› TOPIC GROUPER
A CLUSTERING APPROACH TO TOPIC MODELING

ECIR 2019 Cologne

HEILBRONN UNIVERSITY
OF APPLIED SCIENCES

REFINITIV

Topic Grouper | Daniel Pfeifer & Jochen L. Leidner

IN GENERAL, TOPIC MODELING…

> … is an **unsupervised learning** procedure, usually **on a (training) document collection *D***
  - It computes “topics” *t* as frequently co-occurring words across *D*
  - **Topics *t* are represented as distributions *p(w|t)***, where *w* is a word from the vocabulary *V* based on *D*
  - So, **words w with high probability *p(w|t)* co-occur in *D* and form the “essence” of *t***

> A document ***d* ϵ *D* is a mix of topics, represented via *p(t|d)***
  - So, **topics with high probability *p(t|d)* constitute the “topical essence” of *d***

> Basic approaches require the number of topics |*T*| with *t* ϵ *T* as a hyper parameter

"Taken from [Ble12]"

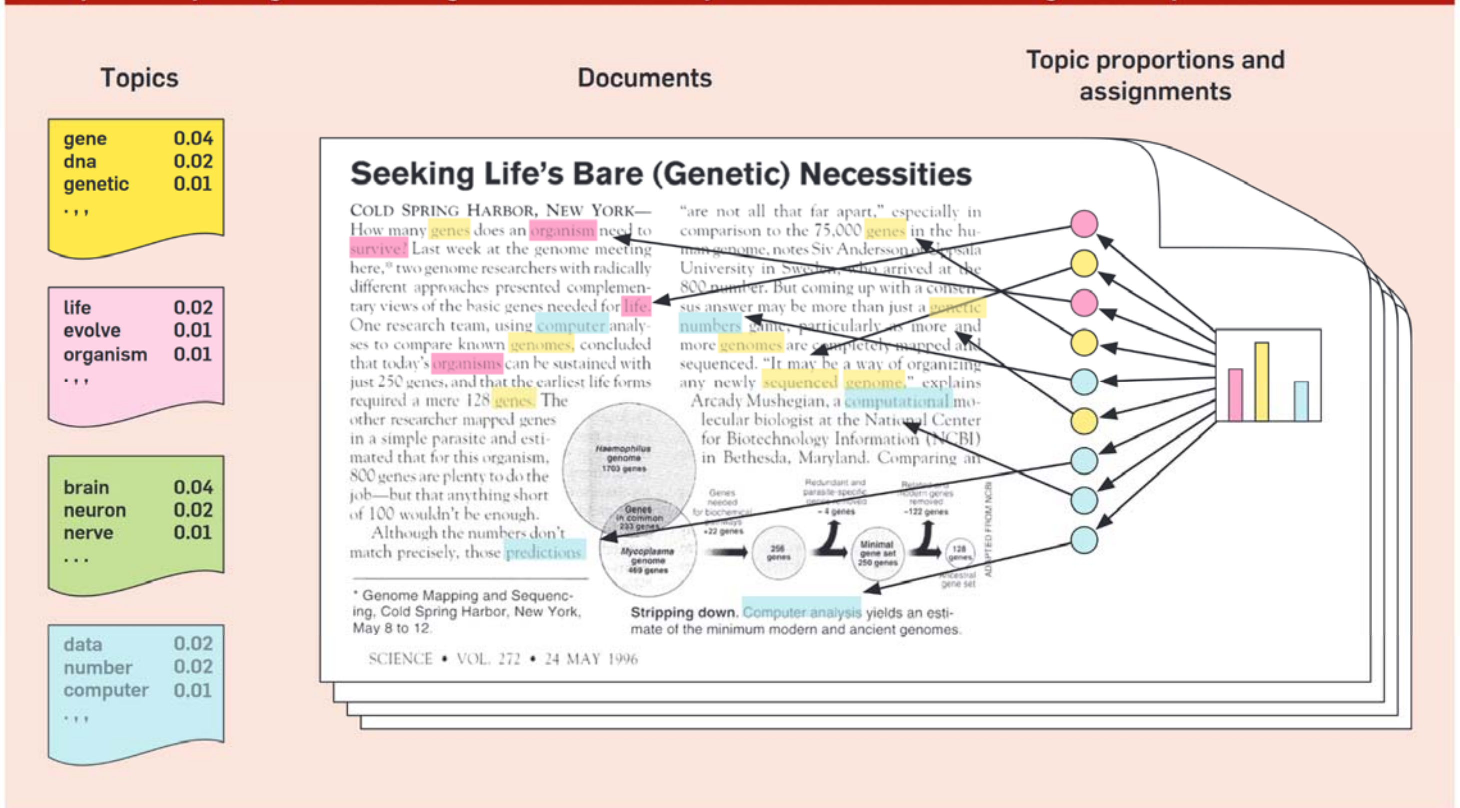


Figure 1. The intuitions behind latent Dirichlet allocation. We assume that some number of "topics," which are distributions over words, exist for the whole collection (far left). Each document is assumed to be generated as follows. First choose a distribution over the topics (the histogram at right); then, for each word, choose a topic assignment (the colored coins) and choose the word from the corresponding topic. The topics and topic assignments in this figure are illustrative—they are not fit from real data. See Figure 2 for topics fit from data.

78 COMMUNICATIONS OF THE ACM | APRIL 2012 | VOL. 55 | NO. 4

Taken from [Ble12]

TOPIC MODELING…

> … is usually based on a **probabilistic background model *Φ***
  - The aim is to **optimize *Φ*'s parameters** in order **to generate *D***
