LDAを用いたAmazonのレビューデータのデータマイニング

B8EM1016 富田優

目次

研究テーマと研究背景

研究テーマ

- **研究テーマ**

消費者の製品やサービスに対する好みがどのように構成されているかを理解する

- **目的と観点**：

インターネット上にある膨大なテキストデータから、消費者の製品やサービスに対する好みを取り出し、マーケティングに活用できるようにする

- **短期的な目標**：

トピックモデルを用いて、カスタマー・レビューから商品・サービスの評価軸(≒好み)を取り出し、それがどう製品のおすすめ度にどう影響を与えているのか調べる手法を提案する

研究の背景

① **マーケティング**

企業の行う製品開発は重要なマーケティング戦略の一つ

新製品開発には**コンジョイント分析**

既出製品にはアンケート調査による**因子分析**

等がある

② **機械学習(自然言語処理)**

トピックモデルという文章のトピック(内容)を分布で表すモデル

LSA,PSA,LDAなど、基本的なモデルのバリエーションが複数ある

③ **マーケティング+機械学習**

カスタマーレビューをLDAで分析することは多く行われている

“Sentence-Based Text Analysis for Customer Reviews”, Joachim Büschken, Greg M. Allenby(2016)

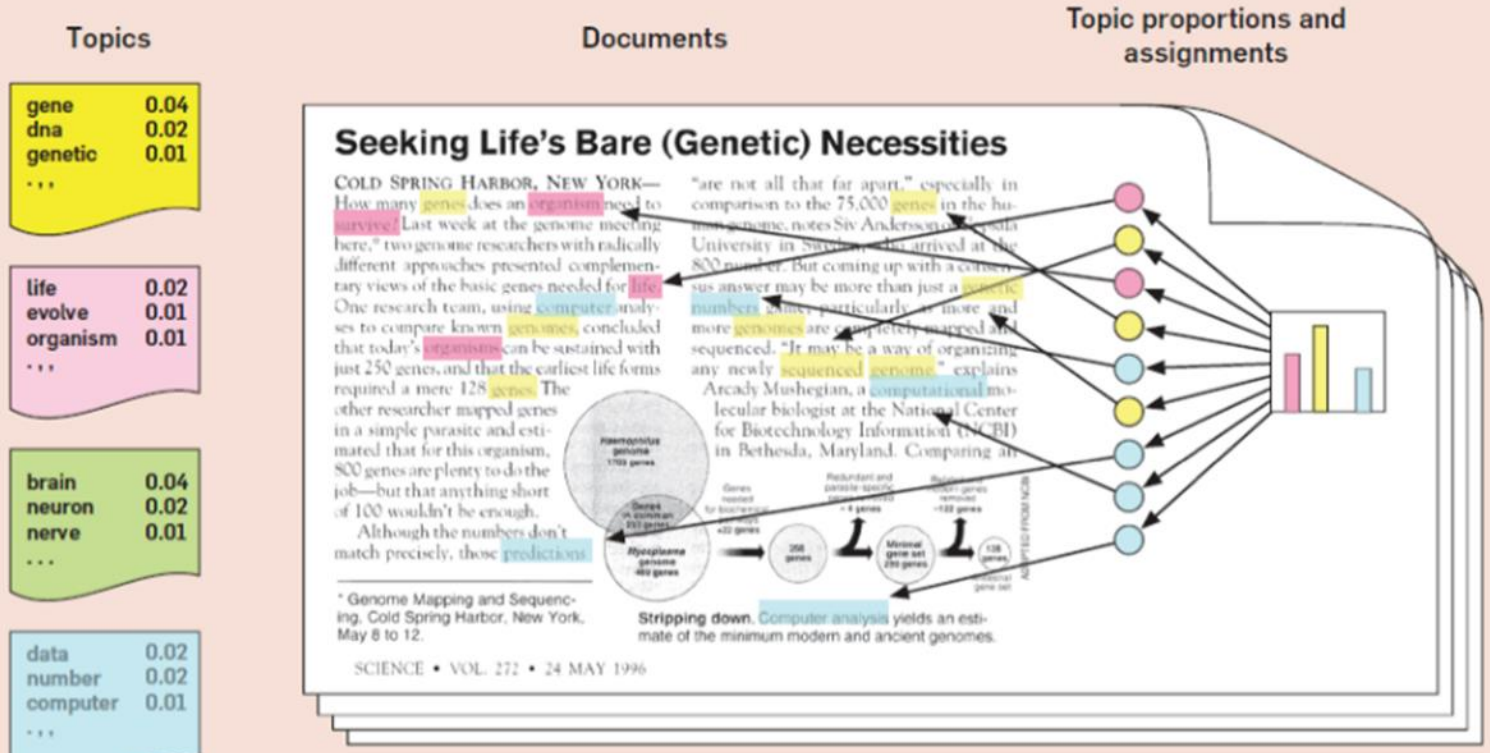
“Mining Marketing Meaning from Online Chatter“, SESHADRI TIRUNILLAI and GERARD J. TELLIS(2014)

