EMORY | GOIZUETA BUSINESS SCHOOL

TM-OKC: An Unsupervised Topic Model for Text in Online Knowledge Communities

Dongcheng Zhang
Emory University

Kunpeng Zhang (KZ)
University of Maryland

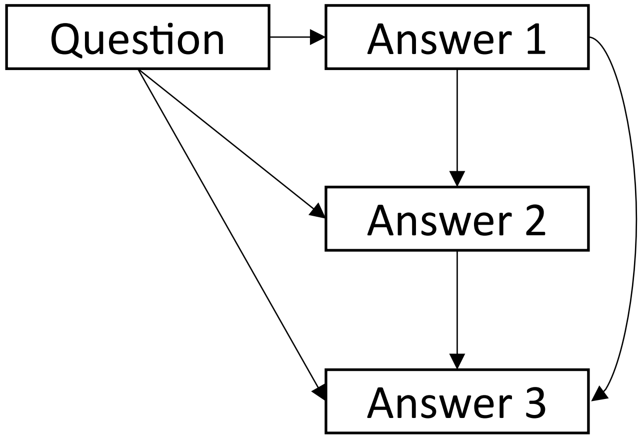
Yi Yang
HKUST

David A. Schweidel
Emory University

October 14, 2023

1 Research motivation

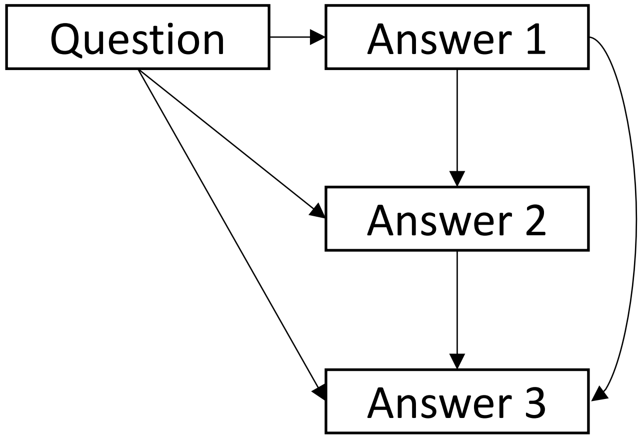
- **“Question-answer” textual data**



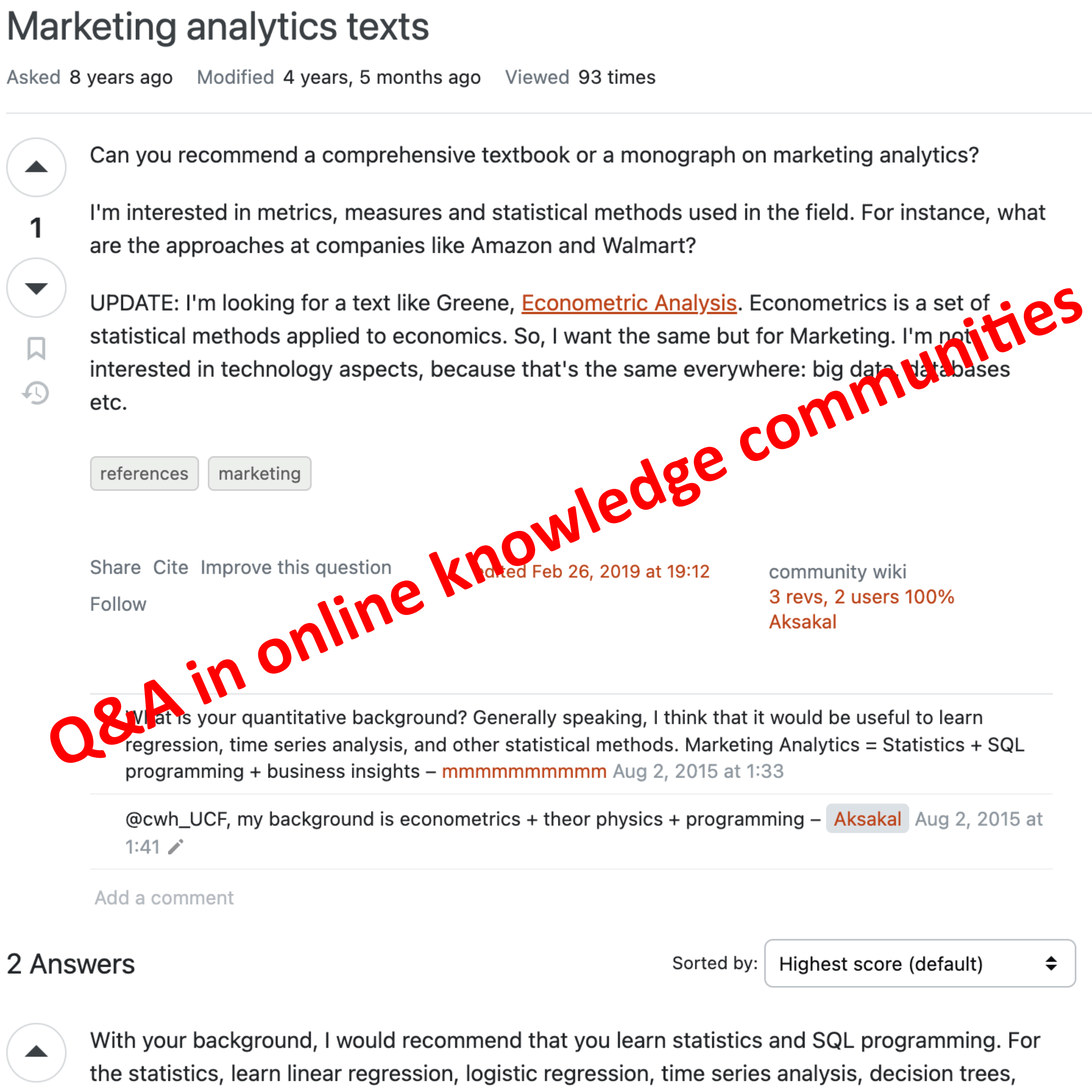
Note: “Question-answer” refers to a general structure