  - Each **document *d* ϵ *D* is a bag of words w ϵ *V with word frequencies* $f_d(w)$**

> A **language model under maximum likelihood** is a customary mathematical start:

This is what makes up *Φ*

$$argmax_\Phi p(D|\Phi) = argmax_\Phi \prod_{d \in D} p(d|\Phi) \qquad p(d|\Phi) = \prod_{w \in V} \overbrace{p(w|d)}^{}{}^{f_d(w)}$$

> **But:** Regarding topic modeling ***p(w|d) is a compound***, such that $p(w|d) := \sum_{t \in T} p(w|t)p(t|d)$

> This implies that to generate a *w* in *d*:
  1. First generate topic *t* via *d*
  2. Then generate *w* via *t*

  → The sum considers all possible ways to get to *w* (via any *t*)

> This is the „oldest“ (probablistic) topic model called **pLSI** [Hof99]
  - **LDA** [BNH03] refines this by using model priors and expectation values over *p(w|d)* and *p(w|d)*
  - Numerous extension, refinements and applications of LDA exist → It's made „IR history“…

PROBLEMS OF LDA AND ITS „DERIVATIVES“

> **Hyper parameters must be set** such as |T|, Dirichlet parameters α and β
  - But results are highly susceptible to corresponding settings [WMM09]

  No „best way“ on how to do this, but many methods:
  - Heuristic setting [GS04]
  - Symmetric versus asymmetric α
  - Grid search for best value combinations [AWST09]
  - Outer EM-loop for α and β nesting core LDA method [AWST09]

> Stop words and function words tend to “pollute” topics

> Hierarchical models exists but
  - simple ones are unsatisfactory regarding resulting topics [BJGT03],
  - complex ones need (even more) hyper parameters and are hard to understand [KKKO12, PWBJ15]
  - Apparently, they allow for only shallow hierarchies
  - No way to switch between hierarchical and flat topic perspective

