

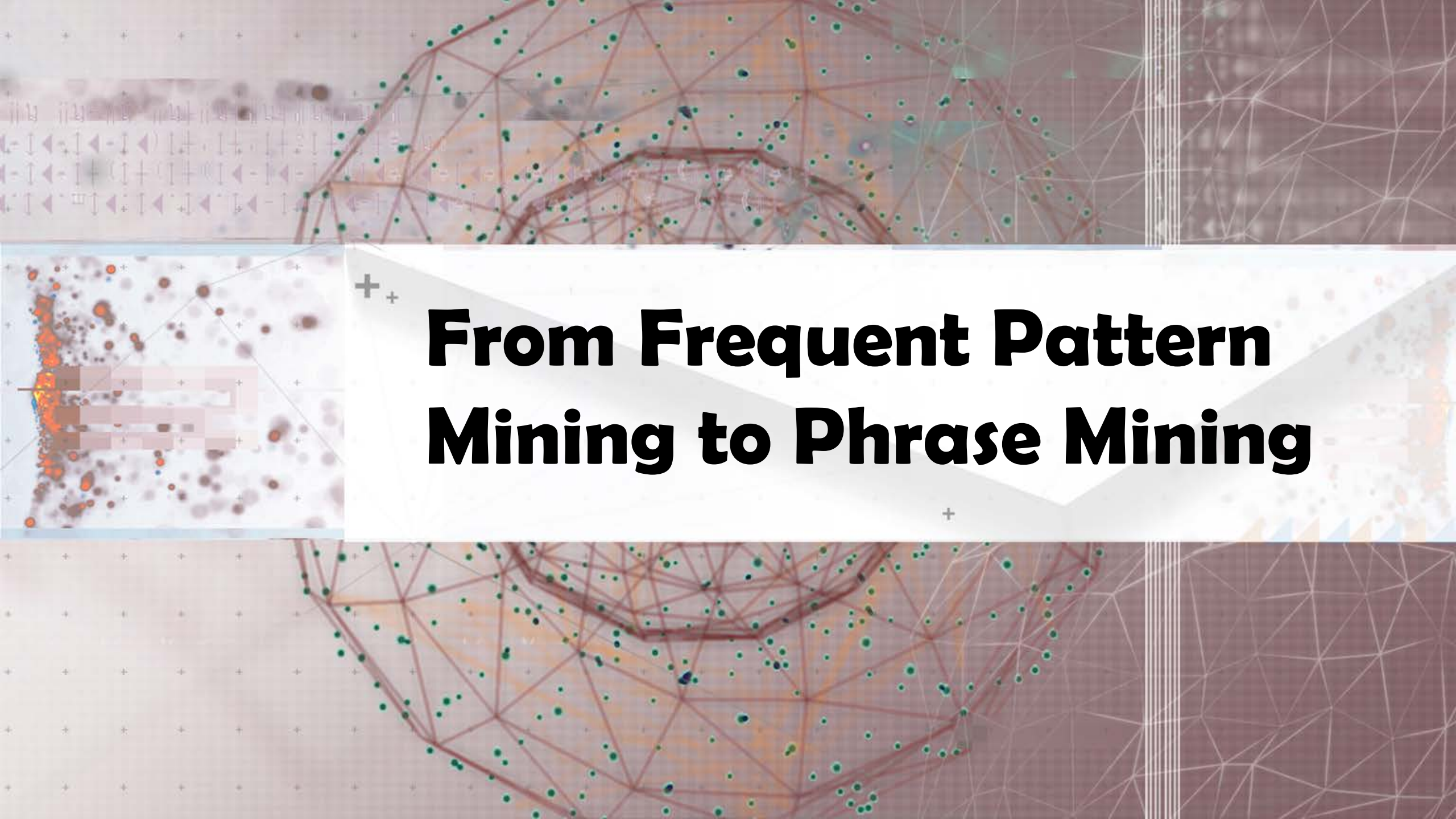
The background features a complex network graph with red lines connecting green and blue nodes. On the left, there is a smaller inset showing a heatmap with orange and red clusters. The text is overlaid on a white, angular geometric shape.

