Supporting Information Appendix

"On the unsupervised analysis of domain-specific Chinese texts"

- **Technical Details.** Detailed calculation of the EM Algorithm.
- **Table S1.** Detailed performance of TopWORDS on texts from *Moby Dick*.
- **Figure S1.** Receiver operating characteristic of the top-K word list discovered from *Moby Dick*.
- **Table S2.** Detailed protocol of the *word embedding pipeline*.
- **Table S3.** Compare TopWORDS with supervised approaches by processing SoS.
- **Table S4.** Words and association patterns discovered from HSD by TopWORDS and TDM.
- **Table S5.** Words and topics discovered from Sina blog posts (SBP) by TopWORDS and LDA.

Technical Details. Detailed Calculation of the EM Algorithm.

A. The EM algorithm.

Let $\boldsymbol{\theta}^{(r)}$ be the estimated parameter at the r-th iteration. The EM algorithm iterates between the two steps: the E-step computes the Q-function:

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(r)}) = \sum_{j=1}^{n} \sum_{S \in \mathcal{C}_{T_j}} P(S \mid T_j; \mathcal{D}, \boldsymbol{\theta}^{(r)}) \log P(S|\mathcal{D}, \boldsymbol{\theta}),$$

and the M-step maximizes $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(r)})$ so as to update

$$\boldsymbol{\theta}^{(r+1)} = \big(n_1^{(r)}, \cdots, n_N^{(r)}, n\big) / \big(n + \sum_i n_i^{(r)}\big),$$

where $\mathcal{C}_{T_j}$ is the set of all allowable segmentations of T_j, $n_i(T_j) = \sum_{S \in \mathcal{C}_{T_j}} n_i(S) \cdot P(S \mid T_j; \mathcal{D}, \boldsymbol{\theta}^{(r)})$, $n_i^{(r)} = \sum_{j=1}^{n} n_i(T_j)$ and $n_i(S)$ is the number of occurrences of w_i in sentence S.

B. Fast computation via dynamic programming.

The significance score of word w_i can be rewritten as

$$\psi_i = -\sum_{j=1}^{n} \log\big[1 - r_i(T_j)\big],$$

where

$$r_i(T_j) = \frac{\sum_{s \in \mathcal{C}_{T_j}} I(w_i \in S) P(S \mid \mathcal{D}, \hat{\boldsymbol{\theta}})}{P(T_j \mid \mathcal{D}, \hat{\boldsymbol{\theta}})}.$$

The computation has four major components:

$$\begin{aligned}
n_i(T) &= \frac{\sum_{S \in \mathcal{C}_T} n_i(S) P(S \mid \mathcal{D}, \boldsymbol{\theta})}{P(T \mid \mathcal{D}, \boldsymbol{\theta})} \text{ in E-step,} \\
r_i(T) &= \frac{\sum_{S \in \mathcal{C}_T} I(w_i \in S) P(S \mid \mathcal{D}, \boldsymbol{\theta})}{P(T \mid \mathcal{D}, \boldsymbol{\theta})} \text{ for getting } \psi_i, \\
\gamma_k(T) &= \frac{\sum_{S \in \mathcal{C}_T} I_k(S) P(S \mid \mathcal{D}, \boldsymbol{\theta})}{P(T \mid \mathcal{D}, \boldsymbol{\theta})} \text{ in PES, and} \\
S^*(T) &= \arg\max_{S \in \mathcal{C}_T} P(S \mid \mathcal{D}, \boldsymbol{\theta}) \text{ in MLS.}
\end{aligned}$$

It can be shown that:

$$\begin{aligned}
n_i(T) &= \sum_{t=1}^{\tau_L} \rho_t \Big[I(T_{[1:t]} = w_i) + n_i(T_{[>t]}) \Big], \\
r_i(T) &= \sum_{t=1}^{\tau_L} \rho_t \Big[I(T_{[1:t]} = w_i) + r_i(T_{[>t]}) I(T_{[1:t]} \neq w_i) \Big], \\
\gamma_k(T) &= \frac{P(T_{[1:t]} \mid \mathcal{D}, \boldsymbol{\theta}) \cdot P(T_{[>t]} \mid \mathcal{D}, \boldsymbol{\theta})}{P(T \mid \mathcal{D}, \boldsymbol{\theta})},
\end{aligned}$$

and $P(T \mid \mathcal{D}, \boldsymbol{\theta}) = \sum_{t=1}^{\tau_L} \theta_{T_{[1:t]}} \cdot P(T_{[>t]} \mid \mathcal{D}, \boldsymbol{\theta})$, where $T_{[1:t]}$ and $T_{[>t]}$ are substrings composed of the first t characters and remaining characters of unsegmented text T, respectively, and

$$\rho_t \triangleq \frac{\theta_{T_{[1:t]}} \cdot P(T_{[>t]} \mid \mathcal{D}, \boldsymbol{\theta})}{P(T \mid \mathcal{D}, \boldsymbol{\theta})}.$$