    → Either the one or the other

OUR APPROACH…

- … holds an actually debatable simplification: **Every word is in exactly one topic!**
- This means, there is a **function *t(w): V → T***, which assigns exactly one topic to each word
  - So, **a topic *t* becomes a set of words *w* …**
  - and **the set of topics T is a partitioning of V**
- Is this a good or a (rather) bad idea? → **Let's discuss later** and go with it for now …
- Let

$$f_d(t) = \sum_{w \in t} f_d(w), f(t) = \sum_{d \in D} f_d(t), f(w) = \sum_{d \in D} f_d(w), |d| = \sum_{w \in V} f_d(w)$$

- Some corresponding maximum likelihood estimates are:

$$p(w|t(w)) = f(w)/f(t(w)), p(t|d) = f_d(t)/|d|$$

  Note, that if ***w* is not in *t*, then *p(w|t)* = *0***!
- So:

$$p(w|d) = \sum_{t \in T} p(w|t)p(t|d) = p(w|t(w))p(t(w)|d)$$

- This allows for solving the orignial problem $argmax_\Phi p(D|\Phi)$
  But what is smart way to compute it?

THE CONSEQUENCES FOR p(D)

> Remember that the set of topics T is now a partitioning V such that $s \cap t = \emptyset$ for any $s, t \in T(n)$, $\bigcup_{t \in T} t = V$

- *So:*

$$p(d) = \prod_{w \in V} p(w|d)^{f_d(w)} = \prod_{w \in V} (p(w|t(w))p(t(w)|d))^{f_d(w)} = \prod_{t \in T} \prod_{w \in t} (p(w|t)p(t|d))^{f_d(w)}$$

- And over D:

$$p(D) = \prod_{d \in D} \prod_{t \in T} \prod_{w \in t} (p(w|t)p(t|d))^{f_d(w)} = \prod_{t \in T} \underbrace{\prod_{d \in D} \prod_{w \in t} (p(w|t)p(t|d))^{f_d(w)}}_{:= h(t)}$$

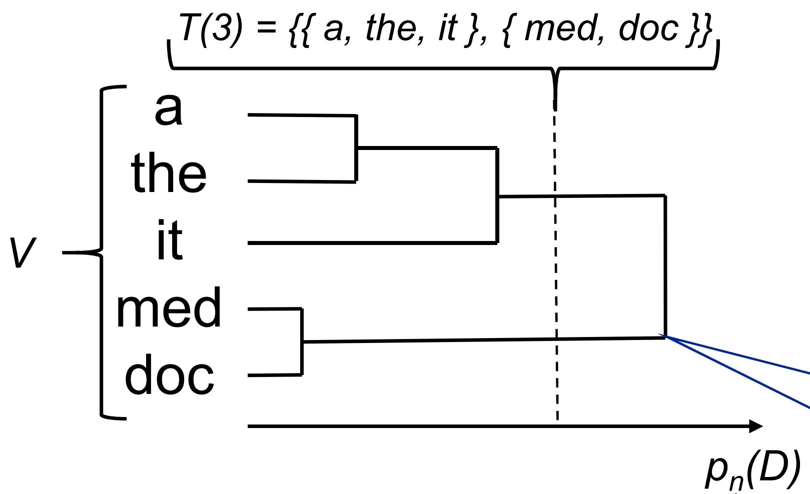
> So, **to find a partitioning *T* that maximizes *p(D), we may resort to mutually independent factors h(t)!***

> **We can use agglomerative clustering for this!**

- Start with |*V*| topics, each containing just a single word
- At each step, join two topics such that *p(D)* remains a large as possible
- The last step ends up with one topic containing all words

ILLUSTRATION OF TOPIC CLUSTERING

An example dendrogram:



T and $p(D)$ have become dependent on the clustering step n

- Clustering starts at step $n = 1$ with

  $T(n) = \{ \{a\}, \{the\}, \{it\}, \{med\}, \{doc\} \}$
- At every step, two topics are joined according to a (yet to be specified) cluster distance
- At the final step $n = |V|$ we have

  $T(n) = \{ a, the, it, med, doc \} = V$

- **One obtains a binary tree, where the number of topics $T(n)$ ranges between $|V|$ and 1**
- So, the $T(n)$, $n = 1..|V|$ form a hierarchy!