Pattern Mining Applications: Mining Quality Phrases from Text Data

Pattern Mining Applications: Mining Quality Phrases from Text Data

- ❑ From Frequent Pattern Mining to Phrase Mining
- ❑ Previous Phrase Mining Methods
- ❑ ToPMine: Phrase Mining without Training Data
- ❑ SegPhrase: Phrase Mining with Tiny Training Sets

Thanks to Ahmed El-Kishky@UIUC, Jialu Liu@UIUC, Jingbo Shang@UIUC, Xiang Ren@UIUC, Chi Wang@MSR and Marina Danilevsky@IBM for their contributions

The background features a complex network of thin, light-colored lines forming a web-like structure. Overlaid on this are various data visualizations: a grid of small plus signs in the top-left, a series of purple arrows pointing left in the top-center, a dense cluster of orange and red dots with a heatmap overlay in the bottom-left, and a large network of green dots connected by red lines in the center and bottom-right. The overall color palette is muted, with earthy tones and soft pastels.

From Frequent Pattern Mining to Phrase Mining

Why Phrase Mining?

- ❑ Unigrams vs. phrases
 - ❑ **Unigrams** (single words) are often *ambiguous*
 - ❑ Example: “United”: United States? United Airline? United Parcel Service?
 - ❑ **Phrase**: A natural, meaningful, *unambiguous* semantic unit
 - ❑ Example: “United States” vs. “United Airline”
- ❑ Mining semantically meaningful phrases
 - ❑ Transform text data from *word granularity* to *phrase granularity*
 - ❑ Enhance the power and efficiency at manipulating unstructured data

From Frequent Pattern Mining to Phrase Mining

- General principle
 - Exploit information redundancy and data-driven criteria to determine phrase boundaries and salience
- Methodology: Exploring three ideas
 - Frequent pattern mining and colocation analysis
 - Phrasal segmentation
 - Quality phrase assessment
- Recent developments of phrase mining methods
 - ToPMine: Mining quality phrase without training (A. El-Kishky, et al., 2015)
 - SegPhrase: Mining quality phrase with tiny training sets (J. Liu, et al., 2015)

The background of the slide is a complex, abstract composition. It features a dark, reddish-brown base with a network of thin, light-colored lines forming a web-like structure. Scattered throughout are numerous small, colored dots in shades of green, blue, and orange. On the left side, there is a vertical strip with a grid of small, light-colored squares. In the center, a large, white, angular shape points downwards, serving as a backdrop for the title. The title itself is in a bold, black, sans-serif font.

Previous Phrase Mining Methods

Phrase Mining: Can We Reduce Annotation Cost?

- ❑ Phrase mining: Originated from the NLP community—“Chunking”
 - ❑ Model it as a sequence labeling problem (B-NP, I-NP, O, ...)
- ❑ Need annotation and training
 - ❑ Annotate hundreds of documents as training data
 - ❑ Train a supervised model based on part-of-speech features
- ❑ Recent trend:
 - ❑ Use distributional features based on web n-grams (Bergsma et al., 2010)
 - ❑ State-of-the-art performance: ~95% accuracy, ~88% phrase-level F-score
- ❑ Limitations
 - ❑ High annotation cost, not scalable to a new language, a new domain/genre
 - ❑ May not fit domain-specific, dynamic, emerging applications
 - ❑ Scientific domains, query logs, or social media (e.g., Yelp and Twitter data)

Unsupervised Phrase Mining and Topic Modeling

- ❑ Many studies of unsupervised phrase mining are linked with topic modeling
- ❑ Topic modeling
 - ❑ Represents documents by multiple topics in different proportions
 - ❑ Each topic is represented by a word distribution
 - ❑ Does not require any prior annotations or labeling of the documents
- ❑ Statistical topic modeling algorithms
 - ❑ The most common algorithm: LDA (Latent Dirichlet Allocation) [Blei, et al., 2003]
- ❑ Three strategies on phrase mining with topic modeling
 - ❑ 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 on the bag-of-words model

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] (a generalization of Bigram Topic Model)
 - ❑ Probabilistic model that generates words in textual order
 - ❑ Create n-grams by concatenating successive bigrams
- ❑ **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
- ❑ Comments on this strategy
 - ❑ High model complexity: Tends to overfitting
 - ❑ High inference cost: Slow