“Topic analysis of online reviews for two competitive products using latent Dirichlet allocation”, Wenxin Wang(2018)

モデルの説明

トピックモデル
~Latent Dirichlet Allocation~

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.



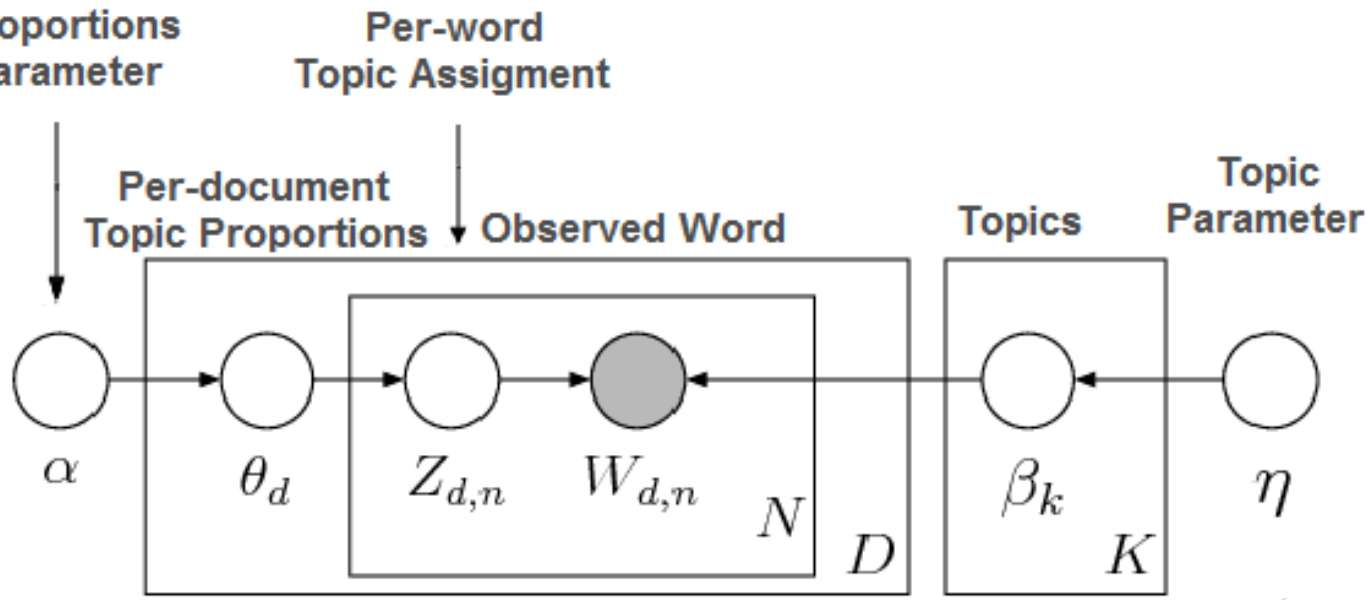
トピックモデル
~Latent Dirichlet Allocation~

○Latent Dirichlet Allocation

　トピックモデルとは、文書集合が与えられたときに、各文書がどのようなトピック(話題)について書かれているかを推定するモデル

　　①LSA②PLSA③LDA

の3つの推定方法が主にある



通常のLDAの計算結果

問題点

① 各トピックが何に言及しているかを出現頻度が上位の単語から判断しなければいけないが、どのトピックにも出現する汎用的な単語が上位に来ることで、トピックの解釈が困難になってしまう

② 名詞、形容詞などに単語を限定することで解釈しやすくするケースもあるが、生データの情報はなるべく削りたくない

③ ストップワードを設定して、解釈しやすくするケースもあるが、どこまでストップワードを設定するかが恣意的になる

拡張モデルの説明

先行研究の説明

“Sparse Additive Generative Models of Text”(Jacob et al.，2011)