HOW TO FIND THE BEST JOIN CANDIDATES AT EACH STEP?

Let‘s consider the change of $p_n(D)$ at step $n+1$:

- Before join of $s, t \in T(n)$: $p_n(D) = \prod_{t \in T} h(t)$
- After join of s, t: $p_{n+1}(D) = p_n(D) \cdot h(s \cup t)/h(s)/h(t)$
- So the best join partners s,t are the ones with maximum $\Delta h(s,t) := h(s \cup t)/h(s)/h(t)$
- In other words: Delta **–Δh() is our cluster distance** that we can use for agglomerative clustering!
- The rest is „detail“ (see paper):
  - We **use log likelihoods** and log sums instead of products of probabilities
  - We show how to **compute Δh() efficiently**
  - Our algorithm adapts a standard agglomerative clustering algorithm „EHAC“ [MRS08]

    EHAC‘s time complexity: $O(k^2 \log k)$ where k is the number of data items / space is in $O(k^2)$

    Our adaptation‘s **time complexity: $O(|V|^2 |D|)$ / space is in $O(|V|^2)$**

EVALUATION

Three types of „testing“ done:

1. Error rate based on simple synthetic data is compared with pLSI and LDA
2. Hold-out perplexity on different real-world data sets is compared with LDA
   - Just remember that perplexity is a derived measure based on $p(D_{test})$
3. Telling example(s) for a real-world text collection (so just anekdotal)

Evaluation based on human assessment is still missing (such as in [CBG+09])

SYNTHETIC DATA ACCORDING TO [TO10]

Synthetic document generation with $|D| = 6000$:

$V = \{0, \ldots, 399\}$, $0..99 \rightarrow t_1$, $100..299 \rightarrow t_2$, etc., so $|T| = 4$

With asymmetric dirichlet prior for each $p(t_i|d)$ with $\alpha = (5, ½, ½, ½)^T$,

- so, t_1 is more prominent and corresponds to a typical „stop word topic",

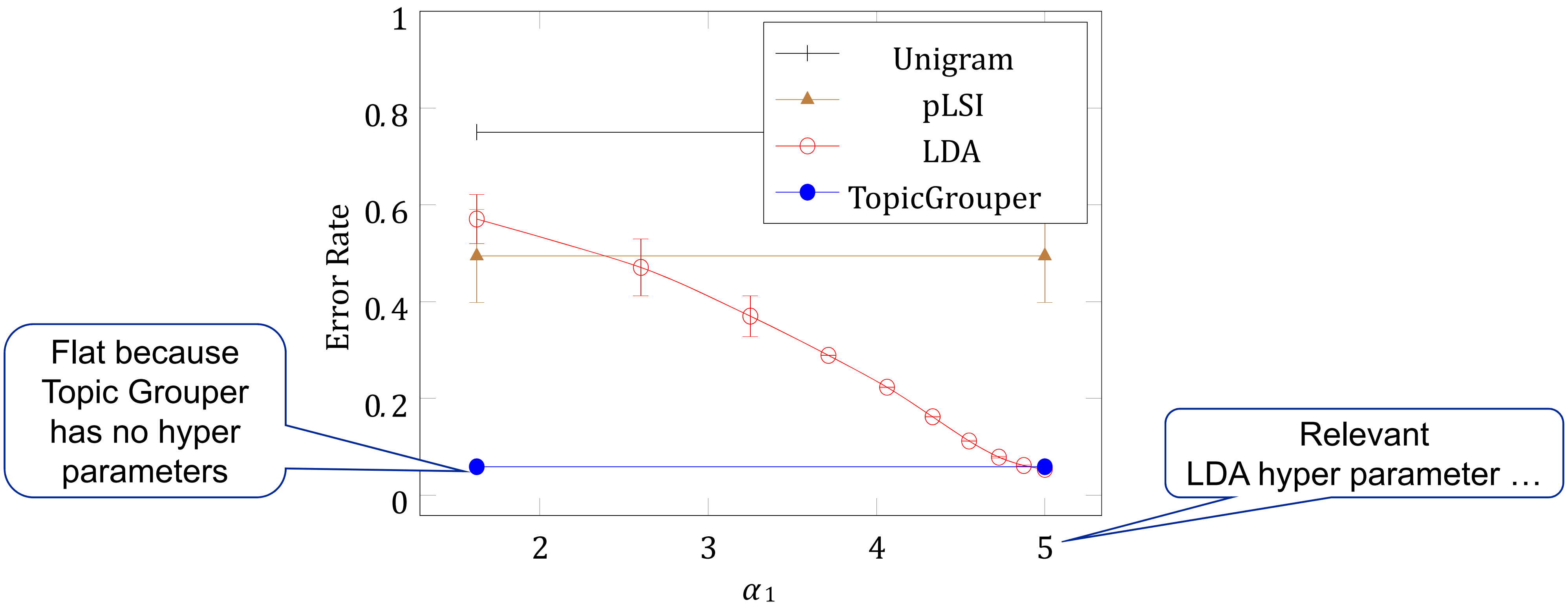
and symmetric dirichlet for $p(w|t_i(w))$ with $\beta = 1/100$