Notation $\theta_{T_{[1:t]}}$ stands for the sampling probability of word $w = T_{[1:t]}$ from the current dictionary $(\mathcal{D}, \boldsymbol{\theta})$, which equals to zero if $w \notin \mathcal{D}$. Moreover, $S^*(T)$ also has a recursive representation as follows:

$$S^*(T) = T_{[1:t]} \circ S^*(T_{[>t]}),$$

where t is selected from $\{1, \cdots, \tau_L\}$ by maximizing the likelihood of $S^*(T)$, and symbol $a \circ b$ means that there is a word boundary between a and b. These facts suggest that all above computations can be done efficiently via standard dynamic programming with a complexity of $O(Len(T) \cdot \tau_L)$.

Table S1. Detailed performance of TopWORDS on texts from *Moby Dick*

(a) Basic information about Moby Dick

Basic Letters	Letter Tokens	Word Tokens	Unique Words	Frequent Words	Rare Words
26	954,654	218,389	16,948	6,730	10,218

(b) Word discovery by TopWORDS with and without rare words as the pre-given vocabulary

	Discovered Words	True Words	True Phrases	Word fragments	Sensitivity	Specificity	Adjusted Specificity
With no rare words	11,397	6,349	3,438	1,610	$\frac{6349}{6730} = 94\%$	$\frac{6349}{11397} = 56\%$	$\frac{6349+3438}{11397} = 85.9\%$
With rare words	20,102	16,106	3,889	108	$\frac{16106}{16948} = 95\%$	$\frac{16106}{20102} = 80\%$	$\frac{16106+3889}{20102} = 99.5\%$

(c) Word segmentation by TopWORDS with and without rare words as pre-given vocabulary

	Predicted Word Boundaries	True Boundaries	Missed Boundaries	Sensitivity	Specificity	Adjusted Sensitivity
Without rare words	191,044	166,110	54,937	$\frac{166110}{221047} = 75\%$	$\frac{166110}{191044} = 87\%$	>85%
With rare words	171,741	168,503	52,544	$\frac{168503}{221047} = 76\%$	$\frac{168503}{171741} = 98\%$	>95%

Remark. More detailed results can be found in "DataFile A.zip" (download link: http://www.stat.tsinghua.edu.cn/wdm/) which contains the following files:

(1) "SDF-A-0 MobyDick_ResultSummary.xlsx": overall summary of TopWORDS results

(2) "SDF-A-1 MobyDick_DiscoveredDict.xlsx": discovered words by TopWORDS ranked by significant score

(3) "SDF-A-2 MobyDick_SegmentedText.txt": segmented texts obtained by TopWORDS

(4) "SDF-A-3 MobyDick_SegmentedText_WithRareWords.txt": segmented texts obtained by TopWORDS when rare words are used as prior knowledge