Strategy 2: Post Topic-Modeling Phrase Construction (I): TurboTopics

- **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

Annotated documents

What is **phase₁₁ transition₁₁**? Why is there **phase₁₁ transitions₁₁**? These is are old₁₂₇ questions₁₂₇ people₁₇₀ have been asking₁₉₅ for many years₁₂₇ but get₁₅₃ few answers₁₂₇ We established₁₂₇ one **general₁₁** theory₁₂₇ based₁₅₃ on game₁₅₃ theory₁₂₇ and topology₈₅ it **provides₁₁** a basic₁₂₇ understanding₁₂₇ to **phase₁₁ transitions₁₁** We **proposed₁₁** a modern₁₂₇ definition₁₁₇ of **phase₁₁ transition₁₁** based₁₅₃ on game₁₅₃ theory₁₂₇ and topology₈₅ of **symmetry₁₁** group₁₈₄ which unified₁₃₅ Ehrenfests definition₁₁₇ A **spontaneous₁₁** result₆₈ of this topological₈₅ **phase₁₁ transition₁₁** theory₁₂₇ is the universal₁₄ equation₁₁₇ of coexistence₁₉₅ curve₁₉₅ in **phase₁₁ diagram₁₁** it holds₁₅₃ both for classical₁₂₂ and **quantum₁₁ phase₁₁ transition₁₁** 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

Post Topic-Modeling Phrase Construction (II): KERT

- ❑ **KERT** [Danilevsky et al.'14] – Phrase construction as a post-processing step to LDA
 - ❑ Run bag-of-words model inference and assign topic label to each token
 - ❑ Perform **frequent pattern mining** to extract candidate phrases within each topic
 - ❑ Perform **phrase ranking** based on four different criteria
 - ❑ **Popularity:** e.g., “information retrieval” vs. “cross-language information retrieval”
 - ❑ **Concordance**
 - ❑ “powerful tea” vs. “strong tea”
 - ❑ “active learning” vs. “learning classification”
 - ❑ **Informativeness:** e.g., “this paper” (frequent but not discriminative, not informative)
 - ❑ **Completeness:** e.g., “vector machine” vs. “support vector machine”

Comparability property: directly compare phrases of mixed lengths

The background features a complex network graph with red lines connecting green and blue nodes. On the left, there is a smaller inset showing a heatmap with orange and red clusters. The overall aesthetic is technical and data-driven.

ToPMine: Phrase Mining without Training Data

Strategy 3: First Phrase Mining then Topic Modeling

- 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 for this strategy
 - Phrase mining, document segmentation, and phrase ranking
 - Topic model inference with phrase constraint

ToPMine: Phrase Mining before Topic Modeling

- ❑ **ToPMine** [El-Kishky et al. VLDB'15]: Phrase mining, then phrase-based topic modeling
- ❑ Phrase mining
 - ❑ **Frequent *contiguous pattern* mining**: Extract candidate phrases and their counts
 - ❑ Agglomerative merging of adjacent unigrams as guided by a **significance score**
 - ❑ Document segmentation to count phrase occurrence
 - ❑ Calculate rectified (i.e., true) phrase frequency
 - ❑ Phrase ranking (using the criteria proposed in KERT)
 - ❑ Popularity, concordance, informativeness, completeness
- ❑ Phrase-based topic modeling
 - ❑ The mined 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

Phrase	Raw frequency	Rectified frequency
[support vector machine]	90	80
[vector machine]	95	0
[support vector]	100	20