1 Research motivation

- "Question-answer" textual data

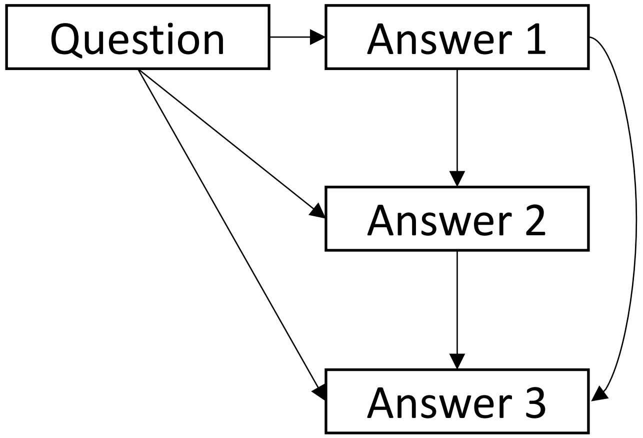


Note: "Question-answer" refers to a general structure

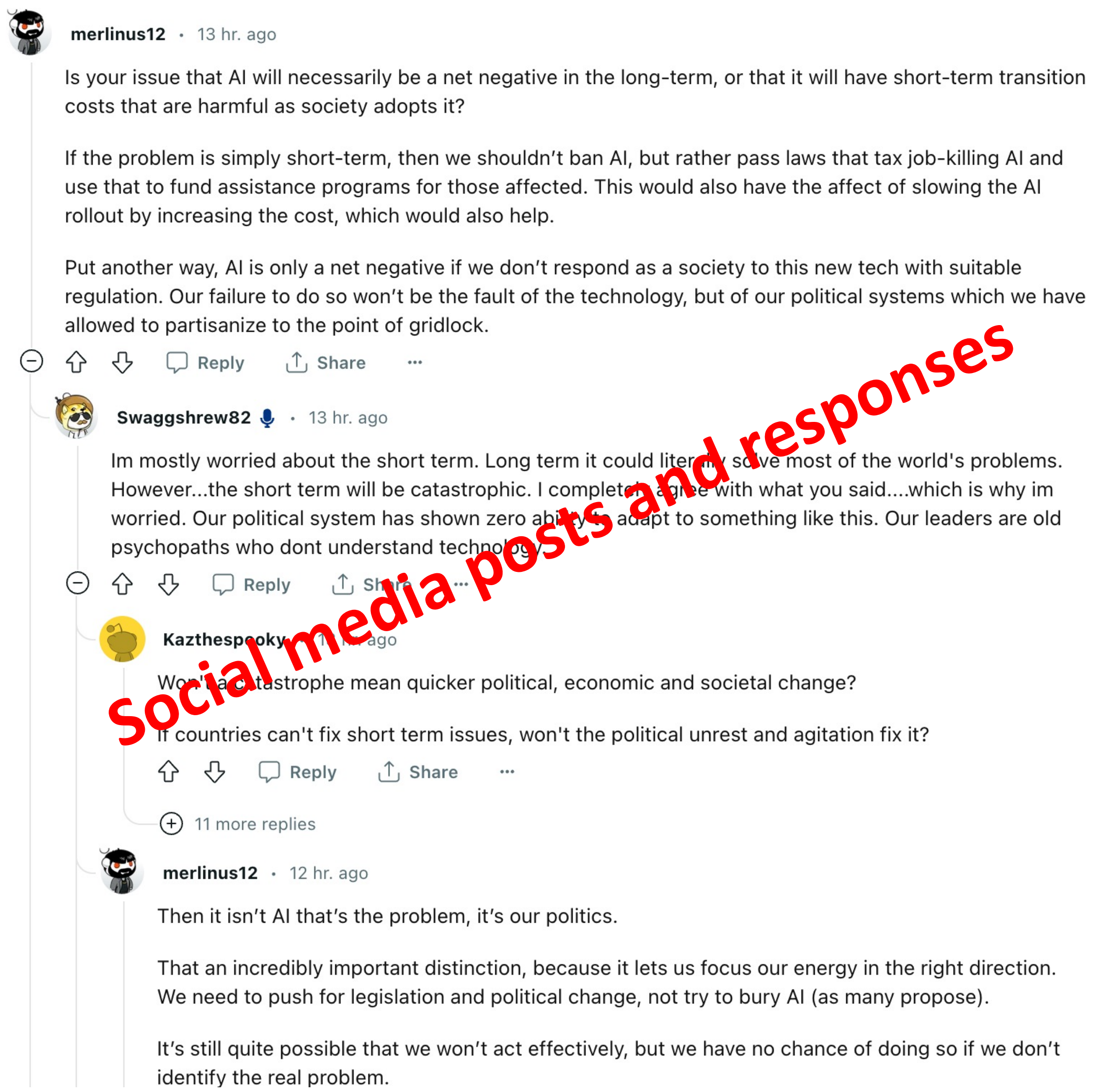


1 Research motivation

- **"Question-answer" textual data**

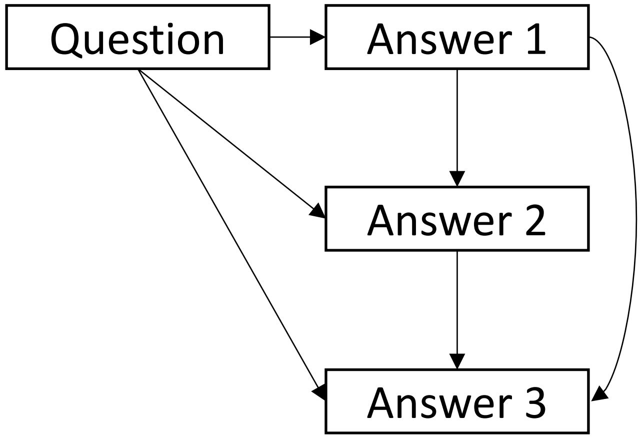


Note: "Question-answer" refers to a general structure



1 Research motivation

- “Question-answer” textual data

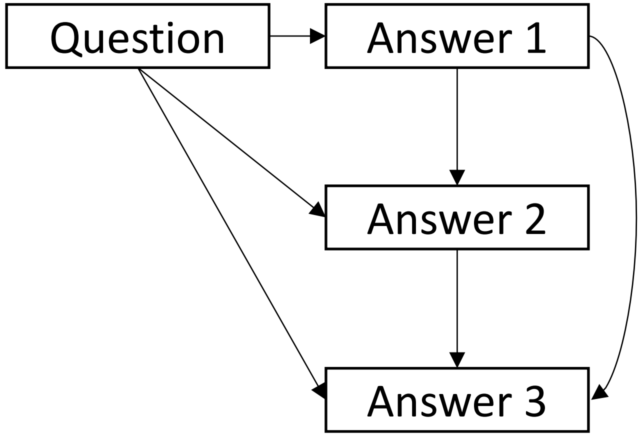


Note: “Question-answer” refers to a general structure

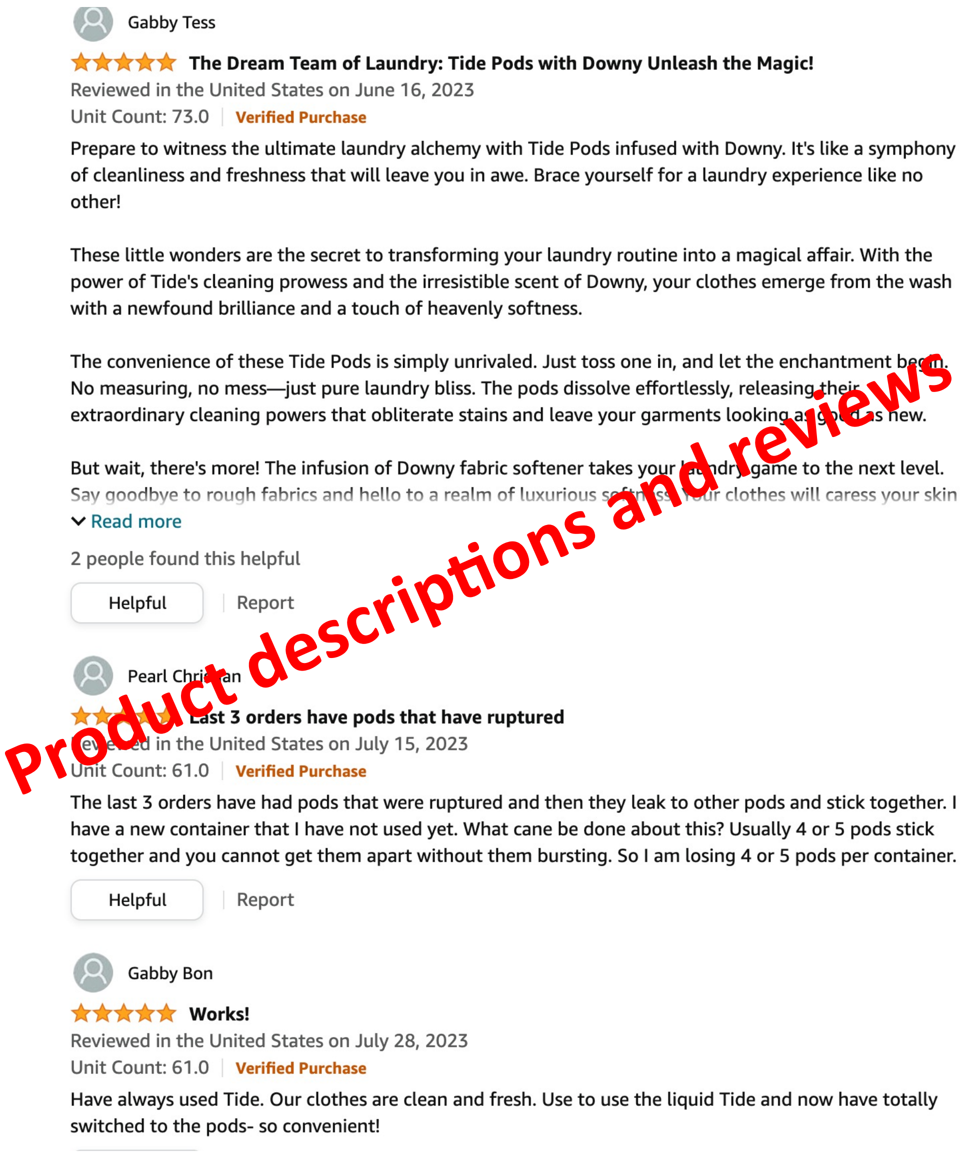
- **Millions of users**
  - Stack Exchange (100+M users), Quora (300+M users)
  - Twitter (300+M users), Reddit (400+M users)
- **Huge business opportunities**
  - **$ 65.31 billion** on social advertising (US) (eMarketer, 2022)