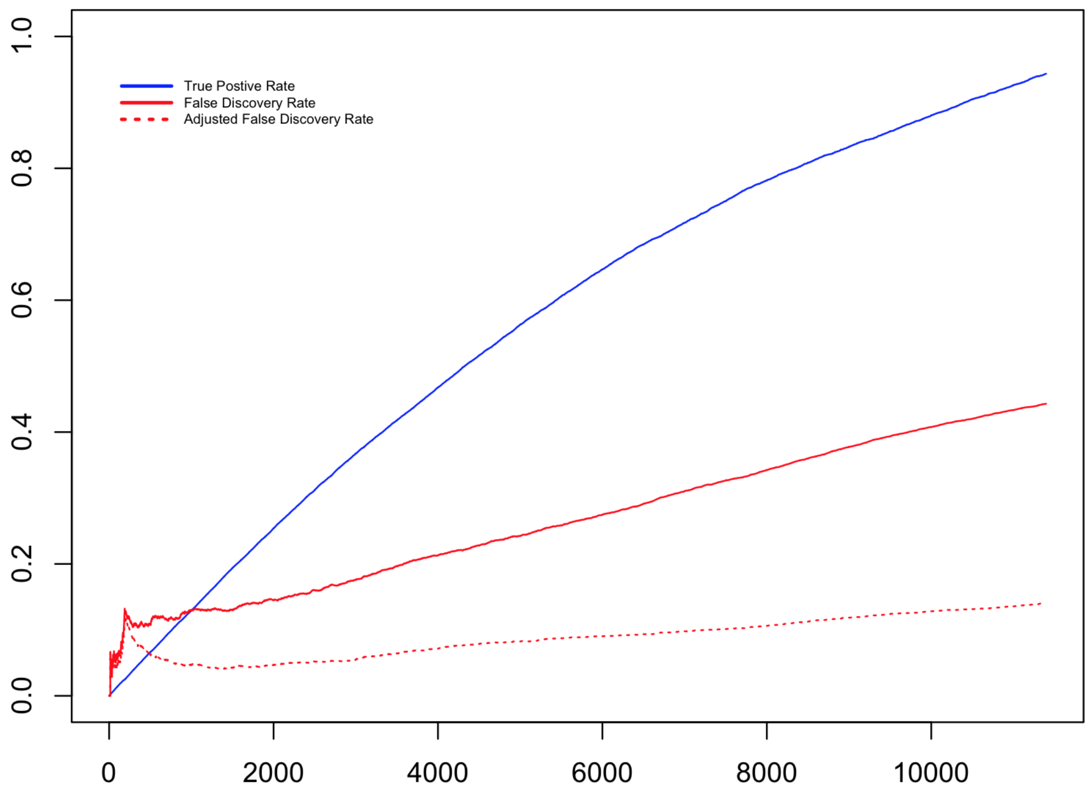

Figure S1. Operating characteristics of TopWORDS for analyzing *Moby Dick*. True positive rate (TPR) is defined as the number of correctly predicted true positives over the total number of true positives, and the false positive rate (FPR) is defined as the total number of false positives over the total number of predictions. These rates are plotted against the rank list of the words produced by TopWORDS in the analysis of English novel *Moby Dick*. Please refer to the first subsection of the Results in the main text.

Table S2. Detailed protocol of the word embedding pipeline

General Protocol

Step 0. Select words:

select a subset of words discovered by TopWORDS denoted as $\mathscr{D}$ (e.g., let $\mathscr{D}$ be the top N words)

Step 1. Get *word count matrix* M:

scan through the segmented text with a sliding windows of size $2K+1$

the word in the window center is called as the *center word*

M_{ij} counts the frequency of a word j falling into the neighborhood of a center word i, where both i and $j \in \mathscr{D}$

Step 2. Get *word relation matrix* R:

$R_{ij} = \log(sum(M) \cdot \frac{M_{ij}}{M_{i\cdot} M_{\cdot j}})$ where $M_{i\cdot} = \sum_j M_{ij}$ and $M_{\cdot j} = \sum_i M_{ij}$

reset $R_{ij} = 0$ if $R_{ij} < k$

Step 3. *Singular value decomposition* (SVD) of R:

$R = U \cdot diag\{\lambda_1, \dots, \lambda_N\} \cdot U^T$, where $\lambda_1 \geq \lambda_1 \geq \cdots \geq \lambda_N \geq 0$, and $U_{N\times N}$ is an orthogonal matrix

Step 4. Get *word embedding vectors* of words:

define $E = U \cdot diag\{\sqrt{\lambda_1}, \dots, \sqrt{\lambda_d}, 0, \cdots, 0\}$ for $d < N$

take E's first d columns as the *word embedding matrix* whose i-th row e_i is the *word embedding vector* of word i

Step 5. Get distance matrix D:

let $D_{ij} = dis(e_i, e_j)$ be either Euclidean or angel distance of e_i and e_j

Step 6. *Multidenmensional scaling* (MDS) of D:

embed d-dimensional vectors $\{e_1, \cdots, e_N\}$ into a 2-dimensional space while trying to keep distance structure D

result in a 2-dimensional coordinate (x_i, y_i) for each word i

Step 7. Draw MDS plot:

put word i to position (x_i, y_i) to illustrate the geometric structure of words in $\mathscr{D}$

Protocol Settings in Different Examples

SoS: $N = 2000,\ K = 3,\ k = 0,\ d = 200$, draw MDS plot for the top 100 discovered words

HSD: $N = 5000,\ K = 3,\ k = 0,\ d = 200$, draw MDS plot for the top technical words (highlighted with colors)

SBP: $N \approx 4500$ (union the top 1000 words of each bloggers to get a pool of ~ 4500 unique words), K=3, k=0, d=200
draw MDS plot for author-specific words (i.e., words falling into the top 1000 list of just one blogger) only

Table S3. Compare TopWORDS with supervised approaches by processing SoS

(a) Basic information about SoS.