Collocation Mining

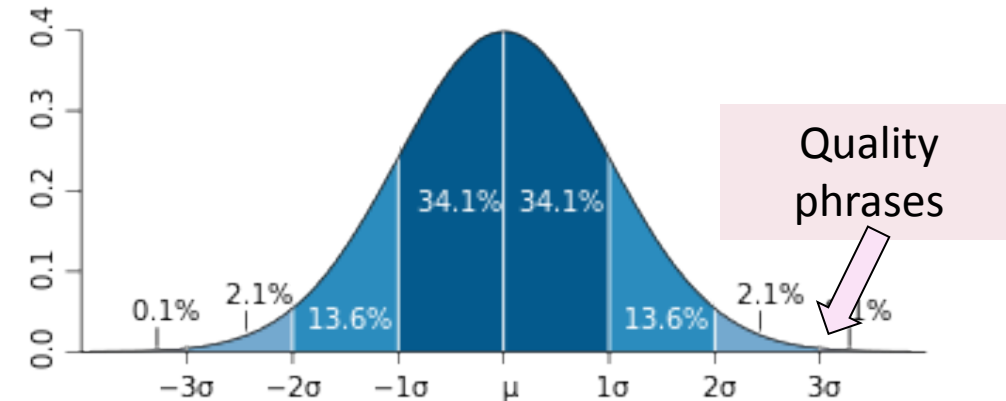
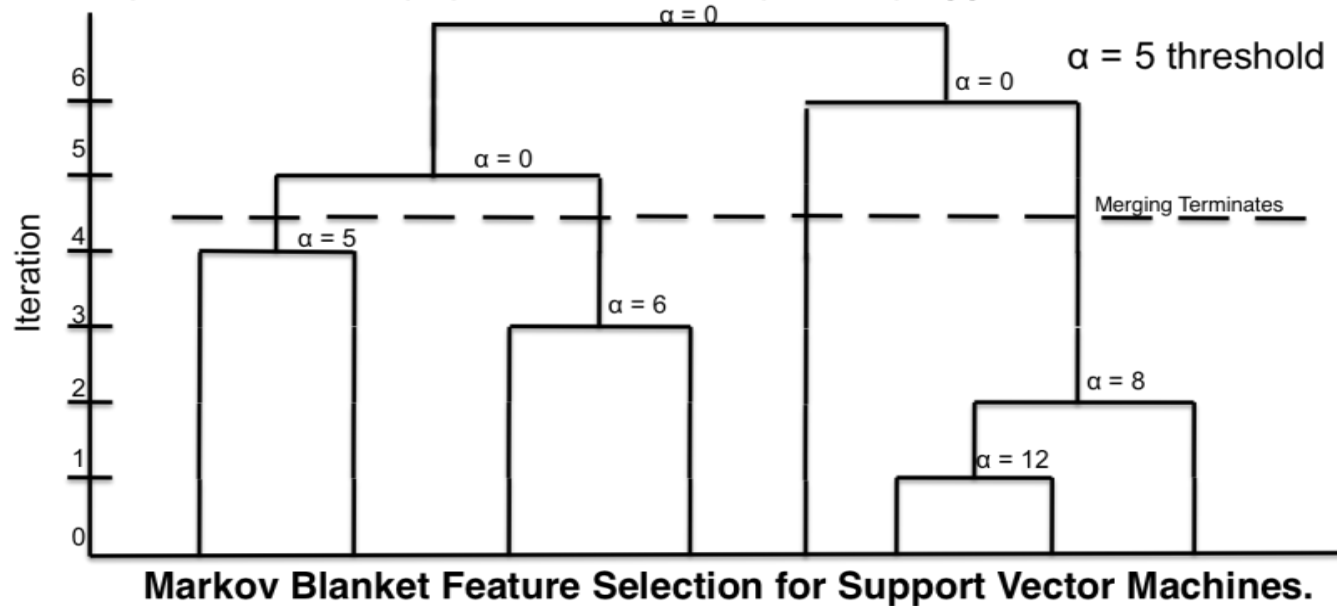
- Collocation: A sequence of words that occur **more frequently than expected**
 - Often “interesting”, relay information not portrayed by their constituent terms
 - Ex. “made an exception”, “strong tea”
- Many different measures used to extract collocations from a corpus [Dunning 93, Pederson 96]
 -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

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

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

Phrase Candidate Generation: Frequent Pattern Mining + Statistical Analysis

(Markov Blanket) (Feature Selection) (for) (Support Vector Machines)



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

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

Note for the first title:

- ❑ [feature selection] forms phrase but not [selection for] based on the significant scores computed
- ❑ [support vector machine] does not contribute to the counts of [support], [vector], [support vector], [vector machine]