1 Research motivation

- **“Question-answer” textual data**

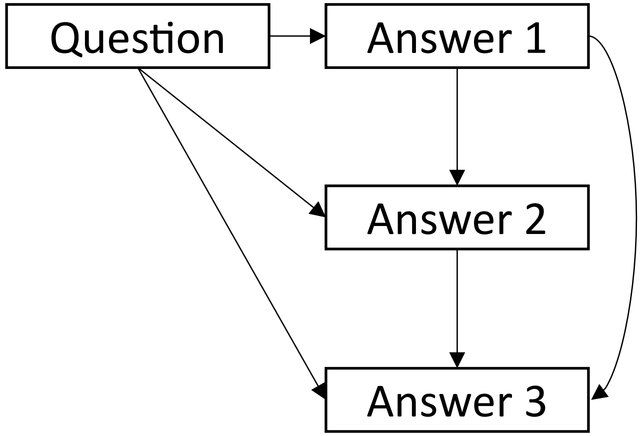


Note: “Question-answer” refers to a general structure



1 Research motivation

- “Question-answer” textual data



Note: “Question-answer” refers to a general structure

Y Hacker News new | past | comments | ask | show | jobs | submit

▲ Ask HN: Is GPT 4's quality lately worst than GPT 3.5?

41 points by agonz253 11 hours ago | hide | past | favorite | 36 comments

Has anyone else encountered this phenomenon lately? I've found myself prompting GPT 3.5 with simple questions that GPT 4 provided an incorrect answer for, and lo and behold I get a much better answe

For ex this is GPT 4: https://chat.openai.com/share/e24501ad-8f1c-4b5a-a6d0-d933f5d1d209

And this is GPT 3.5: https://chat.openai.com/share/b9372bdc-ffff-4655-bee4-2b3f3c3b8285

In the latter case I didn't even need to ask for the order by clause as it anticipates it and provides an answer for it. GPT 4's first answer was wrong.

In the past two days I've seen at least 2 other cases where GPT 4's answer was plain wrong and GPT 3.5's was not only correct but of very high quality, reminding me of what I first felt when using GPT 4 the first time.

add comment

▲ muzani 25 minutes ago | next [–]

My observation (ChatGPT and not the API models):

For code, 3.5 is superior. 3.5 allows for about 21k tokens of input, while GPT 4 allows for around 10k. This also makes it a lot better for boilerplate work as at it can take a lot more input, and handles long conversations and iterations better.

Brainstorming, 4 is better. It's capable of some top tier brainstorming and [illegible] argues back quite frequently.

Unguided creative writing (describe a potato), they're roughly equal.

Guided creative writing (i.e. write a story around (4[illegible]0 words of requirements)), 4 is much better.

Poems and wordplay, 4 absolutely floors 3.5 [illegible] vocabulary and it's able to do rhymes and alliterations better, which humans are usually bad at.

For reasoning and riddles, 4 is still benchmark among the LLMs.

I really dislike that they named it GPT-3.5 instead of something like "Glide". It implies that it's inferior to 4, when they're just suited for different things.

reply

▲ kromem 5 hours ago | prev | next [–]

Yes, and I'm willing to bet that within 12 months we'll be looking back realizing that this was due to the fine tuning taking the world's SorA pretrained model aligned with "completing human tax" and putting it in the box of "you are an AI without feelings or desires tasked with XYZ."

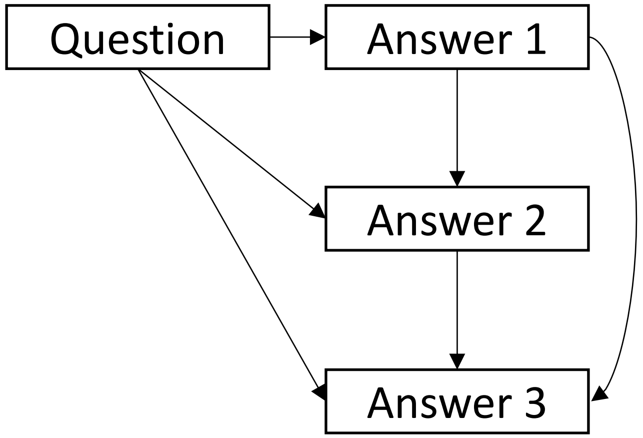
The search space on the fine tuned GPT-3.5 chat models versus the foundational Davinci text completion model is MUCH more narrow, particularly in starting off.

Even with the same temperature, you'll see any marketing-style prompt for chat begin with "Introducing XYZ..." around 30% of the time as if it's a junior door to door salesman, whereas the foundational model doesn't have any single intro that common across runs and generally employs a much broader vocabulary set.

We saw Google shoot Lambda in the foot after Blake's press tour which set them behind the next round of competition.

1 Research motivation

- **“Question-answer” textual data**



Note: “Question-answer” refers to a general structure

How to extract information from texts?

- **User interests’ discovery**
- **Consumer behavior analysis**
- **Empirical IS research**

1 Research motivation

- **Topic models**

Document “Summarize” ⇒ **Interpretable topics**

Example: … my recent Starbucks experience … delightful taste … rich, velvety caramel perfectly complemented the smooth espresso … nuanced flavors ... incredible service … genuine warmth and enthusiasm … passion for coffee … attention to detail … every step executed with precision … danced on my taste buds … memorable journey … marriage of taste and service.

⇓ Topic modeling

Topic 1 (42%): service

Topic 2 (29%): taste

Other topics … … …

1 Research motivation

- **Topic models**

Each topic interpreted by a word distribution

Document $\xrightarrow{\text{"Summarize"}}$ **Interpretable topics**

Example: … my recent Starbucks experience … delightful taste … rich, velvety caramel perfectly complemented the smooth espresso … nuanced flavors … incredible service … genuine warmth and enthusiasm … passion for coffee … attention to detail … every step executed with precision … danced on my taste buds … memorable journey … marriage of taste and service.

Topic modeling

Topic 1 (42%): service	
Service	0.069
Experience	0.041
Journey	0.039
Warmth	0.029
Enthusiasm	0.018

Topic 2 (29%): taste	
Taste	0.180
Flavor	0.049
Smooth	0.042
Rich	0.033
Nuance	0.027

Other topics … … …

1 Research motivation

- **Topic models**

Document → "Summarize" → **Interpretable topics** → **Downstream applications**

User profiling

Friendship suggestion

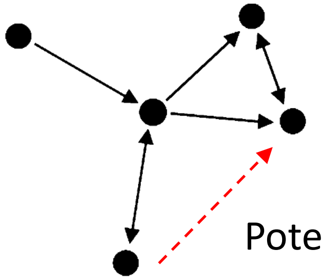
… … …

Trending topic detection

User interests' discovery

User A: football fan
User B: foodie
… …

(Geva et al. 2019, *MIS Quarterly*)



Potential friend

(Li et al. 2016, *Management Science*)

Analyzing customer reviews

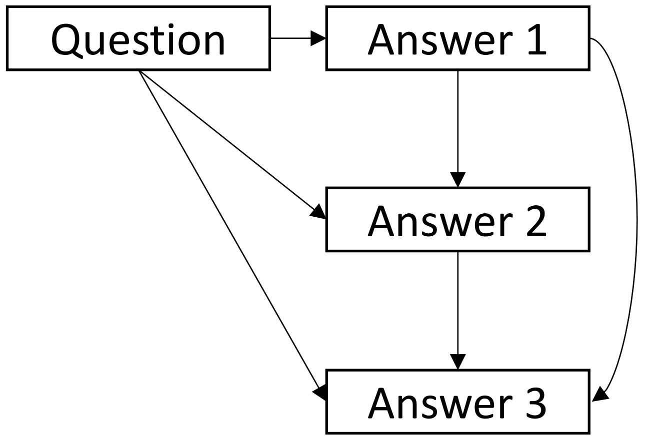
- CRM
- Brand perception
- Product design

(Büschken and Allenby 2020, *Marketing Science*)

1 Research motivation

- **Topic models**
  - Existing generic topic models (e.g., LDA)
    - Posts are independent
    - No structural relationships

Observed structure



Question: … data I am working with is not 100% clean ... What are the best practices in such a case as this?

Answer A: 1. Do not modify the original data. Having the original data source intact is important ...