Documents of size 30 are (randomly) generated accordingly

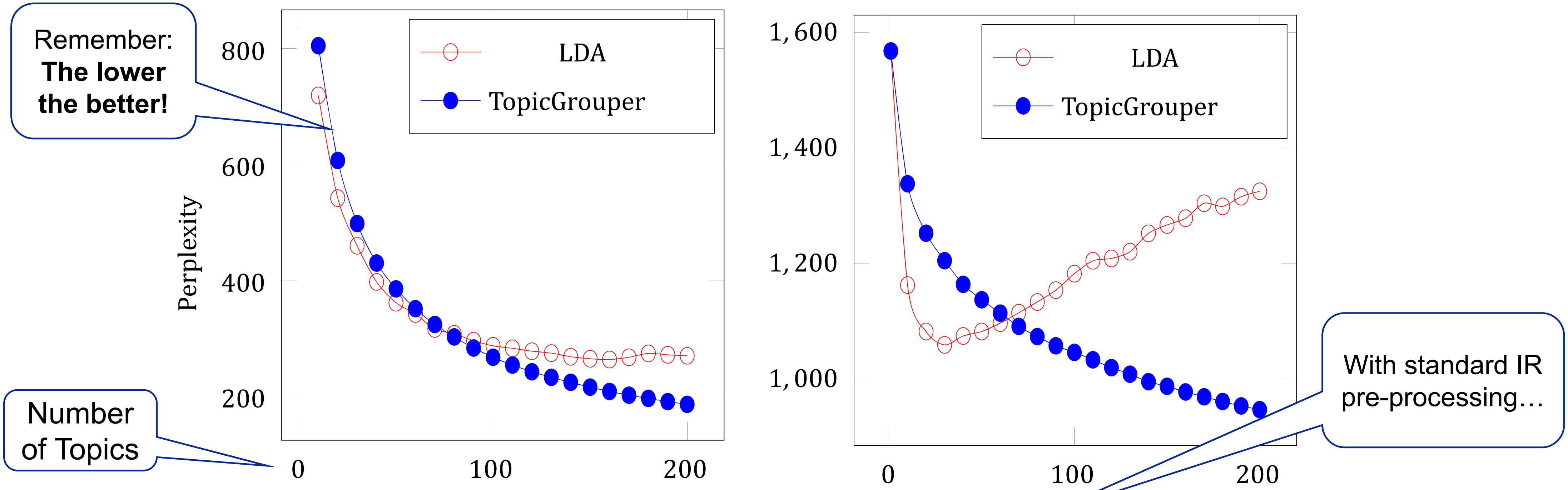
→ **The learning task is to (hopefully) recover the original topics from D!**

→ **Since the ground truth is known, we can use „error rate" (as defined in the paper)**

ERROR RATE ON SYNTHETIC DATASET FROM [TAN ET AL.]



