PLOTTU – Identifying trending Topics

Based on Internet Search History

Aim of the Project

- User narrow down answers to questions often by using internet research
- Knowing the knowlegde gap can help to take action, e.g., trainings or team building
- Interesting topics can be identifyied by
  - search engine key words
  - keywords of HTML pages visited
  - keywords and phrases extracted from webpages visited

Proof of Concept Data Flow

Data Collection Scenario

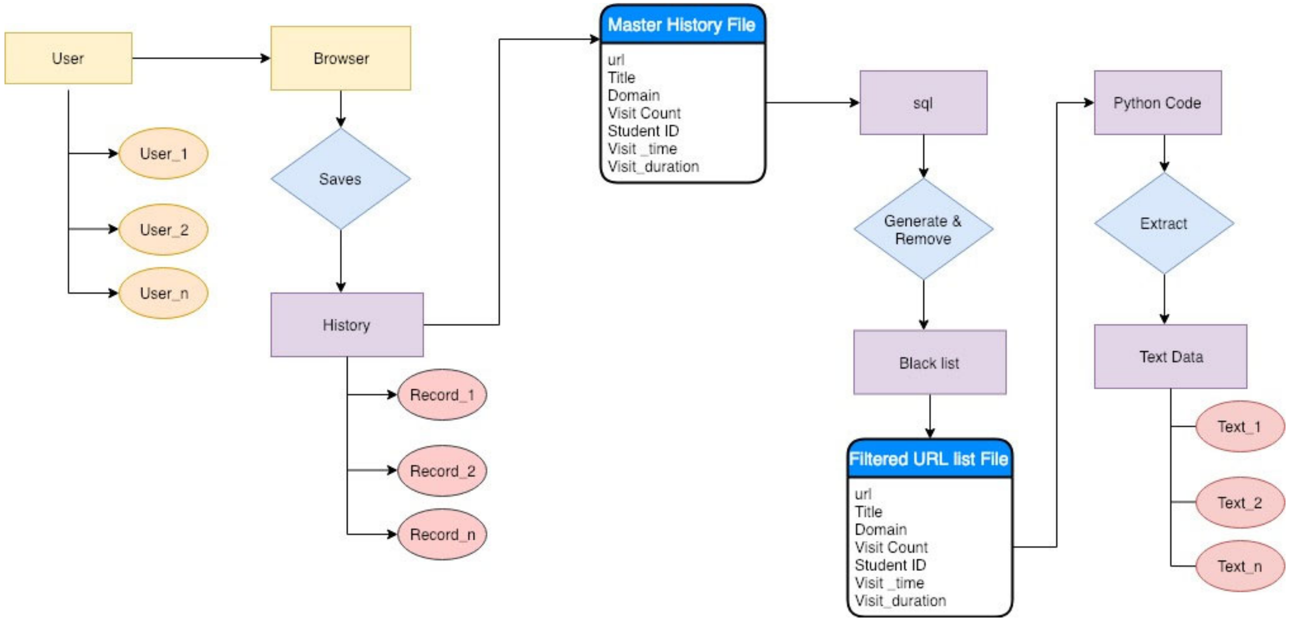
- no relevant downloads available
- students workgroup (n=34) with SPECIFIC data science question „How to assess a linear regression model?“
- Google search with Google Chorme browser
- duration approx. 15 minutes