Unique Chinese Characters	Chinese Character Tokens	Names	Frequent Names
4,505	948,901	788	371

(b) Text segmentation by different methods

	Predicted Word Boundaries	Overlaps with LTP	Overlaps with Stanford Parser	Overlaps with TopWORDS
LTP	405,817	405,817 (100%)	337,366 (83%)	248,743 (61%)
Stanford Parser	369,516	337,366 (91%)	369,516 (100%)	239,115 (64%)
TopWORDS	289,935	248,743 (86%)	239,115 (82%)	289,935 (100%)

(c) Nontrivial words discovered by different methods

	Discovered Words	Overlaps with LTP	Overlaps with Stanford Parser	Overlaps with TopWORDS
LTP	35,590	35,590 (100%)	20,905 (59%)	7,059 (20%)
Stanford Parser	40,712	20,905 (51%)	40,712 (100%)	8,860 (22%)
TopWORDS	17,205	7,059 (41%)	8,860 (52%)	17,205 (100%)

(d) Frequent nontrivial words discovery by different methods

	Discovered Frequent Words	Overlaps with LTP	Overlaps with Stanford Parser	Overlaps with TopWORDS
LTP	10,740	10,740 (100%)	8,758 (82%)	7,059 (66%)
Stanford Parser	14,817	8,758 (59%)	14,817 (100%)	8,860 (60%)
TopWORDS	17,205	7,059 (41%)	8,860 (52%)	17,205 (100%)

(e) Names and frequent names discovered by different methods.

	LTP		Stanford Parser		TopWORDS	
	Discovered	Missed	Discovered	Missed	Discovered	Missed
788 Names	445 (56%)	343 (44%)	384 (49%)	404 (51%)	345 (44%)	443 (56%)
371 Frequent Names	312 (84%)	59 (16%)	282 (76%)	89 (24%)	345 (93%)	26 (7%)

Remark. More detailed results can be found in "DataFile B.zip" (download link: http://www.stat.tsinghua.edu.cn/wdm/) which contains the following files:

(1) "SDF-B-1 SoS_DiscoveredWord.xlsx": discovered words by TopWORDS, Stanford Parser (SP) and LTP

(2) "SDF-B-2 SoS_SegmentedText.xlsx": segmented texts obtained by TopWORDS, Stanford Parser (SP) and LTP

(3) "SDF-B-3 SoS_WCM4WE.txt": Word Count Matrix M for top 2000 words (wing size $K = 3$) for the Word Embedding pipeline

Table S4. Words and association patterns discovered from HSD by TopWORDS and TDM

(a) The top 100 words discovered by TopWORDS from The HSD ranked by significant score ψ

N.o.	*Word*	*N.o.*	*Word*	*N.o.*	*Word*	*N.o.*	*Word*	*N.o.*	*Word*	*N.o.*	*Word*	*N.o.*	*Word*	*N.o.*	*Word*	*N.o.*	*Word*	*N.o.*	*Word*
1	原作(P)	11	五年(T)	21	宰相進拜加官(P)	31	不能(P)	41	真宗(N)	51	諸路(P)	61	庚戌(T)	71	癸未(T)	81	監司(O)	91	丁丑(T)
2	朝廷(P)	12	宰相(O)	22	金人(P)	32	仁宗(N)	42	神宗(N)	52	州縣(P)	62	丁亥(T)	72	乙卯(T)	82	諸州(P)	92	臣(P)
3	陛下(P)	13	天下(P)	23	明年(T)	33	左右(P)	43	戊戌(T)	53	安石(N)	63	京師(A)	73	癸巳(T)	83	庚午(T)	93	乙酉(T)
4	契丹(P)	14	四年(T)	24	侂冑(P)	34	赤黃(P)	44	乙亥(T)	54	臺諫(O)	64	辛亥(T)	74	春秋(P)	84	辛酉(T)	94	辛巳(T)
5	參知政事(O)	15	河北(A)	25	如太白(P)	35	致仕(P)	45	癸酉(T)	55	壬寅(T)	65	乙巳(T)	75	己亥(T)	85	徽宗(N)	95	巡檢(O)
6	三年(T)	16	二年(T)	26	明燭地(P)	36	皇帝(O)	46	壬戌(T)	56	內侍(O)	66	癸卯(T)	76	乙丑(T)	86	高宗(N)	96	諸軍(P)
7	未幾(P)	17	於是(P)	27	六年(T)	37	從之(P)	47	辛卯(T)	57	太后(O)	67	先是(P)	77	孝宗(O)	87	戊午(T)	97	蔡京(N)
8	太祖(N)	18	不可(P)	28	陝西(A)	38	一卷(P)	48	戊寅(T)	58	八年(T)	68	辛丑(T)	78	己酉(T)	88	丙午(T)	98	壬申(T)
9	有尾跡(P)	19	通判(O)	29	河東(A)	39	七年(T)	49	大臣(P)	59	癸丑(T)	69	戊辰(T)	79	簽書樞密院事(O)	89	戊申(T)	99	壬辰(T)
10	太宗(N)	20	執政進拜加官(P)	30	字原脫(P)	40	御史(O)	50	提舉(O)	60	執政(O)	70	癸亥(T)	80	丙戌(T)	90	至是(P)	100	至濁沒(P)