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



ToPMine: Experiments on DBLP Abstracts

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

ToPMine is efficient and generates high-quality topics and phrases without any training data



ToPMine: Experiments on Yelp Reviews

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

ToPMine works well for phrase and topic mining in social media data

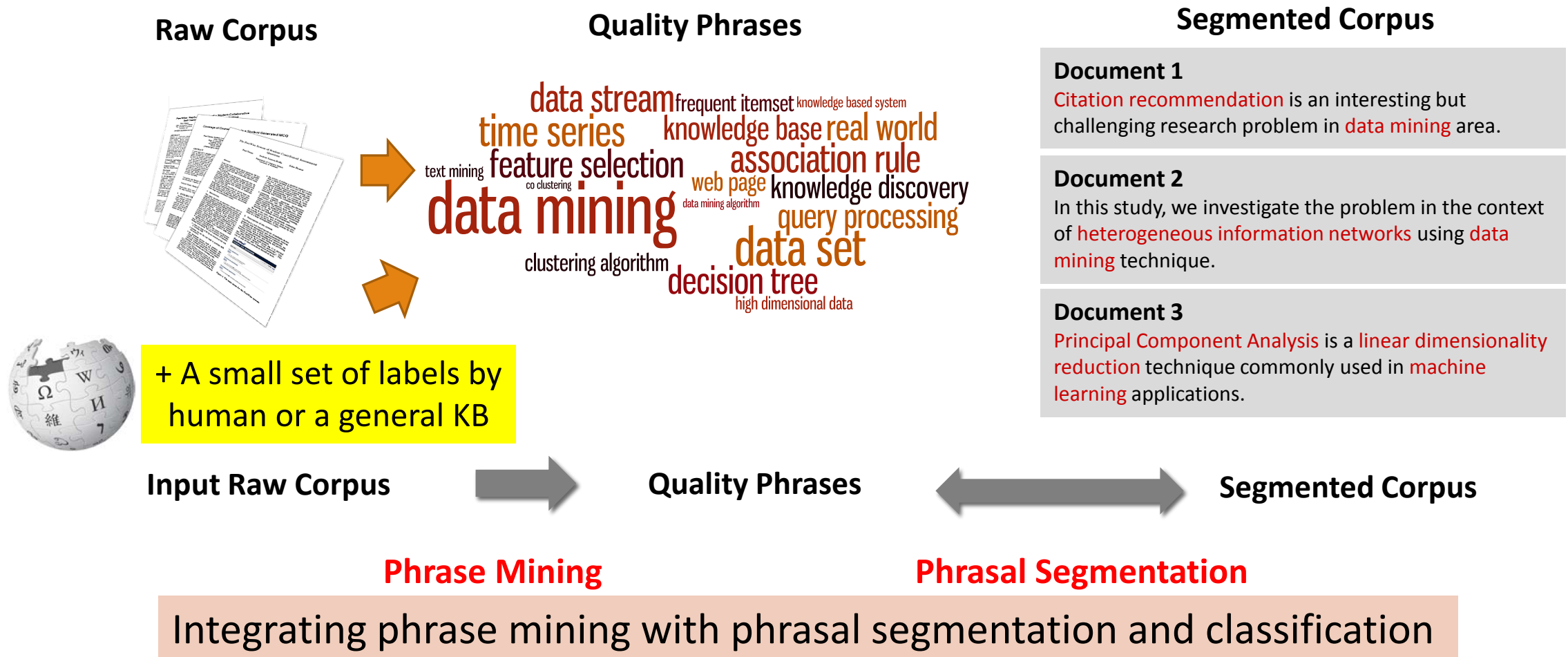


SegPhrase: Phrase Mining with Tiny Training Sets

SagPhrase: Phrase Mining with Tiny Training Sets

- A small set of training data may enhance the quality of phrase mining

J. Liu et al., Mining Quality Phrases from Massive Text Corpora. In *SIGMOD'15*

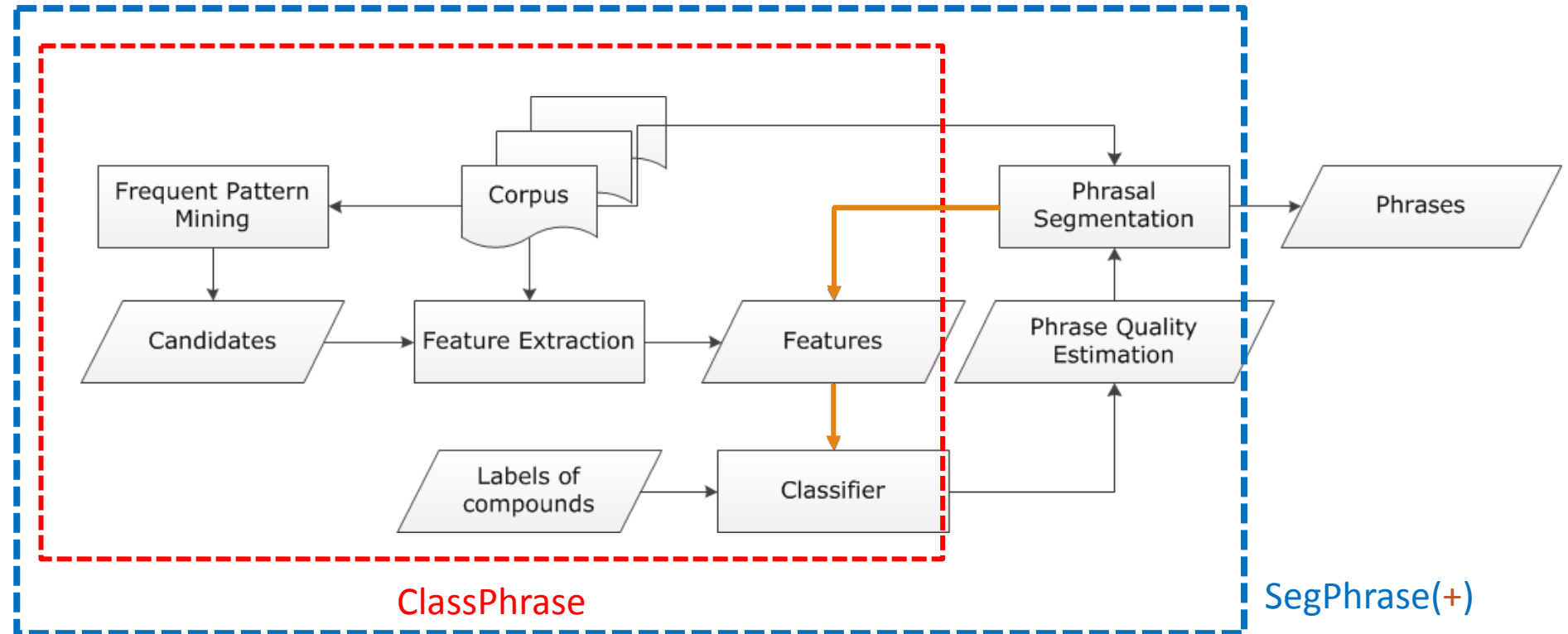


SegPhrase+: The Overall Framework

- ❑ ClassPhrase: Frequent pattern mining, feature extraction, classification
- ❑ SegPhrase: Phrasal segmentation and phrase quality estimation
- ❑ SegPhrase+: One more round to enhance mined phrase quality

SegPhrase (a classifier is used)

Small labeled dataset provided by experts or a distant supervised KB (e.g., Wikipedia / DBPedia)



SegPhrase: Pattern Mining and Feature Extraction

❑ Pattern Mining for Candidate Set

- ❑ Build a candidate phrases set by frequent pattern mining
 - ❑ Mining frequent k -grams (k is typically small, e.g., 6 in the experiments)
 - ❑ **Popularity** measured by *raw* frequent words and phrases mined from the corpus

❑ Feature Extraction: Concordance

- ❑ Partition a phrase into two parts to check whether the co-occurrence is significantly higher than pure random

❑ Feature Extraction: Informativeness

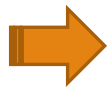
- ❑ Quality phrases typically start and end with a non-stopword
 - ❑ “machine learning is” vs. “machine learning”
- ❑ Use average IDF over words in the phrase to measure the semantics
- ❑ Usually, the probabilities of a quality phrase in quotes, brackets, or connected by hyphen should be higher (punctuations information)
 - ❑ e.g., “state-of-the-art”