Dataset reduction

- 104 queries extracted
- preprocessing and keyword analysis using Python

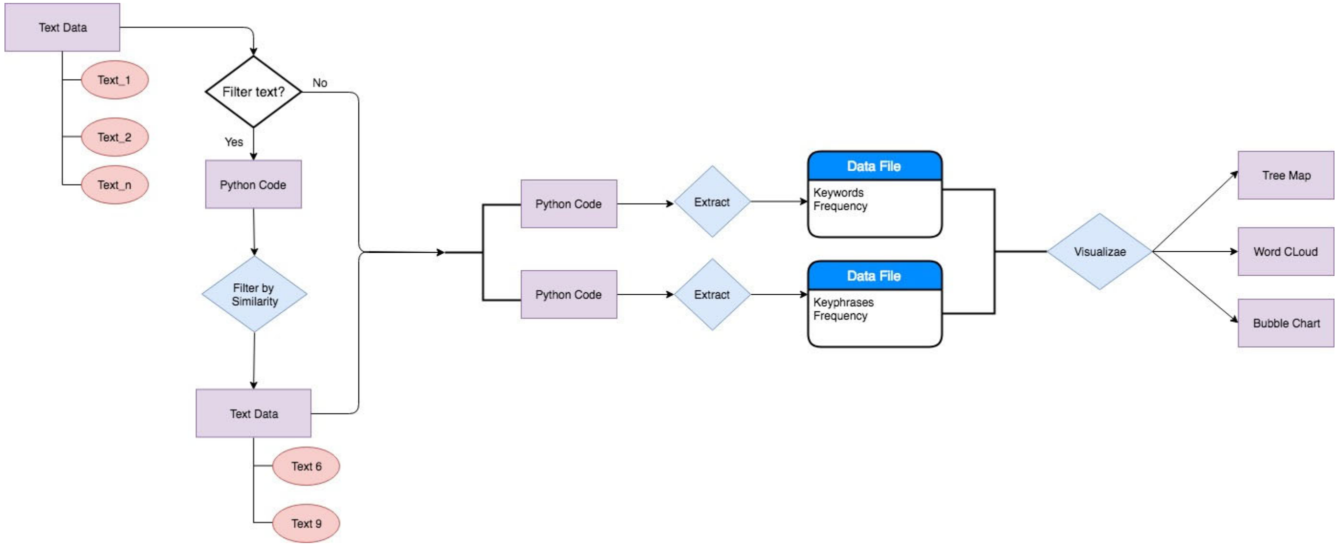
Proof of Concept Data Flow

- Data Flow Part 1



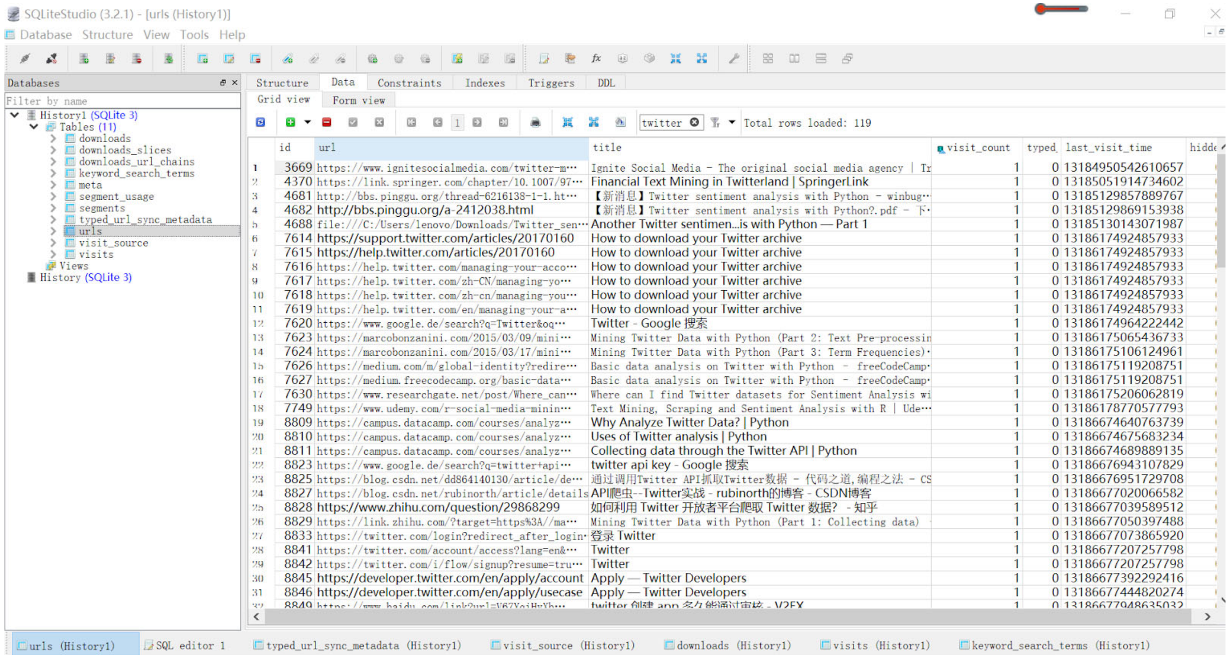
Proof of Concept Data Flow

- Data Flow Part 2



Steps to Analyse Data

Extracting Google Chrome Data



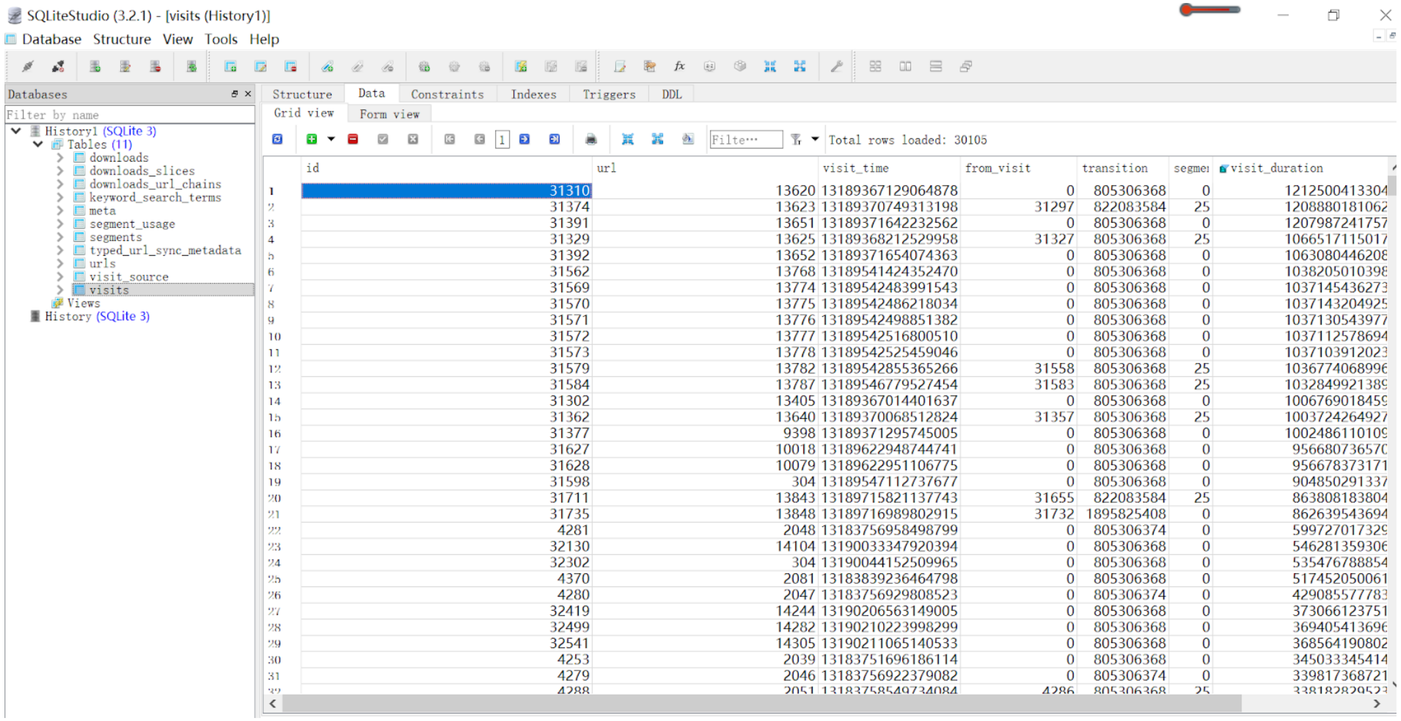
twitter

Total rows loaded: 119

	id	url	title	visit_count	typed	last_visit_time	hidde
1	3669	https://www.ignitesocialmedia.com/twitter-m···	Ignite Social Media - The original social media agency | Tr	1	0	13184950542610657	
2	4370	https://link.springer.com/chapter/10.1007/97···	Financial Text Mining in Twitterland | SpringerLink	1	0	13185051914734602	
3	4681	http://bbs.pinggu.org/thread-6216138-1-1.ht···	【新消息】Twitter sentiment analysis with Python - winbug··	1	0	13185129857889767	
4	4682	http://bbs.pinggu.org/a-2412038.html	【新消息】Twitter sentiment analysis with Python?.pdf - 下·	1	0	13185129869153938	
5	4688	file:///C:/Users/lenovo/Downloads/Twitter_sen···	Another Twitter sentimen...is with Python — Part 1	1	0	13185130143071987	
6	7614	https://support.twitter.com/articles/20170160	How to download your Twitter archive	1	0	13186174924857933	
7	7615	https://help.twitter.com/articles/20170160	How to download your Twitter archive	1	0	13186174924857933	
8	7616	https://help.twitter.com/managing-your-acco···	How to download your Twitter archive	1	0	13186174924857933	