(b) The top 30 words in different word categories ranked by significant score ψ

Name			*Office title*			*Address*			*Reign title*			*Common word*		
太祖	哲宗	章惇	參知政事	簽書樞密院事	開府儀同三司	河北	江南	襄陽	元祐	淳熙	宣和	未幾	士大夫	神道碑
太宗	兀朮	似道	宰相	監司	判官	陝西	京西	秦州	乾元	大觀元年	元祐初	朝廷	弓箭手	賜襲衣
仁宗	秦檜	岳飛	通判	巡檢	鈐轄	河東	涇原	京城	元豐	乾道	永興	陛下	犯壁壘陣	白虹貫日
真宗	王安石	張俊	皇帝	知制誥	給事中	京師	荊湖	淮西	紹興元年	太平興國初	端拱初	契丹	丁母憂	不自安
神宗	張浚	蘇軾	御史	轉運使	進士	淮南	揚州	西京	靖康元年	景德元年	寧宗	明年	丁內艱	奠玉幣
安石	韓琦	富弼	提舉	皇后	中書舍人	兩浙	江西	兩淮	中興	大中祥符元年	端拱元年	赤黃	紗袍	避殿減膳
徽宗	元昊	韓世忠	臺諫	監察御史	中書	京東	太廟	交阯	熙寧	隆興元年	紹圣	天下	以城降	資治通鑑
高宗	世忠	歐陽脩	內侍	主簿	樞密	湖南	成都	荊南	鳳翔	太平興國二年	建炎元年	致仕	丁父憂	赦天下
蔡京	英宗	呂頤浩	太后	樞密院	尚書	太原	福建	鳳翔	乾安	元豐元年	宣和元年	侂冑	奉朝請	中流矢
司馬光	朱熹	范仲淹	執政	翰林學士	同知樞密院事	河南	開封府	湖北	元祐元年	熙寧五年	建炎三年	左右	墓誌銘	善騎射

(c) Top association patterns of technical terms discovered by TDM from the segmented texts of HSD produced by TopWORDS

N.o.	*Name & Name*	*Office title & Name*	*Address & Name*	*Office title & Office title*	*Address & Address*
1	黃潛善, 汪伯彥	同簽書樞密院事, 鄭清之	膠西, 李寶	登聞院, 鼓司	趙州, 平棘
2	苗傅, 劉正彥	諫官, 陳升之	泉州, 陳洪進	監司, 郡守	滄州, 清池
3	蔡京, 王黼	昭宣使, 王繼恩	晉州, 劉崇	樞密院, 三省	金州, 洵陽
4	真德秀, 魏了翁	平章軍國事, 韓侂胄	夏州, 趙保忠	判官, 簽書	河北, 河東
5	張浚, 趙鼎	都部署, 崔彥進	江南, 李景	御史, 諫官	天武, 捧日
6	曾覿, 龍大淵	經制, 余靖	河州, 景思立	通判, 知州	湖南, 江西
7	司馬光, 呂公著	參知政事, 宋庠	揚州, 李重進	同中書門下平章事, 集賢殿大學士	鳳翔, 永興
8	魏杞, 葉顒, 蔣芾	都部署, 周瑩	合州, 王堅	拾遺, 補闕	鎮戎軍, 渭州
9	王曾, 張知白	參知政事, 呂蒙正	河池, 姚仲	右僕射, 左僕射	寧化, 岢嵐
10	程頤, 程顥, 張載, 周敦頤	同平章事, 王欽若	象州, 曹利用	司徒, 司空	尉氏, 太康
11	程頤, 楊時, 游酢	平章事, 寇準	山東, 楊氏	兵部尚書, 御史大夫, 開封牧	高郵, 漣水
12	富弼, 范仲淹, 杜衍	三司使, 包拯	江南, 李煜	太師, 太傅, 太保	臨江, 興國, 南康
13	張俊, 岳飛, 劉光世	督府, 張浚	郢州, 李成	皇太后, 太皇太后, 皇太妃	淮南, 江南, 廣南
14	蔡京, 章惇, 蔡卞	翰林學士, 許將	慶州, 李復圭	上舍, 外舍, 內舍	河北, 河東, 廣南
15	張俊, 岳飛, 韓世忠	參知政事, 魯宗道	潞州,李繼勳	樞密使, 樞密副使, 宣徽使	河北, 河東, 京師