Answer B: … I agree with the other posters that you should not modify the original data, but add fields for corrected values. I developed a technique in our systems (opengeocode.org) …

Goal: Develop a new topic model to capture the complex structural relations

2 Model development: Key features

- **Three key features**

(1) Q&A relationship

Dependency: $Q \longrightarrow A$ + Variation: $Q \neq A$

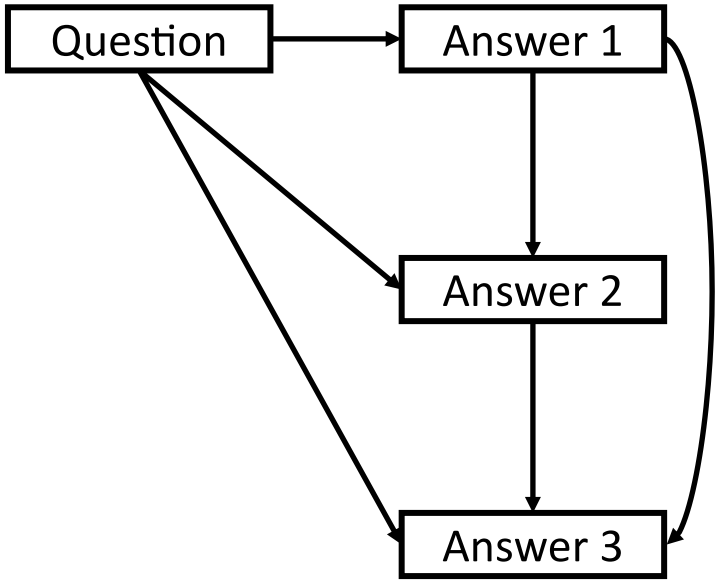
(2) Threaded structure + heterogeneity

Depend on prior answers? Yes → Follow-up answer; No → Novel answer

(3) Heterogeneous impacts

Question → Current answer

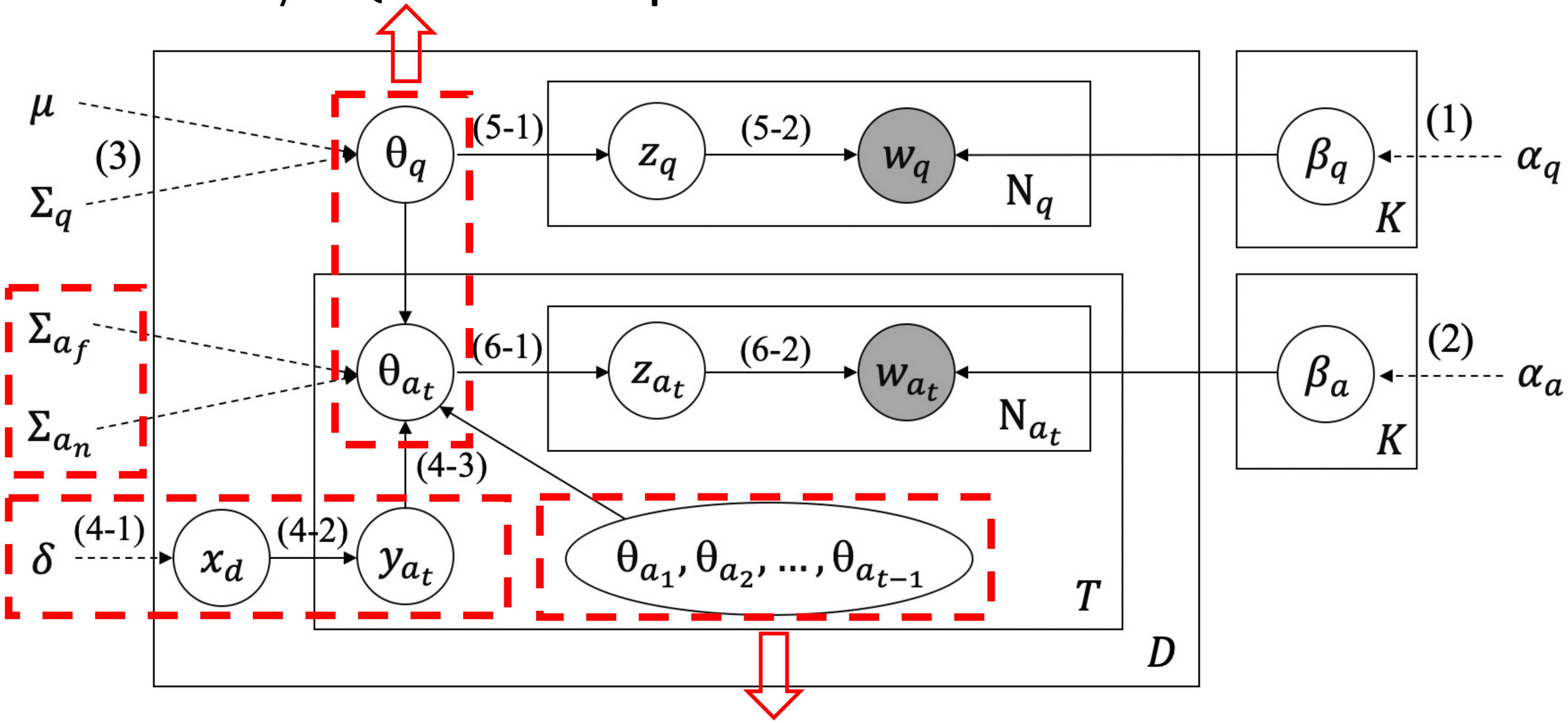
Prior answers → Current answer



2 Model development: Our novel framework

- **Develop a novel unsupervised Bayesian topic modeling framework**

Key 1: **Q&A relationship**



Key 3: **Heterogeneity**

Key 2: **threaded structure**

2 Model development: Inference and estimation

- **Complex model inference and estimation under high interdependencies**
  - Model inference
    - Variational mean-field inference and coordinate ascent algorithm
    - Maximize the evidence lower bound (ELBO)

$$
\begin{aligned}
\log p(\boldsymbol{w_q},\boldsymbol{w_{a_{1:T}}};\boldsymbol{\mu},\boldsymbol{\Sigma_q},\boldsymbol{\Sigma_{a_f}},\boldsymbol{\Sigma_{a_n}},\gamma,\boldsymbol{\delta},\boldsymbol{\alpha_q},\boldsymbol{\alpha_a}) \geq{}& E_u[\log p(\boldsymbol{\eta_q};\boldsymbol{\mu},\boldsymbol{\Sigma_q})] + \sum_{n_q=1}^{N_q} E_u\left[\log p\left(z_q^{n_q} \middle| \boldsymbol{\eta_q}\right)\right] \\
&+ \sum_{n_q=1}^{N_q} E_u\left[\log p\left(w_q^{n_q} \middle| z_q^{n_q},\boldsymbol{\beta_q^{1:K}}\right)\right] + \sum_{k=1}^{K} E_u[\log p(\boldsymbol{\beta_q^k};\boldsymbol{\alpha_q})] + E_u[\log p(\boldsymbol{x_d};\boldsymbol{\delta})] \\
&+ \sum_{t=1}^{T} E_u[\log p(\boldsymbol{y_{a_t}} \mid \boldsymbol{x_d})] + \sum_{t=1}^{T} E_u[\log p(\boldsymbol{\eta_{a_t}} \mid \boldsymbol{\eta_q},\boldsymbol{\bar{\eta}_{a_{t-1}}},\boldsymbol{y_{a_t}};\boldsymbol{\Sigma_{a_f}},\boldsymbol{\Sigma_{a_n}},\gamma)] \\
&+ \sum_{t=1}^{T}\sum_{n_{a_t}=1}^{N_{a_t}} E_u\left[\log p\left(z_{a_t}^{n_{a_t}} \middle| \boldsymbol{\eta_{a_t}}\right)\right] + \sum_{t=1}^{T}\sum_{n_{a_t}=1}^{N_{a_t}} E_u\left[\log p\left(w_{a_t}^{n_{a_t}} \middle| z_{a_t}^{n_{a_t}},\boldsymbol{\beta_a^{1:K}}\right)\right] \\
&+ \sum_{k=1}^{K} E_u[\log p(\boldsymbol{\beta_a^k};\boldsymbol{\alpha_a})] + H(u),
\end{aligned}
$$