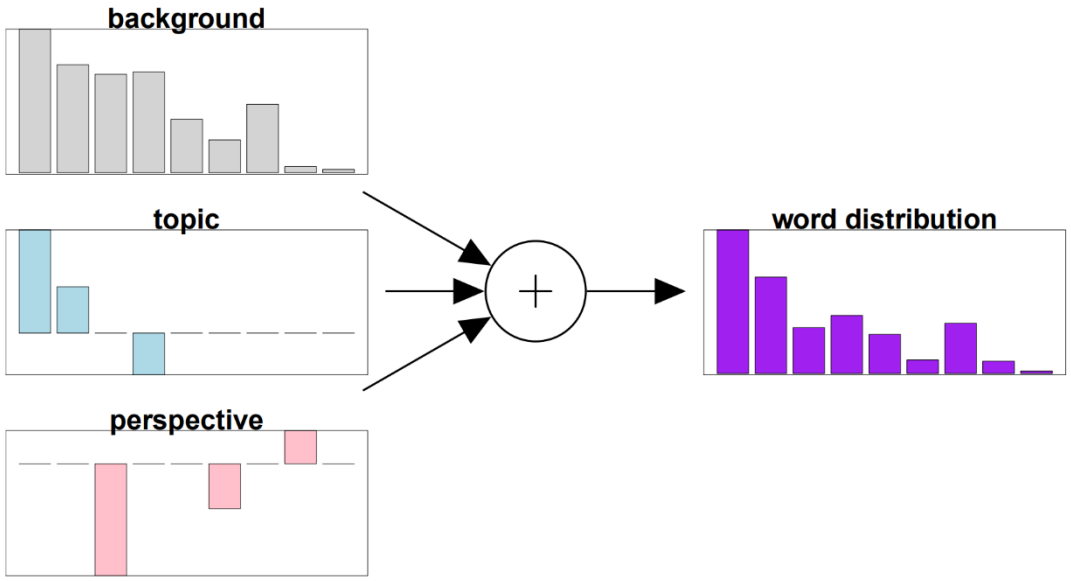
この論文のメインのアイデアは、単語分布をディリクレ分布からexpを用いたものへと変更することで、各単語の出現確率に、単語の頻度、共起性のみならず、他の要因も足すことができるようになる点

このアプローチは主に二つの利点がある:

1 スパース性を強化して、過学習を防ぐことができる

2 二つ以上の生成過程を、スイッチング変数などを導入することなしに結合することができる

先行研究の説明



- ドキュメントDにおける単語iの出現確率を以下のように変更する

- $P(w|z_d, \boldsymbol{m}, \boldsymbol{\eta}) = \frac{\exp\left(\boldsymbol{m}+\boldsymbol{\eta}_{z_d}\right)}{\sum_i \exp\left(m_i+\boldsymbol{\eta}_{z_{d,i}}\right)}$

- w:単語,z_d:単語に割り当てられたトピック
- M:バックグラウンド分布, η: それぞれの単語の構成要素

先行研究の説明

$\eta_{k,v} \sim N(0, \tau_{k,v})$:正規分布

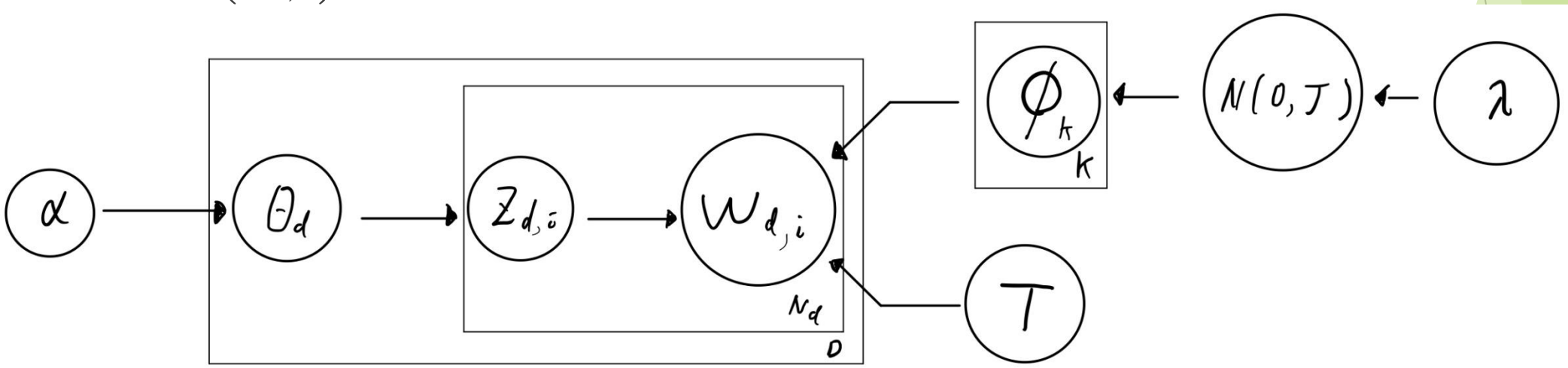
$\tau_{k,v} \sim \epsilon(\gamma)$: 指数分布