Remark. More detailed results can be found in "DataFile C.zip" (download link: http://www.stat.tsinghua.edu.cn/wdm/) which contains the following files:
(1) "SDF-C-1 HSD_DiscoveredWord.xlsx": discovered words by TopWORDS ranked by significant score
(2) "SDF-C-2 HSD_SegmentedText.txt": segmented texts obtained by TopWORDS
(3) "SDF-C-3 HSD_WCM4WE.txt": Word Count Matrix M for top 5,000 words (wing size $K = 3$) for the Word Embedding pipeline

Table S5. Words and topics discovered from Sina blog posts by TopWORDS and LDA

(a) Top 15 words and phrases discovered for eight representative bloggers ranked by relative frequency ϕ

N.o.	李承鹏 *(LC)*	徐静蕾 *(XJ)*	木子李 *(MZ)*	君之 *(JZ)*	当年明月 *(DN)*	马鼎盛 *(MD)*	叶檀 *(YT)*	潘石屹 *(PS)*
1	中国队 *Chinese Men's soccer team*	围裙 *(XJ's cat)**	木子李 *(MZ's name)*	烘焙 *baking*	长篇 *novel*	潜艇 *submarine*	创业板 *Second Board*	组图 *photos*
2	中国足协 *Chinese Soccer Association*	拍戏 *filming*	你丈夫 *your husband*	配料 *ingredients*	明朝的那些事儿 *(DN's book title)*	俄军 *Russian Army*	*a*股市场 *Main Board Market*	我们公司 *our company*
3	杜伊 *(a soccer coach)*	怎么那么 *so/such*	小三 *mistress*	面团 *dough*	朱棣 *(a historical figure)*	北韩 *North Korea*	券商 *broker*	问潘总 *ask Mr. Pan*
4	李承鹏 *(LC's name)*	围脖儿 *(XJ's cat's name)*	婚后 *married*	制作过程 *cooking process*	徐阶 *(a historical figure)*	台军 *Taiwan Army*	我国的 *Our country's*	三里屯*soho* *(PS's building name)*
5	鹏语录 *(LC's book title)*	猫咪 *cat*	出轨 *infidelity*	倒入 *pour into*	张居正 *(a historical figure)*	苏俄 *Soviet Union*	中国资本市场 *China's capital market*	张欣 *(PS's wife)*
6	米卢 *(a soccer coach)*	水蜜桃 *(XJ's friend)*	爱人 *spouse*	参考分量 *reference amount*	明军 *Ming's Army*	北约 *NATO*	再融资 *refinance*	房地产发展商 *real estate developer*
7	朱广沪 *(a soccer coach)*	精彩内容 *wonderful content*	公婆 *parents-in-law*	面糊 *baking mix*	明朝那些事儿 *(DN's book title)*	核潜艇 *nuclear submarine*	证券市场 *stock market*	这个项目 *this project*
8	中超 *Soccer Super League*	一个电影 *a movie*	已婚男 *married man*	烤箱中层 *oven's middle rack*	严嵩 *(a historical figure)*	金正日 *Kim Jong-il*	溢价 *premium price*	前门大街 *(address name)*
9	阎世铎 *(A soccer official)*	小猫 *kitty*	早泄 *premature ejaculation*	搅拌均匀 *mix well*	袁崇焕 *(a historical figure)*	苏军 *Soviet Army*	股指期货 *index futures*	您对 *your opinion*
10	你是我的敌人 *(LC's book title)*	上线 *online*	男人的 *man's*	2小勺 *2 teaspoons*	王守仁 *(a historical figure)*	军方 *the military*	红利 *bonus*	*soho*尚都 *(PS's building name)*
11	国奥 *China National Football Team*	主编的话 *Editor's Word*	婚前 *before marriage*	烤焙 *baking*	魏忠贤 *(a historical figure)*	弹道导弹 *ballistic missile*	减持 *reduction*	老潘 *(PS's nickname)*
12	左一刀右一刀 *(LC's book title)*	大昕子 *(XJ's friend)*	婚姻的 *marital*	细砂糖 *granulated sugar*	万历 *(a historical figure)*	印军 *Indian army*	股改 *stock market reform*	长城脚下的公社 *(PS's hotel name)*
13	菜刀妹 *(an Internet celebrity)*	康康 *(XJ's dog)*	爱情的 *love*	千层酥皮 *puff pastry*	朱祁镇 *(a historical figure)*	中俄 *China & Russia*	注资 *capital injection*	银河*soho* *(PS's building name)*
14	中国女足 *Chinese Women's soccer team*	博学小姐 *(XJ's friend)*	婚外情 *extramarital affair*	蛋黄 *yolk*	高拱 *(a historical figure)*	舰艇 *naval vessel*	内幕交易 *insider trading*	天水 *(PS's hometown name)*
15	切尔西 *Chelsea Football Club*	梦想照进现实 *(XJ's movie title)*	性生活 *sex life*	4小勺 *4 teaspoons*	胡宗宪 *(a historical figure)*	雷达 *radar*	高管 *senior executive*	*soho*中国基金会 *(PS's foundation name)*