PERPLEXITY ON A RETAIL AND A TEXT DATASET



Online Retail Dataset, |*D*| = *17,065*, |*V*| = *3,464*

NIPS Dataset, |*D*| = *1,500*, |*V*| = *8,801*

EXAMPLE

- TREC AP Corpus extract containing 20,000 newswire articles (from the 80‘s)
- $|V| = 25{,}047$ (stemming, stop word filtering etc.)

In the following:

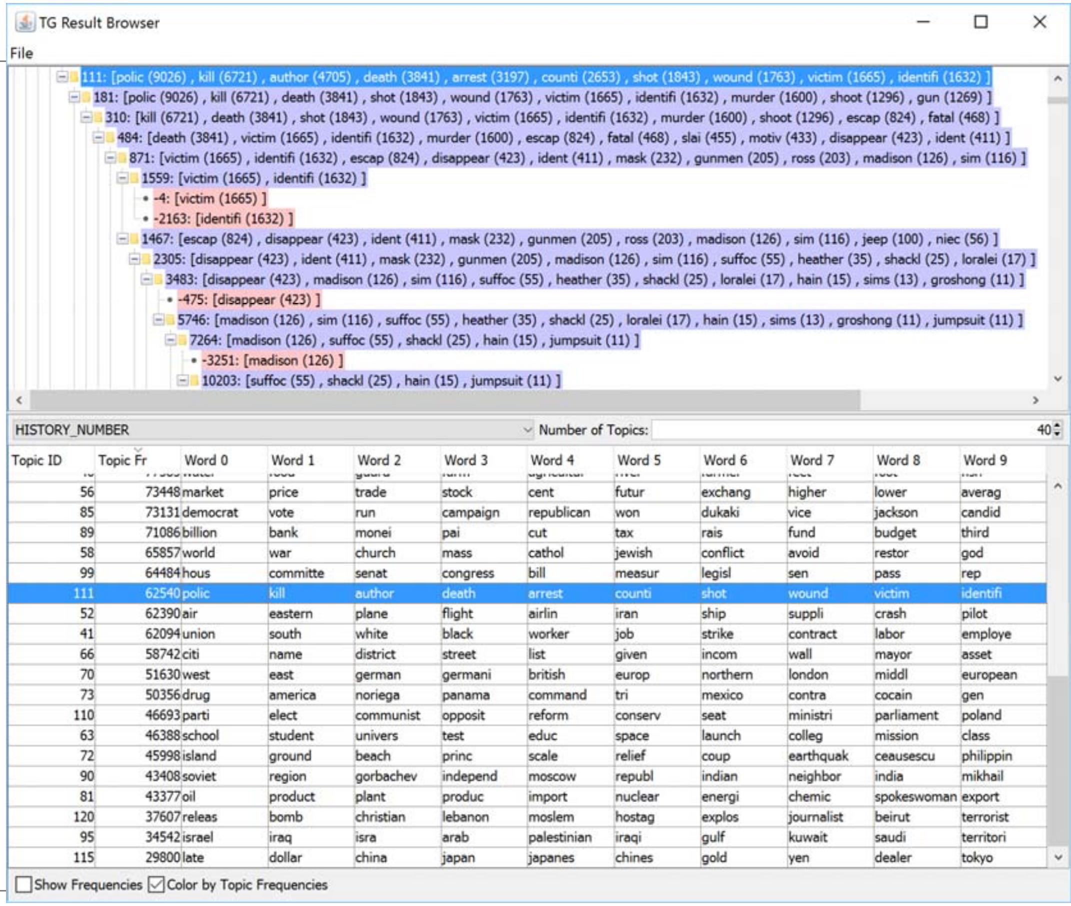
- „Flat“ view of topics at $|T(n)| = 40$
- Hierarchical result view