9	7617	https://help.twitter.com/zh-CN/managing-yo···	How to download your Twitter archive	1	0	13186174924857933	
10	7618	https://help.twitter.com/zh-cn/managing-you···	How to download your Twitter archive	1	0	13186174924857933	
11	7619	https://help.twitter.com/en/managing-your-a···	How to download your Twitter archive	1	0	13186174924857933	
12	7620	https://www.google.de/search?q=Twitter&oq···	Twitter - Google 搜索	1	0	13186174964222442	
13	7623	https://marcobonzanini.com/2015/03/09/mini···	Mining Twitter Data with Python (Part 2: Text Pre-processin	1	0	13186175065436733	
14	7624	https://marcobonzanini.com/2015/03/17/mini···	Mining Twitter Data with Python (Part 3: Term Frequencies)·	1	0	13186175106124961	
15	7626	https://medium.com/m/global-identity?redire···	Basic data analysis on Twitter with Python - freeCodeCamp·	1	0	13186175119208751	
16	7627	https://medium.freecodecamp.org/basic-data···	Basic data analysis on Twitter with Python - freeCodeCamp·	1	0	13186175119208751	
17	7630	https://www.researchgate.net/post/Where_can···	Where can I find Twitter datasets for Sentiment Analysis wi	1	0	13186175206062819	
18	7749	https://www.udemy.com/r-social-media-minin···	Text Mining, Scraping and Sentiment Analysis with R | Ude···	1	0	13186178770577793	
19	8809	https://campus.datacamp.com/courses/analyz···	Why Analyze Twitter Data? | Python	1	0	13186674640763739	
