategy 1

Phrase Quality: Comparison of Strategies



Phrase quality measured by z-score:
strategy 3 > strategy 2 > strategy 1

Summary: Strategies on Topical Phrase Mining

- Strategy 1: Generate bag-of-words → generate sequence of tokens
  - Integrated complex model; phrase quality and topic inference rely on each other
  - Slow and overfitting
- Strategy 2: Post bag-of-words model inference, visualize topics with n-grams
  - Phrase quality relies on topic labels for unigrams
  - Can be fast; generally high-quality topics and phrases
- Strategy 3: Prior bag-of-words model inference, mine phrases and impose to the bag-of-words model
  - Topic inference relies on correct segmentation of documents, but not sensitive
  - Can be fast; generally high-quality topics and phrases

Recommended Readings

- M. Danilevsky, C. Wang, N. Desai, X. Ren, J. Guo, J. Han. Automatic Construction and Ranking of Topical Keyphrases on Collections of Short Documents“, SDM’14
- X. Wang, A. McCallum, X. Wei. Topical n-grams: Phrase and topic discovery, with an application to information retrieval, ICDM’07
- R. V. Lindsey, W. P. Headden, III, M. J. Stipicevic. A phrase-discovering topic model using hierarchical pitman-yor processes, EMNLP-CoNLL’12.
- Q. Mei, X. Shen, C. Zhai. Automatic labeling of multinomial topic models, KDD’07
- D. M. Blei and J. D. Lafferty. Visualizing Topics with Multi-Word Expressions, arXiv:0907.1013, 2009
- M. Danilevsky, C. Wang, N. Desai, J. Guo, J. Han. Automatic Construction and Ranking of Topical Keyphrases on Collections of Short Documents, SDM’14
- A. El-Kishky, Y. Song, C. Wang, C. R. Voss, J. Han. Scalable Topical Phrase Mining from Text Corpora, VLDB'15
- K. Church, W. Gale, P. Hanks, D. Hindle. Using Statistics in Lexical Analysis. In U. Zernik (ed.), Lexical Acquisition: Exploiting On-Line Resources to Build a Lexicon. Lawrence Erlbaum, 1991