$f(t)$	**Top Seven Words per Topic** t					
538739	year	new	two	dai	week	three	month
305812	state	govern	nation	unit	american	includ	countri
281349	said	report	offici	sai	befor	against	told
176138	court	feder	charg	law	case	rule	order
119423	percent	down	rate	increas	industri	econom	point
115641	presid	bush	plan	meet	talk	administr	propos
112332	home	live	found	famili	man	children	life
96161	commun	visit	miss	travel	becam	histori	art
89151	call	show	newspap	appear	televis	radio	publish
82919	john	william	robert	richard	paul	wait	king
77385	water	food	guard	farm	agricultur	river	farmer
73131	democrat	vote	run	campaign	republican	won	dukaki
65857	world	war	church	mass	cathol	jewish	conflict
62540	polic	kill	author	death	arrest	counti	shot
62094	union	south	white	black	worker	job	strike
51630	west	east	german	germani	british	europ	northern
46693	parti	elect	communist	opposit	reform	conserv	seat
45998	island	ground	beach	princ	scale	relief	coup
43377	oil	product	plant	produc	import	nuclear	energi
34542	israel	iraq	isra	arab	palestinian	iraqi	gulf

Every Second Topic at $|T(n)|$ = 40 Sorted by Frequency for the AP Corpus Dataset

HEILBRONN UNIVERSITY OF APPLIED SCIENCES

DEMO: MODEL EXPLORATION BASED ON A SIMPLE VIEWING TOOL



Topic ID	Topic Fr	Word 0	Word 1	Word 2	Word 3	Word 4	Word 5	Word 6	Word 7	Word 8	Word 9
56	73448	market	price	trade	stock	cent	futur	exchang	higher	lower	averag
85	73131	democrat	vote	run	campaign	republican	won	dukaki	vice	jackson	candid
89	71086	billion	bank	monei	pai	cut	tax	rais	fund	budget	third
58	65857	world	war	church	mass	cathol	jewish	conflict	avoid	restor	god
99	64484	hous	committe	senat	congress	bill	measur	legisl	sen	pass	rep
111	62540	polic	kill	author	death	arrest	counti	shot	wound	victim	identifi
52	62390	air	eastern	plane	flight	airlin	iran	ship	suppli	crash	pilot
41	62094	union	south	white	black	worker	job	strike	contract	labor	employe
66	58742	citi	name	district	street	list	given	incom	wall	mayor	asset
70	51630	west	east	german	germani	british	europ	northern	london	middl	european
73	50356	drug	america	noriega	panama	command	tri	mexico	contra	cocain	gen
110	46693	parti	elect	communist	opposit	reform	conserv	seat	ministri	parliament	poland
63	46388	school	student	univers	test	educ	space	launch	colleg	mission	class
72	45998	island	ground	beach	princ	scale	relief	coup	earthquak	ceausescu	philippin
90	43408	soviet	region	gorbachev	independ	moscow	republ	indian	neighbor	india	mikhail
81	43377	oil	product	plant	produc	import	nuclear	energi	chemic	spokeswoman	export
120	37607	releas	bomb	christian	lebanon	moslem	hostag	explos	journalist	beirut	terrorist
95	34542	israel	iraq	isra	arab	palestinian	iraqi	gulf	kuwait	saudi	territori
115	29800	late	dollar	china	japan	japanes	chines	gold	yen	dealer	tokyo

A (SURELY BIASED) LIST OF PROS AND CONS

Cons:

- **„Each word in exactly one topic“ is a serious limitation for polysemic words and multiple topical contexts of a word**
- Extrinsic evaluation, e.g. based on human assessment, is still missing [CBG+09, NLGB10, LNB14]

Pros:

- No hyper parameter hell → „Just click and run …“
- No stop word / function word pollution of topics
- „Well behaved“ in practice (according to our findings) → „It does about what you want…“
- Deep hierarchies of topics and also |V| „flat“ topic views (you may choose…)
- Apparently useful hierarchies
- Reasonably efficient, **but** vocabulary size is critical for runtime and memory consumption – so filter it
- Exploration of hierarchical model as in demo seems useful

MORE STUFF ...