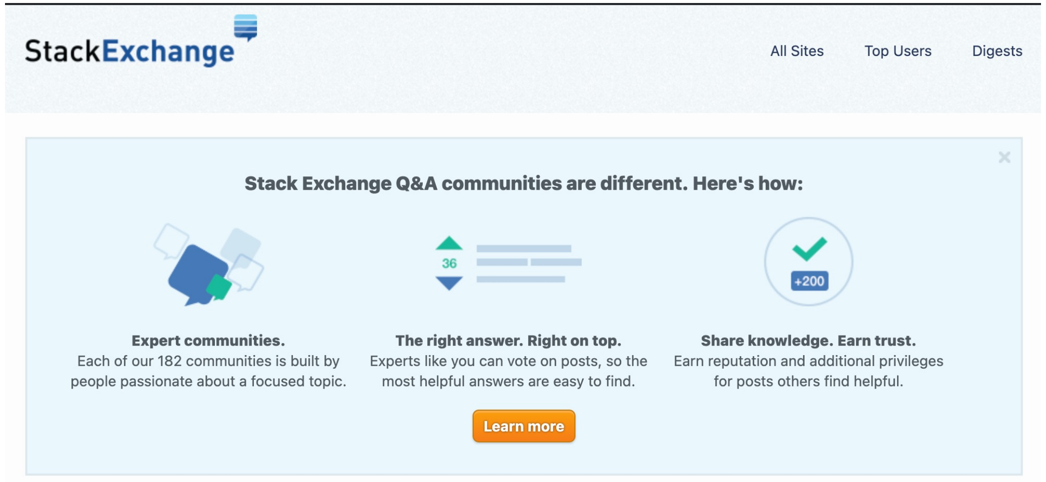
$$
\begin{aligned}
&u(\boldsymbol{\eta_q},\boldsymbol{\eta_{a_{1:T}}},\boldsymbol{x_d},\boldsymbol{y_{a_{1:T}}},\boldsymbol{z_q},\boldsymbol{z_{a_{1:T}}},\boldsymbol{\beta_q},\boldsymbol{\beta_a}) \\
&= \prod_{k=1}^{K} u\left(\eta_q^k;\lambda_q^k,(\sigma_q^k)^2\right) \prod_{t=1}^{T}\prod_{k=1}^{K} u\left(\eta_{a_t}^k;\lambda_{a_t}^k,(\sigma_{a_t}^k)^2\right) u(\boldsymbol{x_d};\boldsymbol{\nu_d}) \prod_{t=1}^{T} u(\boldsymbol{y_{a_t}};\boldsymbol{\psi_{a_t}}) \\
&\prod_{n_q=1}^{N_q} u(z_q^{n_q};\boldsymbol{\phi_q^{n_q}}) \prod_{t=1}^{T}\prod_{n_{a_t}=1}^{N_{a_t}} u\left(z_{a_t}^{n_{a_t}};\boldsymbol{\phi_{a_t}^{n_{a_t}}}\right) \prod_{k=1}^{K} u(\boldsymbol{\beta_q^k};\boldsymbol{\tau_q^k}) \prod_{k=1}^{K} u(\boldsymbol{\beta_a^k};\boldsymbol{\tau_a^k}).
\end{aligned}
$$

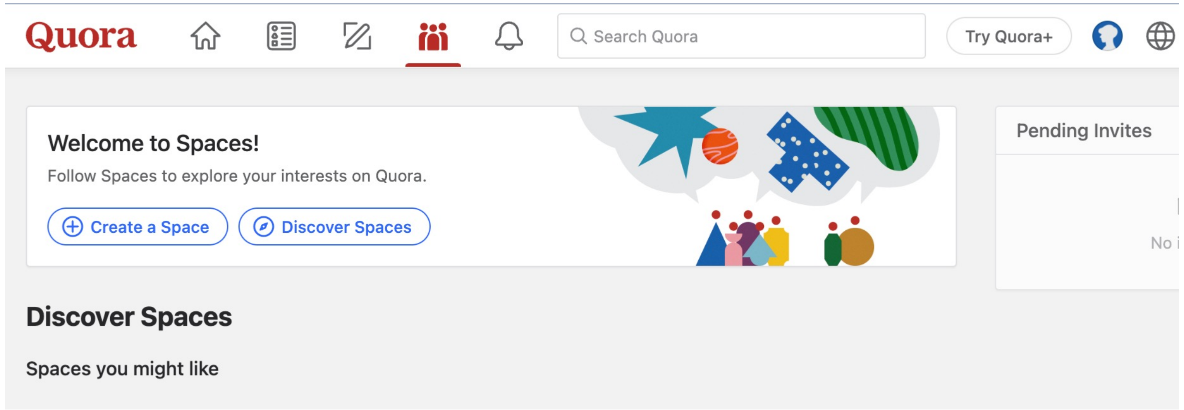
  - Parameter estimation
    - Variational expectation-maximization (VEM)
    - Maximize the likelihood bound

$$
\begin{aligned}
&L\left(\boldsymbol{\mu},\boldsymbol{\Sigma_q},\boldsymbol{\Sigma_{a_f}},\boldsymbol{\Sigma_{a_n}},\gamma;\boldsymbol{w_{1:D,q}},\boldsymbol{w_{1:D,a_{1:T}}}\right) \geq \hat{L} \\
&= \sum_{d=1}^{D}\{E_{u_d}[\log p(\boldsymbol{\eta_{d,q}},\boldsymbol{\eta_{d,a_{1:T}}},\boldsymbol{x_d},\boldsymbol{y_{d,a_{1:T}}},\boldsymbol{z_{d,q}},\boldsymbol{z_{d,a_{1:T}}},\boldsymbol{\beta_q},\boldsymbol{\beta_a},\boldsymbol{w_{d,q}},\boldsymbol{w_{d,a_{1:T}}})] \\
&+ H(u_d)\}.
\end{aligned}
$$

3 Model evaluation: Datasets

- **Real-world datasets**
  - Stack Exchange, 6 categories
  - Quora, 3 categories





3 Model evaluation: Datasets

- **Real-world datasets**
  - Stack Exchange, 6 categories
  - Quora, 3 categories

Number of Q&A threads: 1,000 ~ 100,000

Descriptive statistics of the Stack Exchange dataset

Section (Category)	# of questions	# of answers	# of words per question/answer
Technology (Data Science)	20740	31181	73.10
Culture/Recreation (English Language & Usage)	109977	268356	47.45
Life/Arts (Cooking)	23387	58078	57.49
Science (Computer Science)	31156	45807	72.49
Professional (Writing)	10429	34116	77.61
Business (Project Management)	5782	17724	70.26

Descriptive statistics of the Quora dataset

Section (Category)	# of questions	# of answers	# of words per question/answer
Science and Technology	1471	3144	28.82
Business and Marketing	1140	1981	23.50
Health and Life	2281	4144	20.18

3 Model evaluation: Statistical model fit

- Perplexity: based on **log-likelihood** (Blei et al. 2003; Roberts et al. 2019)

How "perplexed" the model is?

$$perplexity = \exp\left\{-\frac{\log p(\boldsymbol{w})}{\sum_{d}^{|D_{test}|}\sum_{v=1}^{V} num_d^v}\right\}$$

Lower perplexity $\Rightarrow$ Higher likelihood $\Rightarrow$ Better model fit / generalizability

3 Model evaluation: Statistical model fit

- Perplexity: based on **log-likelihood** (Blei et al. 2003; Roberts et al. 2019)

Lower perplexity ⇨ Higher likelihood ⇨ Better model fit / generalizability

Perplexity comparison for the Stack Exchange dataset

Model	Technology (Data Science)	Culture/ Recreation (English Language & Usage)	Life/Arts (Cooking)	Science (Computer Science)	Professional (Writing)	Business (Project Management)
LDA	1152.11	2181.84	1520.59	1063.53	1726.48	1087.31
NTM	1021.39	2176.38	1407.79	1028.45	1639.31	1063.06
TRTM	1029.95	2030.30	1296.18	976.34	1727.38	998.90
STM	1054.08	2015.43	1271.62	975.40	1718.43	1010.79
SCHOLAR	951.94	1586.73	1251.94	934.26	1615.01	1037.08
QATM	931.21	1625.43	1297.68	958.23	1479.63	939.54
LeadLDA1	1256.93	1702.28	1376.98	1063.31	1512.57	956.25
LeadLDA2	1100.50	1707.15	1326.48	1075.64	1498.23	1064.47
SITS	1008.46	1635.08	1280.41	979.68	1482.28	934.51
TM-OKC (mean)	**863.44***	1591.63	1220.97	891.59	1400.93	901.03
TM-OKC (decay)	863.93	**1582.44**	1221.01	**891.30***	1394.29	**899.39***
TM-OKC (weight)	867.69	1591.73	**1218.70***	892.54	**1390.96***	900.59