$\boldsymbol{\varphi_k} \propto \exp(\eta_{\boldsymbol{k}} + M)$:提案された分布

$\theta_d \sim Dir(\alpha)$:ディリクレ分布

$z_{d,i} \sim Multi(\theta_d)$:多項分布

$w_{d,i} \sim Multi\left(\varphi_{z_{d,i}}^{new}\right)$



推定方法

推定方法

○**推定方法**

単語分布を変更したので、ギブスサンプリングができない

→　変分ベイズ法を使用

変分ベイズ：決定論的な近似アルゴリズム

決定論的→サンプル生成ではなく、数値計算によるアルゴリズム

近似アルゴリズム→計算が容易な分布を求めることを考える

変分法：関数fを入力とする汎関数L$[q(x)]$の極値となる関数を求めるための手法

汎関数$\mathbf{L}[\boldsymbol{q}(\boldsymbol{x})] = \int \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{q})\, \boldsymbol{dx}$　の極値を与えるqは、以下のオイラー・ラジュランジュ方程式を解けばよいことが知られている

$$\frac{\boldsymbol{\partial f}(\boldsymbol{x}, \boldsymbol{q})}{\boldsymbol{\partial q}(\boldsymbol{x})} = \mathbf{0}$$

推定方法

- 求めたい事後分布

$$p(\boldsymbol{z_{1:n}}, \boldsymbol{\Phi}, \boldsymbol{\theta} | \boldsymbol{w_{1:n}}, \boldsymbol{\alpha}, \boldsymbol{\beta})$$

これを求めるのは難しいので、以下の近似された分布を求める

$$q(\boldsymbol{z_{1:n}}, \boldsymbol{\Phi}, \boldsymbol{\theta}) = \boldsymbol{q}(\boldsymbol{z})\boldsymbol{q}(\boldsymbol{\varphi})\boldsymbol{q}(\boldsymbol{\theta})$$

対数周辺尤度を考えると、イエンセンの不等式により以下のようになる(補足1,2)

$$\log \boldsymbol{P}(\boldsymbol{w_{1:n}} | \boldsymbol{\alpha}, \boldsymbol{\beta}) \geq F[q(\boldsymbol{z_{1:n}}, \boldsymbol{\Phi}, \boldsymbol{\theta})]$$

この右辺は変分下限と呼ばれる

$$F[q(\boldsymbol{z_{1:n}}, \boldsymbol{\Phi}, \boldsymbol{\theta})] = \int \sum_{\boldsymbol{z_{1:n}}} q(\boldsymbol{z_{1:n}}, \boldsymbol{\Phi}, \boldsymbol{\theta}) \log \frac{p(\boldsymbol{x_{1:n}}, \boldsymbol{z_{1:n}}, \boldsymbol{\Phi}, \boldsymbol{\theta} | \boldsymbol{\alpha}, \boldsymbol{\beta})}{q(\boldsymbol{z_{1:n}}, \boldsymbol{\Phi}, \boldsymbol{\theta})} \boldsymbol{d\varphi d}\theta$$