20	8810	https://campus.datacamp.com/courses/analyz···	Uses of Twitter analysis | Python	1	0	13186674675683234	
21	8811	https://campus.datacamp.com/courses/analyz···	Collecting data through the Twitter API | Python	1	0	13186674689889135	
22	8823	https://www.google.de/search?q=twitter+api···	twitter api key - Google 搜索	1	0	13186676943107829	
23	8825	https://blog.csdn.net/dd864140130/article/de···	通过调用Twitter API抓取Twitter数据 - 代码之道,编程之法 - CS	1	0	13186676951729708	
24	8827	https://blog.csdn.net/rubinorth/article/details	API爬虫--Twitter实战 - rubinorth的博客 - CSDN博客	1	0	13186677020066582	
25	8828	https://www.zhihu.com/question/29868299	如何利用 Twitter 开放者平台爬取 Twitter 数据? - 知乎	1	0	13186677039589512	
26	8829	https://link.zhihu.com/?target=https%3A//ma···	Mining Twitter Data with Python (Part 1: Collecting data)	1	0	13186677050397488	
27	8833	https://twitter.com/login?redirect_after_login·	登录 Twitter	1	0	13186677073865920	
28	8841	https://twitter.com/account/access?lang=en&···	Twitter	1	0	13186677207257798	
29	8842	https://twitter.com/i/flow/signup?resume=tru···	Twitter	1	0	13186677207257798	
30	8845	https://developer.twitter.com/en/apply/account	Apply — Twitter Developers	1	0	13186677392292416	
31	8846	https://developer.twitter.com/en/apply/usecase	Apply — Twitter Developers	1	0	13186677444820274	
32	8849	https://www.baidu.com/link?url=V67YoiHyYb···	twitter 创建 app 多久能通过审核 - V2EX	1	0	13186677948635032	