Perplexity comparison for the Quora dataset

Model	Science and Technology	Business and Marketing	Health and Life
LDA	1579.54	1698.37	1357.71
NTM	1204.54	1368.05	1083.70
TRTM	1131.22	1310.21	1104.22
STM	1117.93	1301.27	1109.51
SCHOLAR	1192.71	1321.77	1087.89
QATM	1033.00	1243.23	1065.07
LeadLDA1	1399.21	1227.56	1088.73
LeadLDA2	1350.26	1242.85	1132.91
SITS	1313.06	1264.70	1058.69
TM-OKC (mean)	**1014.30***	**1202.66***	1042.97
TM-OKC (decay)	1015.86	1212.19	1040.38
TM-OKC (weight)	1018.13	1215.98	**1032.61***

3 Model evaluation: Statistical model fit

- Coherence score
  - "Similarity" (word co-occurrences) of the top words within a topic
  - Higher score indicates the learned topics are more coherent (Syed and Spruit 2017; Dieng et al. 2020)

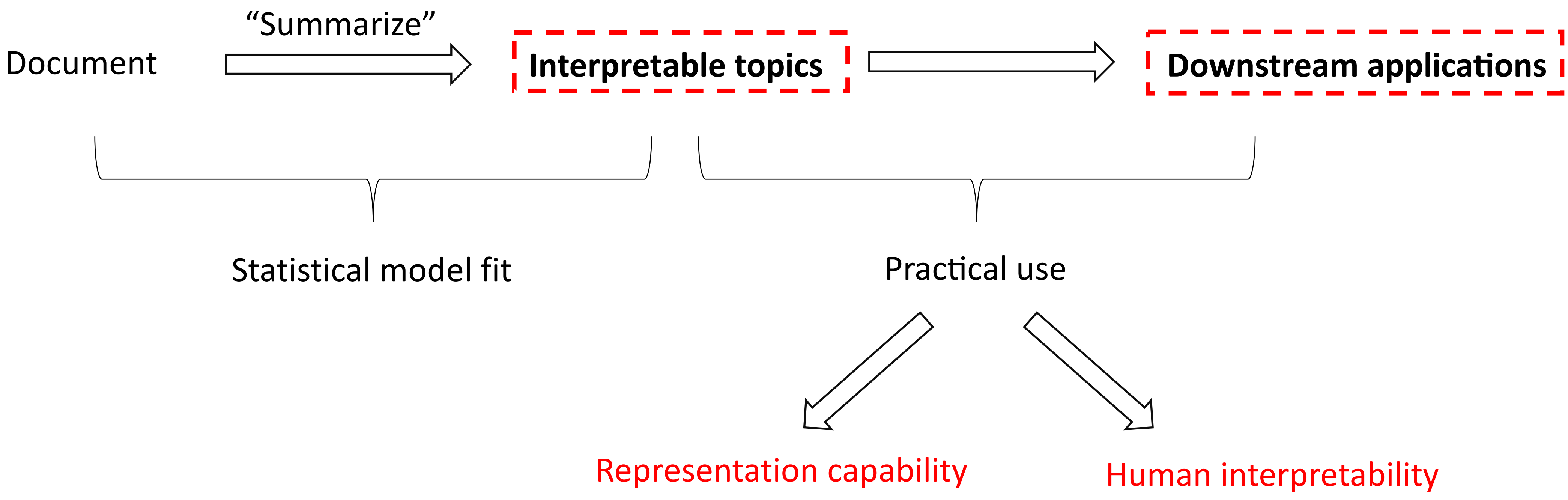
Coherence comparison for the Stack Exchange dataset

Model	**Technology (Data Science)**	**Culture/ Recreation (English Language & Usage)**	**Life/Arts (Cooking)**	**Science (Computer Science)**	**Professional (Writing)**	**Business (Project Management)**
LDA	0.51	0.42	0.47	0.47	0.37	0.37
NTM	0.53	0.43	0.47	0.48	0.37	0.39
TRTM	0.51	0.42	0.46	0.48	0.37	0.36
STM	0.53	0.41	0.48	0.49	**0.39**	0.37
SCHOLAR	0.53	0.43	0.48	0.49	0.37	0.37
QATM	0.52	0.42	0.48	0.48	0.37	**0.42**
LeadLDA1	0.50	0.40	0.47	0.47	0.35	0.36
LeadLDA2	0.51	0.40	0.46	0.48	0.36	0.37
SITS	0.52	0.41	0.47	0.48	0.38	0.39
TM-OKC (mean)	0.53	**0.45***	**0.50***	**0.51****	**0.39**	**0.42**
TM-OKC (decay)	0.53	0.43	**0.50***	**0.51****	0.38	0.41
TM-OKC (weight)	**0.56*****	0.44	**0.50***	**0.51****	0.38	0.41

Coherence comparison for the Quora dataset

Model	**Science and Technology**	**Business and Marketing**	**Health and Life**
LDA	0.40	0.39	0.41
NTM	0.46	0.41	0.42
TRTM	0.42	0.42	0.42
STM	0.43	0.43	0.43
SCHOLAR	0.45	0.40	0.42
QATM	0.43	0.44	0.42
LeadLDA1	0.46	0.38	0.44
LeadLDA2	0.41	0.39	0.42
SITS	0.43	0.41	0.43
TM-OKC (mean)	**0.48****	0.45	**0.47*****
TM-OKC (decay)	0.47	**0.46****	0.46
TM-OKC (weight)	**0.48****	0.45	0.46

3 Model evaluation: More direct evaluation



3 Model evaluation: Representation capability

- *Document classification* task (Zeng et al. 2019; Yang et al. 2022)

Topic vectors $\xrightarrow{\text{Predict}}$ Document category

Prediction accuracy of different methods

		Stack Exchange			Quora		
	Number of topics	40	80	120	40	80	120
Basic text feature extraction methods	TF-IDF features (top 100 words)		0.517			0.600	
	TF-IDF features (top 500 words)		0.717			0.779	
	TF-IDF features (top 1000 words)		0.718			0.804	
Bayesian topic modeling methods	LDA	0.727	0.710	0.699	0.728	0.726	0.712
	TRTM	0.891	0.885	0.893	0.935	0.920	0.937
	STM	0.886	0.847	0.888	0.935	0.935	0.930
	QATM	0.898	0.853	0.860	0.960	0.962	0.937
	LeadLDA1	0.867	0.832	0.751	0.762	0.749	0.747
	LeadLDA2	0.865	0.865	0.749	0.796	0.762	0.737
	SITS	0.903	0.891	0.868	0.881	0.893	0.876
Topic modeling combined with deep language models	NTM	0.905	0.900	0.859	0.813	0.827	0.848
	SCHOLAR	0.901	0.903	0.885	0.826	0.855	0.818
Representation learning methods	Bi-LSTM		0.822			0.919	
	Pre-trained BERT		0.696			0.722	
	Fine-tuned BERT		0.936			0.992	
Our method	TM-OKC	0.911^{*}	0.933^{***}	0.918^{***}	0.983^{**}	0.990^{***}	0.968^{***}

3 Model evaluation: Interpretability

- Lab studies
  - *word intrusion* and *topic intrusion* tasks (Chang et al. 2009; Bao and Datta 2014; Palese and Piccoli 2020)

consistency

Model $\Longleftrightarrow$ Human

Human evaluation results of *word intrusion* and *topic intrusion* tasks

	WIP_m in *word intrusion*			TLO_m in *topic intrusion*		
Number of topics	40	80	120	40	80	120
LDA	0.806	0.794	0.788	-1.61	-1.65	-1.85
NTM	0.825	0.813	0.819	-1.52	-1.47	-1.48
TRTM	0.813	0.838	0.806	-1.43	-1.35	-1.49
STM	0.819	0.825	0.819	-1.37	-1.25	-1.47
SCHOLAR	0.813	0.819	0.813	-1.36	-1.30	-1.51
QATM	0.825	0.831	0.813	-1.31	-1.22	-1.39
SITS	0.819	0.825	0.800	-1.32	-1.29	-1.38
LeadLDA1	0.813	0.806	0.800	-	-	-
LeadLDA2	0.819	0.813	0.794	-	-	-
TM-OKC	0.856**	0.869**	0.838*	-1.12**	-0.96***	-1.23**