推定方法

- $F[q(\boldsymbol{z_{1:n}}, \boldsymbol{\Phi}, \boldsymbol{\theta})] = \int \sum_{z_{1:n}} q(\boldsymbol{z_{1:n}}, \boldsymbol{\Phi}, \boldsymbol{\theta}) \log \frac{p(\boldsymbol{w_{1:n}}, \boldsymbol{z_{1:n}}, \boldsymbol{\Phi}, \boldsymbol{\theta}|\boldsymbol{\alpha}, \boldsymbol{\beta})}{q(\boldsymbol{z_{1:n}}, \boldsymbol{\Phi}, \boldsymbol{\theta})} \boldsymbol{d\varphi d}\theta$
- $= \int \sum_{z_{1:n}} q(\boldsymbol{z_{1:n}}, \boldsymbol{\Phi}, \boldsymbol{\theta}) \log p(\boldsymbol{w_{1:n}}|\boldsymbol{z_{1:n}}, \boldsymbol{\Phi}) \boldsymbol{d\varphi d}\theta$
- $+ \int \sum_{\boldsymbol{z_{1:n}}} q(\boldsymbol{z_{1:n}}, \boldsymbol{\Phi}, \boldsymbol{\theta}) \log p(\boldsymbol{z_{1:n}}|\boldsymbol{\theta}) \boldsymbol{d\varphi d}\theta$
- $+ \int \sum_{\boldsymbol{z_{1:n}}} q(\boldsymbol{z_{1:n}}, \boldsymbol{\Phi}, \boldsymbol{\theta}) \log p(\boldsymbol{\theta}|\boldsymbol{\alpha}) \boldsymbol{d\varphi d}\theta$
- $+ \int \sum_{z_{1:n}} q(\boldsymbol{z_{1:n}}, \boldsymbol{\Phi}, \boldsymbol{\theta}) \log p(\boldsymbol{\Phi}|\boldsymbol{\beta}) \boldsymbol{d\varphi d}\theta$
- $+ \int \sum_{z_{1:n}} q(\boldsymbol{z_{1:n}}, \boldsymbol{\Phi}, \boldsymbol{\theta}) \log q(\boldsymbol{z_{1:n}}, \boldsymbol{\Phi}, \boldsymbol{\theta}) \boldsymbol{d\varphi d}\theta$

推定方法

θの分布を知りたかったら、$F[q(\boldsymbol{z}_{1:n}, \boldsymbol{\Phi}, \boldsymbol{\theta})]$のθに関係のある項を集めて変分することで

$$q(\boldsymbol{\theta}_d) \propto \prod_{k=1}^{K} \theta_{d,k}^{E[n_{d,k}]+\alpha_k-1}$$

を求められる

$$\xi_{d,k}^{\theta} = E[n_{d,k}] + \alpha_k$$

とおくと、θはディリクレ分布と分かっているので

$$q(\boldsymbol{\theta}_d) = \frac{\Gamma\left(\sum_{k=1}^{K} \xi_{d,k}^{\theta}\right)}{\prod_{k=1}^{K} \Gamma\left(\xi_{d,k}^{\theta}\right)} \prod_{k=1}^{K} \theta_{d,k}^{\xi_{d,k}^{\theta}-1}$$

と求めることができる

データセット

データセット

- Amazonから得られたカスタマーレビュー
- 2013/1~2013/12
- 今回は「Kindle fire」のレビューに限定
- 価格：約199＄
- レビュー数:2067
- 総単語数：6857
- 一般的なストップワード以外の全ての品詞

,単語を使用

- トピック数：5
- MにはTF-IDF,IDF,TFを用いた

TF－IDF

- TF-IDF とは文書集合における単語の重要性を表す指標
- **TF:**$\frac{n_j}{n}$, n_j : テキスト集合における単語jの出現頻度

  n:テキスト集合における全単語数
- **IDF:**$\log\left(\frac{N}{df_j}\right)$, N: テキストの総数,$df_j$: 単語 j を含むレビューの数
- **TF-IDF** = $\frac{n_j}{n} \times \log\left(\frac{N}{df_j}\right)$

TF-IDFの計算結果
～上位20単語～

TF

Index	Prob
use	0.0158324
guitar	0.0147568
sound	0.0112174
one	0.0106613
string	0.0103552
like	0.0093356
great	0.00889587
good	0.00824059
pedal	0.00794744
work	0.00784829
get	0.00739778
well	0.00697314
play	0.00637606
pick	0.00544271
amp	0.00542978
price	0.0052638
would	0.00514956
realli	0.00483485
need	0.00473786
look	0.00443824