**: English words within parentheses are description of the corresponding Chinese word, instead of its direct translation.*

(b) Top 100 named entities discovered from blog posts of **当年明月** *(DNMY) ranked by significant score ψ*

N.o.	*Word*	*N.o.*	*Word*	*N.o.*	*Word*	*N.o.*	*Word*	*N.o.*	*Word*	*N.o.*	*Word*	*N.o.*	*Word*	*N.o.*	*Word*	*N.o.*	*Word*	*N.o.*	*Word*
1	朱元璋(N)	11	魏忠贤(N)	21	洪承畴(N)	31	当年明月(N)	41	杨士奇(N)	51	小管(N)	61	王锡爵(N)	71	朱高炽(N)	81	冯保(N)	91	周延儒(N)
2	朱棣(N)	12	孙承宗(N)	22	高拱(N)	32	张献忠(N)	42	李自成(N)	52	王振(N)	62	张永(N)	72	赵率教(N)	82	耿炳文(N)	92	曹文诏(N)
3	皇帝(O)	13	崇祯(N)	23	申时行(N)	33	海瑞(N)	43	内阁(O)	53	高迎祥(N)	63	钱谦益(N)	73	杨廷和(N)	83	郭子兴(N)	93	叶向高(N)
4	徐阶(N)	14	戚继光(N)	24	朱宸濠(N)	34	张璁(N)	44	李成梁(N)	54	太子(O)	64	徐海(N)	74	京城(A)	84	徐渭(N)	94	御史(O)
5	努尔哈赤(N)	15	朝廷(O)	25	皇太极(N)	35	夏言(N)	45	徐有贞(N)	55	锦衣卫(O)	65	朝鲜(A)	75	刘基(N)	85	徐达(N)	95	朱重八(N)
6	朱祁镇(N)	16	嘉靖(N)	26	俞大猷(N)	36	沈惟敬(N)	46	杨一清(N)	56	倭寇(N)	66	解缙(N)	76	许显纯(N)	86	日本(A)	96	盛庸(N)
7	张居正(N)	17	胡宗宪(N)	27	于谦(N)	37	常遇春(N)	47	蓝玉(N)	57	石亨(N)	67	朱樘(N)	77	李景隆(N)	87	杨嗣昌(N)	97	丰臣秀吉(N)
8	袁崇焕(N)	18	万历(N)	28	陈友谅(N)	38	刘瑾(N)	48	卢象升(N)	58	东林党(O)	68	胡惟庸(N)	78	温体仁(N)	88	江彬(N)	98	李善长(N)
9	严嵩(N)	19	李如松(N)	29	小西行长(N)	39	汪直(N)	49	严世蕃(N)	59	祖大寿(N)	69	魏公公(N)	79	南京(A)	89	崔呈秀(N)	99	王保保(N)
10	王守仁(N)	20	朱厚照(N)	30	太监(O)	40	朱瞻基(N)	50	朱见深(N)	60	李舜臣(N)	70	巡抚(O)	80	毛文龙(N)	90	辽东(A)	100	顾宪成(N)