4 Downstream applications

Statistical model fit

Representation capability

Interpretability

A better topic model

User profiling

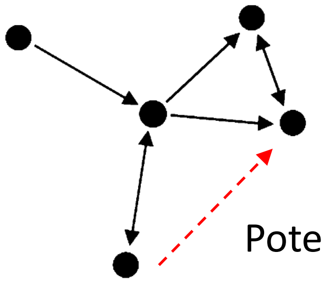
Friendship suggestion

… … …

Trending topic detection

User interests' discovery

User A: football fan
User B: foodie
… …



Potential friend

Analyzing customer reviews

- CRM
- Brand perception
- Product design

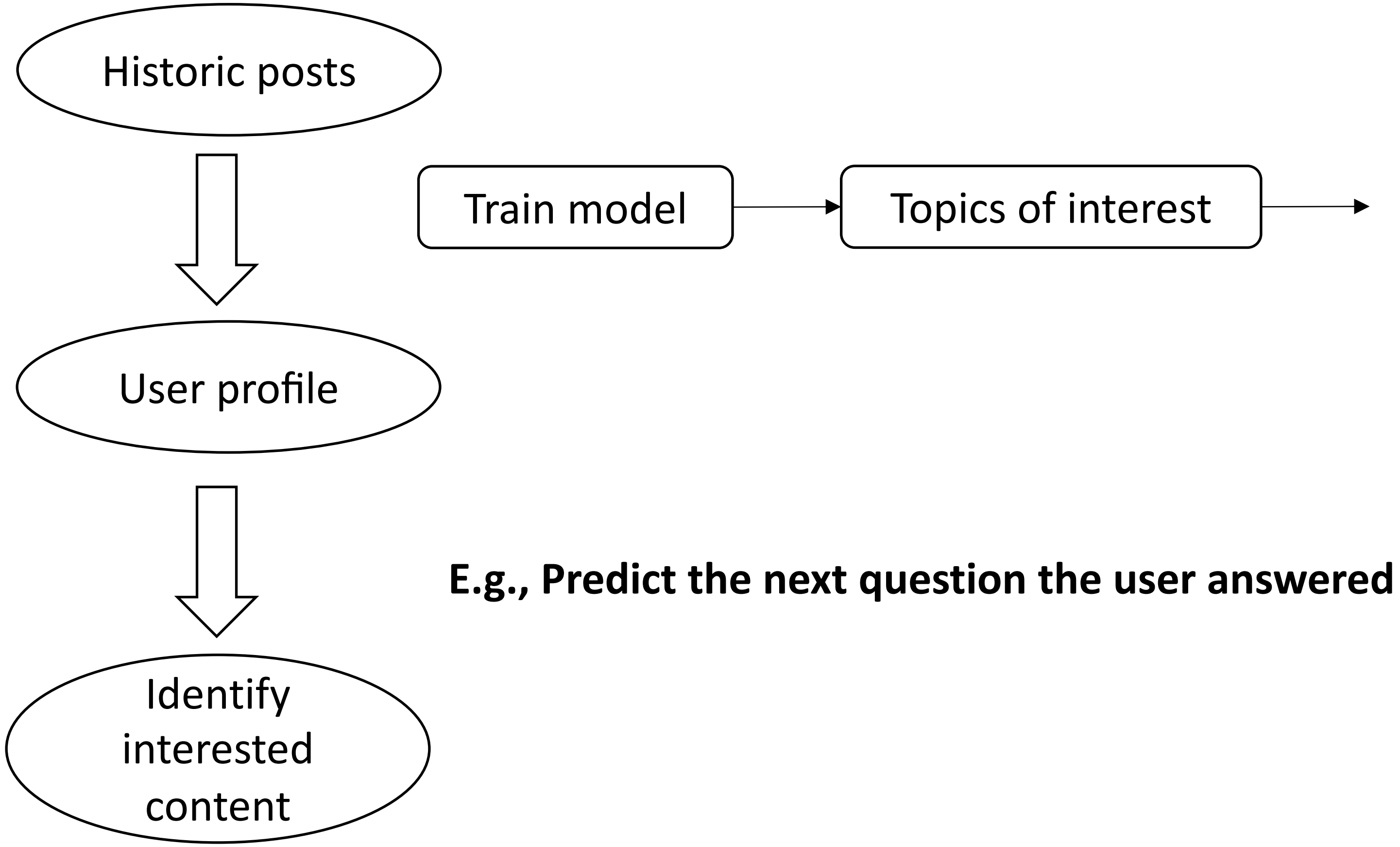
(Geva et al. 2019, *MIS Quarterly*)

(Li et al. 2016, *Management Science*)

(Büschken and Allenby 2020, *Marketing Science*)

4 Downstream applications

- **Example: user profiling**
  - User interests' discovery (He et al. 2017; Dhillon and Aral 2021)



User profile of me ☺

Machine learning	0.50
Economics of IS	0.30
…	…

4 Downstream applications

- **Example: user profiling**
  - User interests' discovery (He et al. 2017; Dhillon and Aral 2021)

User profiling performance comparison of different methods

		Hit rate for top K		
		K=5	K=10	K=20
Basic text feature extraction methods	TF-IDF features (top 100 words)	5.1%	11.7%	21.4%
	TF-IDF features (top 500 words)	5.3%	11.5%	21.2%
	TF-IDF features (top 1000 words)	5.5%	11.8%	21.6%
Bayesian topic modeling methods	LDA	20.9%	33.9%	49.7%
	TRTM	26.6%	43.7%	59.3%
	STM	26.5%	44.1%	59.4%
	QATM	26.5%	45.2%	60.1%
	LeadLDA1	6.2%	11.7%	22.1%
	LeadLDA2	6.5%	12.8%	24.1%
	SITS	26.0%	41.4%	58.3%
Pre-trained deep language models	Pre-trained BERT	8.6%	14.9%	26.6%
Topic modeling combined with deep language models	NTM	18.2%	32.1%	42.8%
	SCHOLAR	19.4%	33.6%	50.6%
Neural matrix factorization	NMF	19.0%	32.1%	46.0%
Our method	TM-OKC	32.6%***	49.1%**	64.8%**

		Data size N (number of Q&A threads)						
		1,000	2,000	4,000	8,000	16,000	32,000	64,000
Basic text feature extraction methods	TF-IDF features (top 100 words)	10.5%	10.7%	11.1%	11.7%	11.1%	12.4%	12.2%
	TF-IDF features (top 500 words)	10.3%	10.6%	11.5%	11.5%	11.8%	11.3%	11.5%
	TF-IDF features (top 1000 words)	10.6%	10.4%	11.9%	11.8%	11.2%	11.2%	11.7%
Bayesian topic modeling methods	LDA	26.9%	29.6%	31.5%	33.9%	34.8%	35.9%	36.4%
	TRTM	37.7%	39.8%	40.5%	43.7%	44.3%	45.1%	45.1%
	STM	36.4%	40.9%	42.0%	44.1%	44.9%	45.2%	45.6%
	QATM	30.0%	35.0%	40.0%	45.2%	45.7%	45.3%	46.0%
	LeadLDA1	10.2%	10.1%	10.7%	11.7%	12.2%	14.8%	15.2%
	LeadLDA2	10.5%	10.6%	10.6%	12.8%	12.1%	15.4%	15.9%
	SITS	34.5%	40.1%	39.8%	41.4%	43.9%	44.7%	45.8%
Pre-trained deep language models	Pre-trained BERT	13.3%	14.1%	13.4%	14.9%	14.6%	13.6%	14.1%
Topic modeling combined with deep language models	NTM	15.5%	23.6%	27.0%	32.1%	41.7%	47.2%	50.6%
	SCHOLAR	20.4%	22.5%	28.8%	33.6%	42.6%	47.8%	51.4%
Neural matrix factorization	NMF	20.8%	23.7%	27.8%	32.1%	40.8%	46.7%	49.8%
Our method	TM-OKC	46.8%***	47.7%***	48.0%***	49.1%**	49.7%**	50.1%*	51.3%