IDF

Index	Prob
kingdom	9.54306
mcdonald	9.54306
mbrace	9.54306
mc404	9.54306
mc7201b	9.54306
mcapo10	9.54306
mccarti	9.54306
mccoy	9.54306
mcquad	9.54306
mayor	9.54306
mcyntir	9.54306
md100	9.54306
md421	9.54306
mdf	9.54306
mdr7506	9.54306
me501	9.54306
mb	9.54306
maverick	9.54306
meaningless	9.54306
matchless	9.54306

TF－IDF

Index	Prob
guitar	0.20311
use	0.201539
string	0.18545
sound	0.176257
pedal	0.165213
one	0.158309
like	0.14154
great	0.132445
good	0.128072
amp	0.123207
get	0.122456
work	0.122375
pick	0.115291
play	0.115049
well	0.112951
would	0.0953805
price	0.093343
realli	0.0911697
need	0.08862
cabl	0.0885133

推定結果

TF-IDFを用いた計算結果　～上位20単語～

トピック1

Index	Word
5564	snappi
4858	radio
5025	rememb
595	app
516	amaz
4421	passag
812	batman
1408	commend
15	100x
1009	bounc
4086	newberri
3805	medal
5110	revisit
804	basic
6260	turn
5725	star
2889	harri
4636	potter
2482	flexibl
5124	ride

トピック2

Index	Word	Prob
3461	kindl	0.000188343
2438	fire	0.000178419
522	amazon	0.00017426
6439	use	0.000173852
2911	hd	0.000169868
595	app	0.000169058
1788	devic	0.000168259
4894	read	0.000165985
5954	tablet	0.000165747
974	book	0.000165722
4255	one	0.000165128
3605	like	0.000164932
2794	great	0.000164094
2680	get	0.000162997
5275	screen	0.000162553
3679	love	0.000160894
3322	ipad	0.000160341
6572	want	0.000160162
6768	would	0.000159733
502	also	0.000159072

トピック3

Index	Word
3461	kindl
2438	fire
6439	use
522	amazon
2911	hd
595	app
1788	devic
4255	one
4894	read
5954	tablet
3605	like
974	book
2680	get
5275	screen
2794	great
3679	love
6768	would
3322	ipad
6743	work
6572	want

トピック4

Index	Word
4242	okay
5204	safari
618	appstor
1893	disppoint
2288	extend
2940	heaver
874	benefit
6339	unfair
6597	watchespn
5786	straightforwa…
2669	geniu
3641	log
6368	unlik
4546	pixel
5002	rel
2289	extendedli
1733	deliber
2594	frivol
6557	wad
5190	run2

トピック5

Index	Word	Prob
3461	kindl	0.000189022
2438	fire	0.00017895
522	amazon	0.00017253
6439	use	0.000171646
2911	hd	0.000170964
1788	devic	0.000168891
595	app	0.000168073
4255	one	0.000166407
4894	read	0.000165725
5954	tablet	0.000164964
974	book	0.000164891
2680	get	0.000164642
3605	like	0.000164518
2794	great	0.000162873
3679	love	0.000162833
6768	would	0.000161766
5275	screen	0.000161766
3322	ipad	0.000160858
6572	want	0.00015974
6743	work	0.000159438

IDFを用いた計算結果　～上位20単語～

トピック1

Index	Word
3461	kindl
2438	fire
6439	use
522	amazon
2911	hd
595	app
1788	devic
4255	one
4894	read
3605	like
974	book
5954	tablet
2680	get
2794	great
5275	screen
3679	love
6768	would
3322	ipad
6572	want
6743	work

トピック2

Index	Word
3461	kindl
2438	fire
6439	use
522	amazon
2911	hd
595	app
1788	devic
4255	one
974	book
4894	read
3605	like
5954	tablet
2680	get
2794	great
5275	screen
3679	love
6768	would
3322	ipad
6572	want
6743	work

トピック3

Index	Word
3461	kindl
2438	fire
6439	use
522	amazon
2911	hd
595	app
1788	devic
4255	one
974	book
3605	like
4894	read
5954	tablet
2680	get
5275	screen
2794	great
3679	love
6768	would
3322	ipad
6572	want
6743	work

トピック4

Index	Word
3461	kindl
2438	fire
6439	use
522	amazon
2911	hd
595	app
1788	devic
4255	one
974	book
4894	read
3605	like
5954	tablet
2680	get
2794	great
5275	screen
3679	love
6768	would
3322	ipad
6572	want
6743	work