(c) Top words of the 10 topics discovered by LDA from the combined blog posts segmented by TopWORDS

N.o.	*Topic* 1 Economy	*Topic* 2 History	*Topic* 3 Soccer	*Topic* 4 *Background*	*Topic* 5 Finance	*Topic* 6 Sports	*Topic* 7 Military	*Topic* 8 Real Estate	*Topic* 9 Bakery	*Topic* 10 Family Life
1	中国 *China*	历史 *history*	一个 *one*	我们 *we*	市场 *market*	因为 *because*	中国 *China*	潘石屹 *(PSY's name)**	的时候 *time of*	我们 *we*
2	政府 *government*	应该可以写得好看 *(DNMY's slogan)*	中国足球 *Chinese soccer*	他们 *they*	公司 *company*	所以 *therefore*	美国 *USA*	我们 *we*	烘焙 *baking*	一个 *one*
3	市场 *market*	长篇 *novel*	足球 *soccer*	就是 *be*	股市 *stock market*	就是 *be*	日本 *Japan*	北京 *Beijing*	蛋糕 *cake*	自己 *oneself*
4	美国 *USA*	的人 *person*	我们 *we*	没有 *no*	中国 *China*	其实 *in fact*	美军 *U.S. Army*	主持人 *host*	如果 *if*	的时候 *time of*
5	我国 *our country*	明朝的那些事儿 *(DNMY's book title)*	一样 *same as*	的人 *person*	成为 *become*	一个 *one*	台湾 *Taiwan*	就是 *be*	可以 *can*	因为 *because*
6	政策 *policy*	自己 *oneself*	中国队 *Chinese team*	一个 *one*	对于 *for*	我们 *we*	解放军 *People's Liberation Army*	*soho*中国 *(PSY's company name)*	配料 *ingredients*	没有 *no*
7	如果 *if*	就是 *be*	没有 *no*	这些 *these*	投资者 *investor*	这个 *this*	俄国 *Russia*	市场 *market*	制作 *make*	就是 *be*
8	房地产 *real estate*	皇帝 *emperor*	因为 *because*	社会 *society*	企业 *enterprise*	中国足球 *Chinese soccer*	印度 *India*	开发商 *real estate developer*	黄油 *butter*	还是 *or*
9	经济 *economics*	因为 *because*	就是 *be*	自己 *oneself*	如果 *if*	作者 *author*	但是 *but*	房地产 *real estate*	以后 *after*	的人 *person*
10	目前 *currently*	他们 *they*	他们 *they*	中国 *China*	投资 *invest*	中国 *China*	俄罗斯 *Russia*	公司 *company*	所以 *therefore*	他们 *they*
11	对于 *for*	朱元璋 *(Ming's first emperor)*	就像 *just like*	一样 *same as*	资本市场 *capital market*	的人 *person*	导弹 *missile*	这些 *these*	制作过程 *cooking process*	女人 *woman*
12	已经 *already*	这个 *this*	这个 *this*	都是 *all are*	上市 *IPO*	还是 *or*	朝鲜 *North Korea*	政府 *government*	因为 *because*	男人 *man*
13	央行 *central bank*	一个 *one*	所以 *therefore*	因为 *because*	上市公司 *public company*	而且 *furthermore*	北京 *Beijing*	没有 *no*	面包 *bread*	生活 *life*
14	银行 *bank*	朱棣 *(Ming's second emperor)*	不是 *not*	一些 *some*	通过 *via*	不是 *not*	俄军 *Russian Army*	网友 *Internet acquaintance*	即可 *enough*	之后 *after*
15	成为 *become*	虽然 *although*	米卢 *(a soccer coach)*	如果 *if*	资金 *capital*	现在 *now*	北韩 *North Korea*	他们 *they*	或者 *or*	只是 *just*

Remark. More detailed results can be found in "DataFile D.zip" (download link: http://www.stat.tsinghua.edu.cn/wdm/) which contains the following files:

(1) "SDF-D-1 SBP_DiscoveredWord.xlsx": discovered words by TopWORDS ranked by significant score and relative frequency

(2) "SDF-D-2 SBP_SegmentedText.txt": segmented texts obtained by TopWORDS

(3) "SDF-D-3 SBP_WCM4WE.txt": Word Count Matrix M for $\sim$4,500 words (wing size $K = 3$) for the Word Embedding pipeline