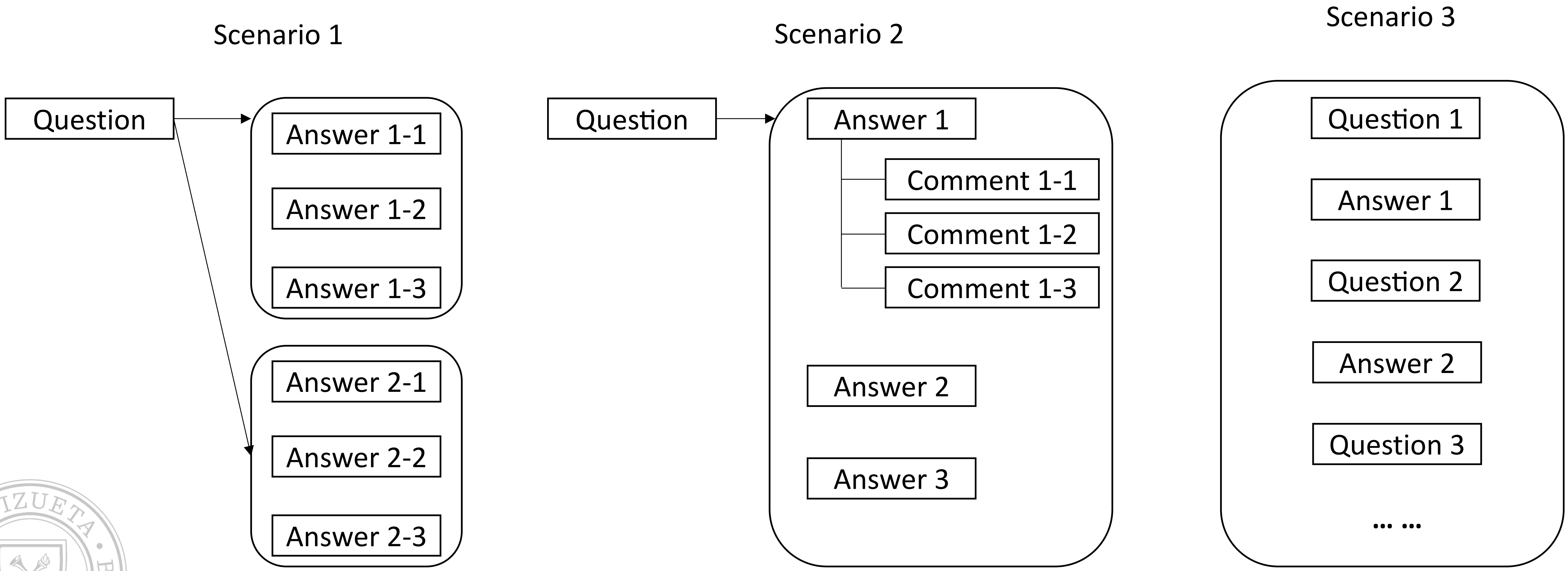
4 Downstream applications

Document $\xrightarrow{\text{“Summarize”}}$ **Interpretable topics** $\rightarrow$ **Practical applications**

Interpretable topics $\downarrow$ **Empirical research**

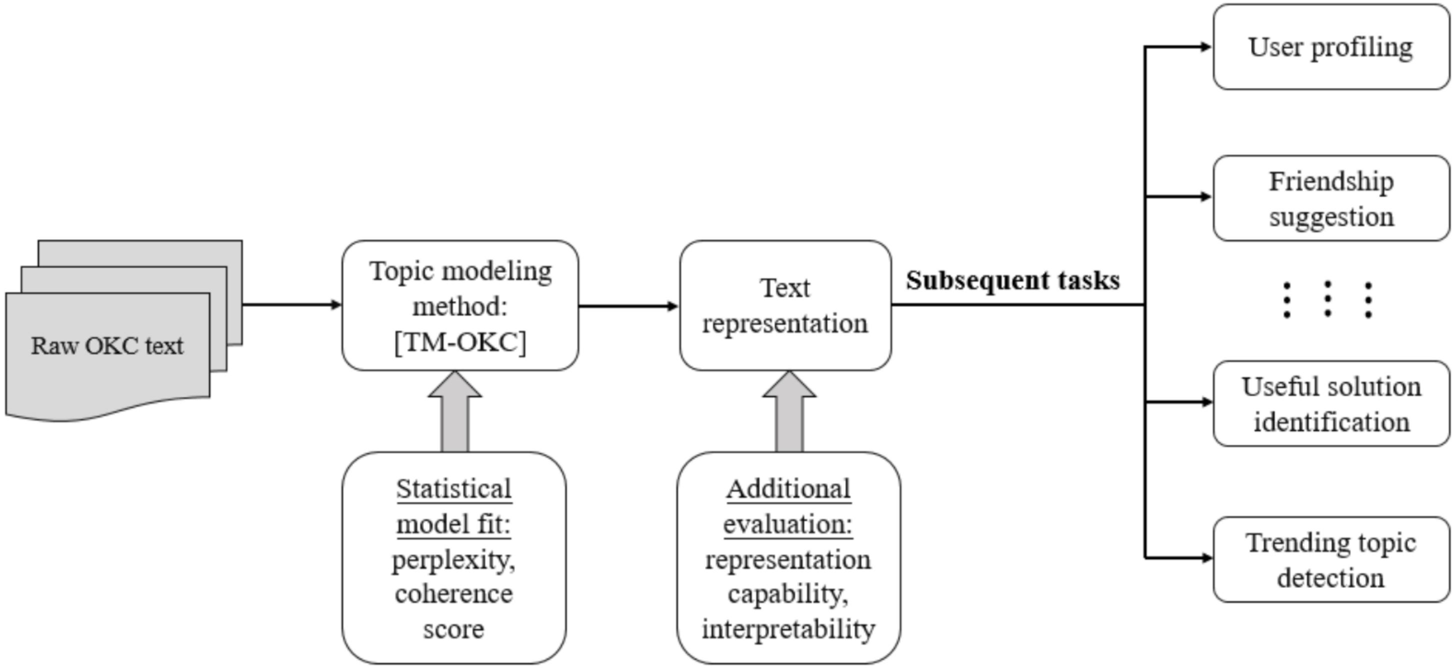
- **Topics as “independent variables”** (Yue et al. 2019, *MIS Quarterly*; Narang et al. 2022, *Journal of Marketing Research*; Gour et al. 2022, *Production and Operations Management*)
- **Topics as “dependent variables”** (Singh et al. 2014, *Information Systems Research*; Tirunillai and Tellis 2014, *Journal of Marketing Research*; Geva et al. 2019, *MIS Quarterly*)
- **Topics used to derive new variables** (Ghose et al. 2019, *Management Science*; Pu et al. 2022, *Information Systems Research*; Bachura et al. 2022, *MIS Quarterly*)

5 Model extensions



6 Summary

- Develop a novel unsupervised Bayesian topic modeling framework
  - Complex Q&A relations and threaded structures among texts with heterogeneity
- Superior performance
  - Statistical model fit, representation capability, and human interpretability
- Implications to IS researchers and practitioners in downstream empirical studies and practical tasks



Thank you very much!

Appendix: Human interpretability

- Well-known lab studies
  - *Word intrusion* and *topic intrusion* tasks (Chang et al. 2009; Bao and Datta 2014; Palese and Piccoli 2020)
  - Amazon Mechanical Turk (MTurk)

consistency

Model ⇔ Human

{elephant, tiger, horse, pig, cow} ⇐ apple
“intruder”

⇓

{elephant, tiger, horse, apple, pig, cow}

word intrusion example

Appendix: Human interpretability

- Well-known lab studies
  - *Word intrusion* and *topic intrusion* tasks (Chang et al. 2009; Bao and Datta 2014; Palese and Piccoli 2020)
  - Amazon Mechanical Turk (MTurk)

1/20

Not every model is able to learn sample-by-sample or incrementally. However, in scikit-learn, there're some models which have partial_fit method: Incremental fit on a batch of samples … You can just search for methods name in sklearn's documentation … Also, you can use Random Forest and set number of samples (or sample ratio) per tree is small to fit the memory. Or use Dask and Dask ML to fit your data in memory.

model	data	train	test	feature	value	predict	class
memory	access	address	map	block	bit	store	device
inform	method	news	govern	descript	like	medium	office
problem	number	algorithm	time	sum	log	function	frac

topic intrusion example