トピック5

Index	Word	Prob
3461	kindl	0.000189175
2438	fire	0.000179061
6439	use	0.00017387
522	amazon	0.000172357
2911	hd	0.000170767
595	app	0.000168849
1788	devic	0.000167931
974	book	0.000166085
4894	read	0.000165958
4255	one	0.000165747
3605	like	0.000165413
5954	tablet	0.000164969
2680	get	0.000163718
2794	great	0.000162747
5275	screen	0.000162637
3679	love	0.000162389
3322	ipad	0.000161113
6768	would	0.000160811
6572	want	0.000159519
6743	work	0.0001594

TFを用いた計算結果　～上位20単語～

トピック1

Index	Word
3461	kindl
2438	fire
6439	use
2911	hd
522	amazon
974	book
595	app
5954	tablet
1788	devic
2794	great
4894	read
4255	one
2680	get
5275	screen
3605	like
3679	love
3322	ipad
3980	movi
6768	would
6595	watch

トピック2

Index	Word
3145	importantli
1765	describ
4941	recommend
2501	focus
4708	primer
5712	sr
1960	dp
751	b006gwo5wk
51	1349632797
752	b008j7eu9i
6848	zoho
5713	sr_1_1
4818	qid
4602	polari
4236	oh_details_o0…
3442	keyword
2942	heavi
1691	deal
1612	cross
6727	woman

トピック3

Index	Word
2134	enhanc
5790	strap
5962	tad
6229	trip
6342	unfinish
2171	er
6166	total
559	angl
1953	downright
4772	proprietari
6602	way
1418	compani
6266	tv
3944	mongolia
353	accesscon
3851	metropolitan
1534	coolest
455	agree4
3793	mcd
4424	passwords8

トピック4

Index	Word
2752	gorilla
5291	sd
3743	mani
181	32
927	blend
4828	qualm
3175	includ
3625	littl
5616	sophist
1660	dad
1696	debat
4911	rear
5841	subscript
1431	complaint
1430	complain
6703	wireless
2431	finger
4432	patient
4813	put
5839	submit

トピック5

Index	Word	Prob
1026	brand	0.692705
409	adob	0.307295
3461	kindl	3.3999e-13
2438	fire	3.22201e-13
6439	use	3.12712e-13
522	amazon	3.09238e-13
2911	hd	3.07036e-13
595	app	3.03542e-13
1788	devic	3.01215e-13
4255	one	2.97736e-13
974	book	2.97193e-13
4894	read	2.96898e-13
3605	like	2.96747e-13
5954	tablet	2.965e-13
2680	get	2.95005e-13
2794	great	2.92714e-13
5275	screen	2.92592e-13
3679	love	2.9111e-13
6768	would	2.89242e-13
3322	ipad	2.89023e-13

通常のLDAの計算結果

今後の課題

- ①トピックの解釈に関して、元のレビューを見ながら考察
- ②トピック分布を使ってレーティングに回帰
- ③通常のLDAと回帰の精度を比較
- ④他のカテゴリーの製品にも使用してみる
- ⑤Mに関して他にも使用できるものがないか考える(コーパスの単語頻度等)

参考文献

- [1] ” Sparse Additive Generative Models of Text”, 2011, ICML'11 Proceedings of the 28th International Conference on Machine Learning, Pages 1041-1048
- [2]「トピックモデルによる統計的潜在意味解析」奥村・佐藤
- [3]「自然言語処理概論」黒橋禎夫・柴田智秀
- [4]「トピックモデルを用いた商品の評判要因分析に関する検討」月岡・他

補足 1

$$\log p(w_{1:n} \mid \alpha, \gamma) = \log \int \sum_{z_{1:n}} p(w_{1:n}, z_{1:n}, \eta, \theta, \tau \mid \alpha, \gamma)\, d\eta\, d\theta\, d\tau$$

$$= \log \int \sum_{z_{1:n}} q(z, \eta, \theta, \tau) \frac{p(w_{1:n}, z_{1:n}, \eta, \theta, \tau \mid \alpha, \gamma)}{q(z, \eta, \theta, \tau)}\, d\eta\, d\theta\, d\tau$$

$$\geq \int \sum_{z_{1:n}} q(z, \eta, \theta, \tau) \log \frac{p(w, z, \eta, \theta, \tau \mid \alpha, \gamma)}{q(z, \eta, \theta, \tau)}\, d\eta\, d\theta\, d\tau$$