- An extended version of the paper is available on Arxiv:

  https://arxiv.org/abs/1904.06483

- We devised an improved algorithm with **expected complexity in $O(|V|^2 |D|)$ but space only in $O(|V|)$**

- An implementation in Java along with scripts for all related experiments is available on GitHub:

  https://github.com/pfeiferd/TopicGrouperJ

  (Its out there and usable but no documentation...)

THANK YOU.
QUESTIONS?
DISCUSSION...

This work is partially based on a collaboration between Refinitiv Labs London and Hochschule Heilbronn. Part of this work was carried out while the first author visited Refinitiv on a sabbatical.

REFERENCES (1)

AWST09. Arthur Asuncion, Max Welling, Padhraic Smyth, and Yee Whye Teh. On smoothing and inference for topic models. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09, pages 27–34, Arlington, VA, USA, 2009. AUAI Press.

BJGT03. David M. Blei, Michael I. Jordan, Thomas L. Griffiths, and Joshua B. Tenenbaum. Hierarchical topic models and the nested Chinese restaurant process. In *Proceedings of the 16th International Conference on Neural Information Processing Systems*, NIPS '03, pages 17–24, Cambridge, MA, USA, 2003. MIT Press.

Ble12. David M. Blei. Probabilistic topic models. *Commun. ACM*, 55(4):77–84, 2012.

BNJ03. David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, 2003.

CBG+09. Jonathan Chang, Jordan L. Boyd-Graber, Sean Gerrish, Chong Wang, and David M. Blei. Reading tea leaves: How humans interpret topic models. In Yoshua Bengio, Dale Schuurmans, John D. Lafferty, Christopher K. I. Williams, and Aron Culotta, editors, *Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems, 7-10 December 2009, Vancouver, British Columbia, Canada*, pages 288–296. Curran Associates, Inc., 2009.

GS04. Thomas L. Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(Suppl. 1):5228–5235, 2004.

REFERENCES (2)

Hof99. Thomas Hofmann. Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, UAI' 99, pages 289–296, San Francisco, CA, USA, 1999. Morgan Kaufmann.

KKKO12. Joon Hee Kim, Dongwoo Kim, Suin Kim, and Alice Oh. Modeling topic hierarchies with the recursive chinese restaurant process. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, pages 783–792, New York, NY, USA, 2012. ACM.

LNB14. Jey Han Lau, David Newman, and Timothy Baldwin. Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '14, pages 530–539, Gothenburg, Sweden, 2014. Association for Computational Linguistics.

MRS08. Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

NLGB10. David Newman, Jey Han Lau, Karl Grieser, and Timothy Baldwin. Automatic evaluation of topic coherence. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT, pages 100–108, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

PWBJ15. J. Paisley, C. Wang, D. M. Blei, and M. I. Jordan. Nested hierarchical dirichlet processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2):256–270, 2015.

TO10. Y. Tan and Z. Ou. Topic-weak-correlated latent Dirichlet allocation. In *7th International Symposium on Chinese Spoken Language Processing*, pages 224–228, 2010.

WMM09. Hanna M. Wallach, David M. Mimno, and Andrew McCallum. Rethinking LDA: Why priors matter. In Yoshua Bengio, Dale Schuurmans, John D. Lafferty, Christopher K. I. Williams, and Aron Culotta, editors, *NIPS*, pages 1973–1981. Curran Associates, Inc., 2009.