SegPhrase: Classification Using Tiny Training Sets

- Use tiny training sets (300 labels for 1GB corpus; can also use phrases extracted from KBs)
 - Label: indicating whether a phrase is a high quality one
 - E.g., “support vector machine”: 1; “the experiment shows”: 0
- Classification: Construct models to distinguish quality phrases from poor ones
 - Use *Random Forest* algorithm to bootstrap different datasets with limited labels
- Phrasal segmentation can tell which phrase is more appropriate
 - Ex: “A standard [feature vector] [machine learning] setup is used to describe”
 - Not counted towards the rectified frequency
 - Partition a sequence of words by maximizing the likelihood
 - Consider length penalty and filter out phrases with low rectified frequency
- Process: Classification → Phrasal segmentation // **SegPhrase**
 - Classification → Phrasal segmentation // **SegPhrase+**

Performance: Precision Recall Curves on DBLP

□ Datasets:



□ Evaluation

- Wiki Phrases (based on internal links, ~7K high quality phrases)
- Sampled 500*7 Wiki-uncovered phrases: Results evaluated by 3 reviewers

□ Compared with other phrase-mining methods

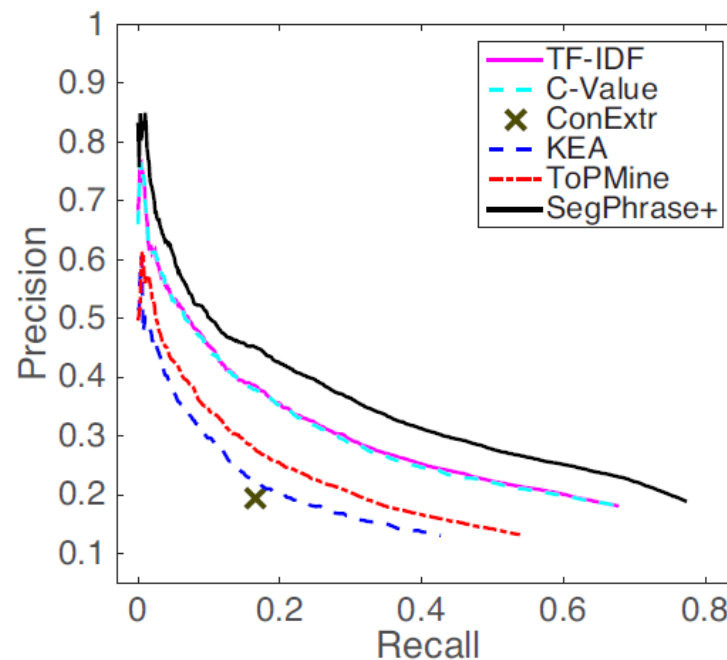
- TF-IDF, C-Value, ConExtr, KEA, and ToPMine

□ Also, Segphrase+ is efficient, linearly scalable

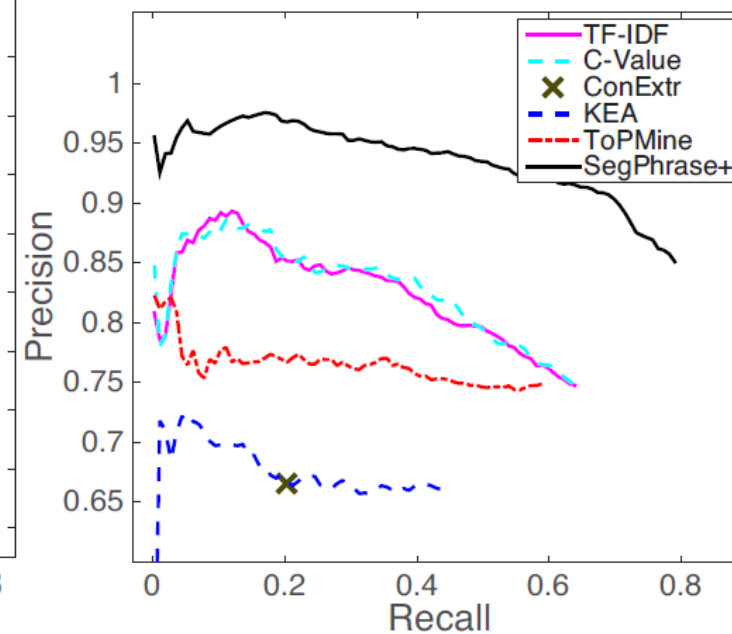
Dataset	#docs	#words	#labels
DBLP	2.77M	91.6M	300
Yelp	4.75M	145.1M	300