$$= F[q(z, \eta, \theta, \tau)]$$

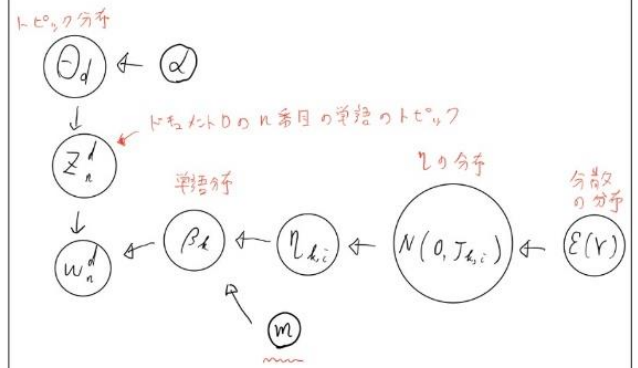
変化

イェンセンの不等式

補足２

- $\log p(\boldsymbol{w_{1:n}}|\boldsymbol{\alpha},\boldsymbol{\beta}) = F[\boldsymbol{q}(\boldsymbol{z_{1:n}},\boldsymbol{\Phi},\boldsymbol{\theta})] + KL(q(\boldsymbol{z_{1:n}},\boldsymbol{\Phi},\boldsymbol{\theta})|p(\boldsymbol{z_{1:n}},\boldsymbol{\Phi},\boldsymbol{\theta}|\boldsymbol{w_{1:n}},\boldsymbol{\alpha},\boldsymbol{\beta}))$
- ここで
- $F[q(\boldsymbol{z_{1:n}},\boldsymbol{\Phi},\boldsymbol{\theta})] = \int \sum_{\boldsymbol{z_{1:n}}} q(\boldsymbol{z_{1:n}},\boldsymbol{\Phi},\boldsymbol{\theta}) \log \frac{p(\boldsymbol{x_{1:n}},\boldsymbol{z_{1:n}},\boldsymbol{\Phi},\boldsymbol{\theta}|\alpha,\beta)}{q(\boldsymbol{z_{1:n}},\boldsymbol{\Phi},\boldsymbol{\theta})} \boldsymbol{d\varphi d}\theta$
- $KL(q(\boldsymbol{z_{1:n}},\boldsymbol{\Phi},\boldsymbol{\theta})|p(\boldsymbol{z_{1:n}},\boldsymbol{\Phi},\boldsymbol{\pi}|\boldsymbol{w_{1:n}},\alpha,\beta)) = \int q(\boldsymbol{z_{1:n}},\boldsymbol{\Phi},\boldsymbol{\theta})\log \frac{q(\boldsymbol{z_{1:n}},\boldsymbol{\Phi},\boldsymbol{\theta})}{p(\boldsymbol{z_{1:n}},\boldsymbol{\Phi},\boldsymbol{\theta}|\boldsymbol{w_{1:n}},\alpha,\beta)}$
- KLはカルバックライブラー情報量で、統計モデルqとpの近さを表し、0以上の値をとる。
- p＝qのとき、KL＝0となる。

グラフィカル・モデル



まとめ

- θ_d, z_n^d は普通のLDAと同じ
- ϕ_{kv} が変更
- $\beta_{ki} = \dfrac{\exp(\eta_{ki} + m_i)}{\sum \exp(\eta_{ki} + m_i)}$
- $\eta_{ki} \sim N(0, \tau_{ki})$
- $\tau_{ki} \sim \mathcal{E}(\gamma)$ (指数分布)
- 共役性と使えないから変分ベイズ

変分下限

$$\log\{P(w|\alpha,\gamma)\} \geqq F[q(z,\eta,\theta,\tau)]$$

$$F[q(z,\eta,\theta,\tau)] = \sum_{d=1}^{M}\langle \log P(\theta_d|\alpha)\rangle + \sum_{d=1}^{M}\sum_{n}^{N_d}\langle \log P(w_n^d|m,\eta_{z_n^d})\rangle + \sum_{d=1}^{M}\sum_{n}^{N_d}\langle \log P(z_n^d|\theta_d)\rangle +$$

$$+\sum_{k=1}^{k}\langle \log P(\eta_k|0,\tau_k)\rangle + \sum_{k=1}^{k}\langle P(\tau_k|\gamma)\rangle - \langle \log q(\tau,z,\theta)\rangle$$