Use only 300 human labeled phrases for training



Precision-Recall Curves on DBLP Data (Wiki Phrases)



Precision-Recall Curves on DBLP Data (Non Wiki-phrases)

Experimental Results: Interesting Phrases Generated (From Titles & Abstracts of SIGKDD)

Query	SIGKDD	
Method	SegPhrase+	Chunking (TF-IDF & C-Value)
1	data mining	data mining
2	data set	association rule
3	association rule	knowledge discovery
4	knowledge discovery	frequent itemset
5	time series	decision tree
...
51	association rule mining	search space
52	rule set	domain knowledge
53	concept drift	important problem
54	knowledge acquisition	concurrency control
55	gene expression data	conceptual graph
...
201	web content	optimal solution
202	frequent subgraph	semantic relationship
203	intrusion detection	effective way
204	categorical attribute	space complexity
205	user preference	small set
...

Only in Chunking

Only in SegPhrase+

Mining Quality Phrases in Multiple Languages

Both ToPMine and SegPhrase+ are extensible to mining quality phrases in multiple languages

SegPhrase+ on Chinese (From Chinese Wikipedia)



ToPMine on Arabic (From Quran (Fus7a Arabic)(no preprocessing)

Experimental results of Arabic phrases:

كفروا → Those who disbelieve

بسم الله الرحمن الرحيم → In the name of God the Gracious and Merciful

Rank	Phrase	In English
...
62	首席_执行官	CEO
63	中间_偏右	Middle-right
...
84	百度_百科	Baidu Pedia
85	热带_气旋	Tropical cyclone
86	中国科学院_院士	Fellow of Chinese Academy of Sciences
...
1001	十大_中文_金曲	Top-10 Chinese Songs
1002	全球_资讯网	Global News Website
1003	天一阁_藏_明代_科举_录_选刊	A Chinese book name
...
9934	国家_戏剧_院	National Theater
9935	谢谢_你	Thank you
...



Summary

Summary: Pattern Mining Applications: Mining Quality Phrases from Text Data

- From Frequent Pattern Mining to Phrase Mining
- Previous Phrase Mining Methods
- New Methods that Integrate Pattern Mining with Phrase Mining
 - ToPMine: Phrase Mining without Training Data
 - SegPhrase: Phrase Mining with Tiny Training Sets

Recommended Readings

- ❑ S. Bergsma, E. Pitler, D. Lin, Creating robust supervised classifiers via web-scale n-gram data, ACL'2010
- ❑ D. M. Blei and J. D. Lafferty. Visualizing Topics with Multi-Word Expressions. arXiv:0907.1013, 2009
- ❑ D.M. Blei, A. Y. Ng, M. I. Jordan, J. D. Lafferty, Latent Dirichlet allocation. JMLR 2003
- ❑ 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
- ❑ 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
- ❑ A. El-Kishky, Y. Song, C. Wang, C. R. Voss, and J. Han. Scalable Topical Phrase Mining from Text Corpora. VLDB'15
- ❑ R. V. Lindsey, W. P. Headden, III, M. J. Stipicevic. A Phrase-Discovering Topic Model Using Hierarchical Pitman-Yor Processes. EMNLP-CoNLL'12.
- ❑ J. Liu, J. Shang, C. Wang, X. Ren, J. Han, Mining Quality Phrases from Massive Text Corpora. SIGMOD'15
- ❑ A. Parameswaran, H. Garcia-Molina, and A. Rajaraman. Towards the Web of Concepts: Extracting Concepts from Large Datasets. VLDB'10
- ❑ X. Wang, A. McCallum, X. Wei. Topical n-grams: Phrase and topic discovery, with an application to information retrieval. ICDM'07