Steps to Analyse Data

Extracting Google Chrome Data

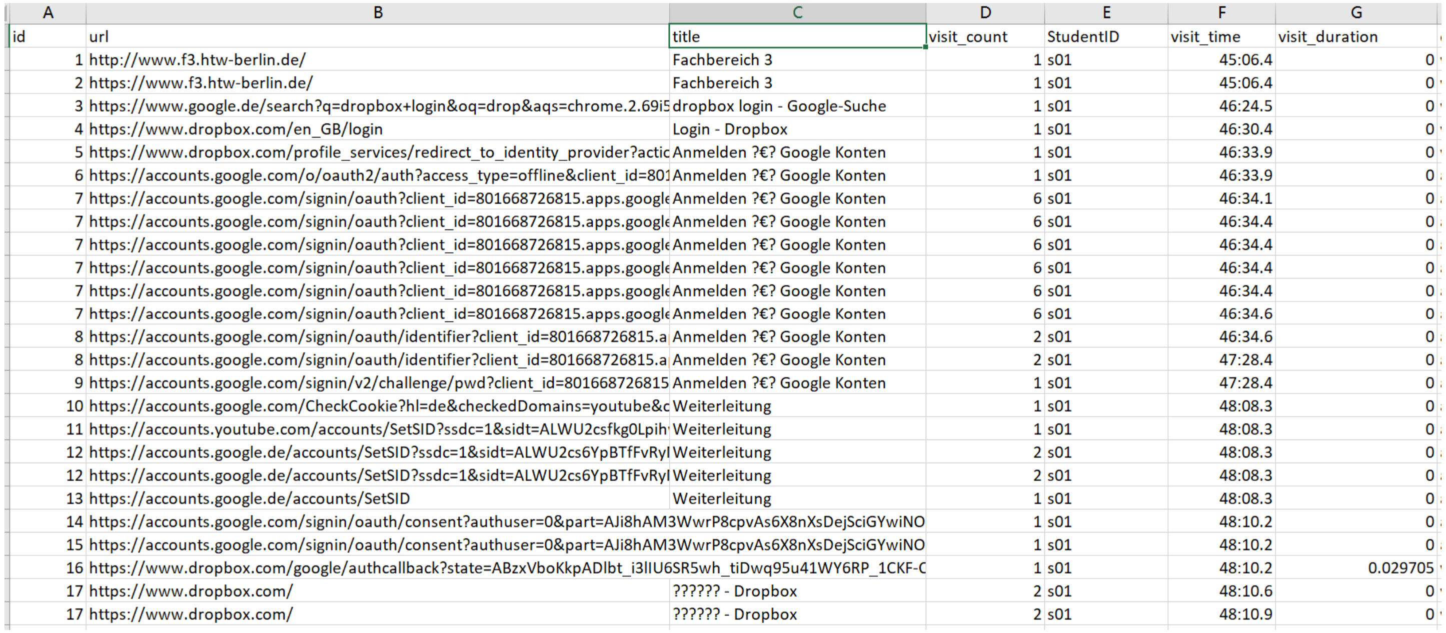


Total rows loaded: 30105

	id	url	visit_time	from_visit	transition	segme	visit_duration
1	31310	13620	13189367129064878	0	805306368	0	1212500413304
2	31374	13623	13189370749313198	31297	822083584	25	1208880181062
3	31391	13651	13189371642232562	0	805306368	0	1207987241757
4	31329	13625	13189368212529958	31327	805306368	25	1066517115017
5	31392	13652	13189371654074363	0	805306368	0	1063080446208
6	31562	13768	13189541424352470	0	805306368	0	1038205010398
7	31569	13774	13189542483991543	0	805306368	0	1037145436273
8	31570	13775	13189542486218034	0	805306368	0	1037143204925
9	31571	13776	13189542498851382	0	805306368	0	1037130543977
10	31572	13777	13189542516800510	0	805306368	0	1037112578694
11	31573	13778	13189542525459046	0	805306368	0	1037103912023
12	31579	13782	13189542855365266	31558	805306368	25	1036774068996
13	31584	13787	13189546779527454	31583	805306368	25	1032849921389
14	31302	13405	13189367014401637	0	805306368	0	1006769018459
15	31362	13640	13189370068512824	31357	805306368	25	1003724264927
16	31377	9398	13189371295745005	0	805306368	0	1002486110109
17	31627	10018	13189622948744741	0	805306368	0	956680736570
18	31628	10079	13189622951106775	0	805306368	0	956678373171
19	31598	304	13189547112737677	0	805306368	0	904850291337
20	31711	13843	13189715821137743	31655	822083584	25	863808183804
21	31735	13848	13189716989802915	31732	1895825408	0	862639543694
22	4281	2048	13183756958498799	0	805306374	0	599727017329
23	32130	14104	13190033347920394	0	805306368	0	546281359306
24	32302	304	13190044152509965	0	805306368	0	535476788854
25	4370	2081	13183839236464798	0	805306368	0	517452050061
26	4280	2047	13183756929808523	0	805306374	0	429085577783
27	32419	14244	13190206563149005	0	805306368	0	373066123751
28	32499	14282	13190210223998299	0	805306368	0	369405413696
29	32541	14305	13190211065140533	0	805306368	0	368564190802
30	4253	2039	13183751696186114	0	805306368	0	345033345414
31	4279	2046	13183756922379082	0	805306374	0	339817368721
32	4288	2051	13183758549734084	4286	805306368	25	338182829523

Steps to Analyse Data

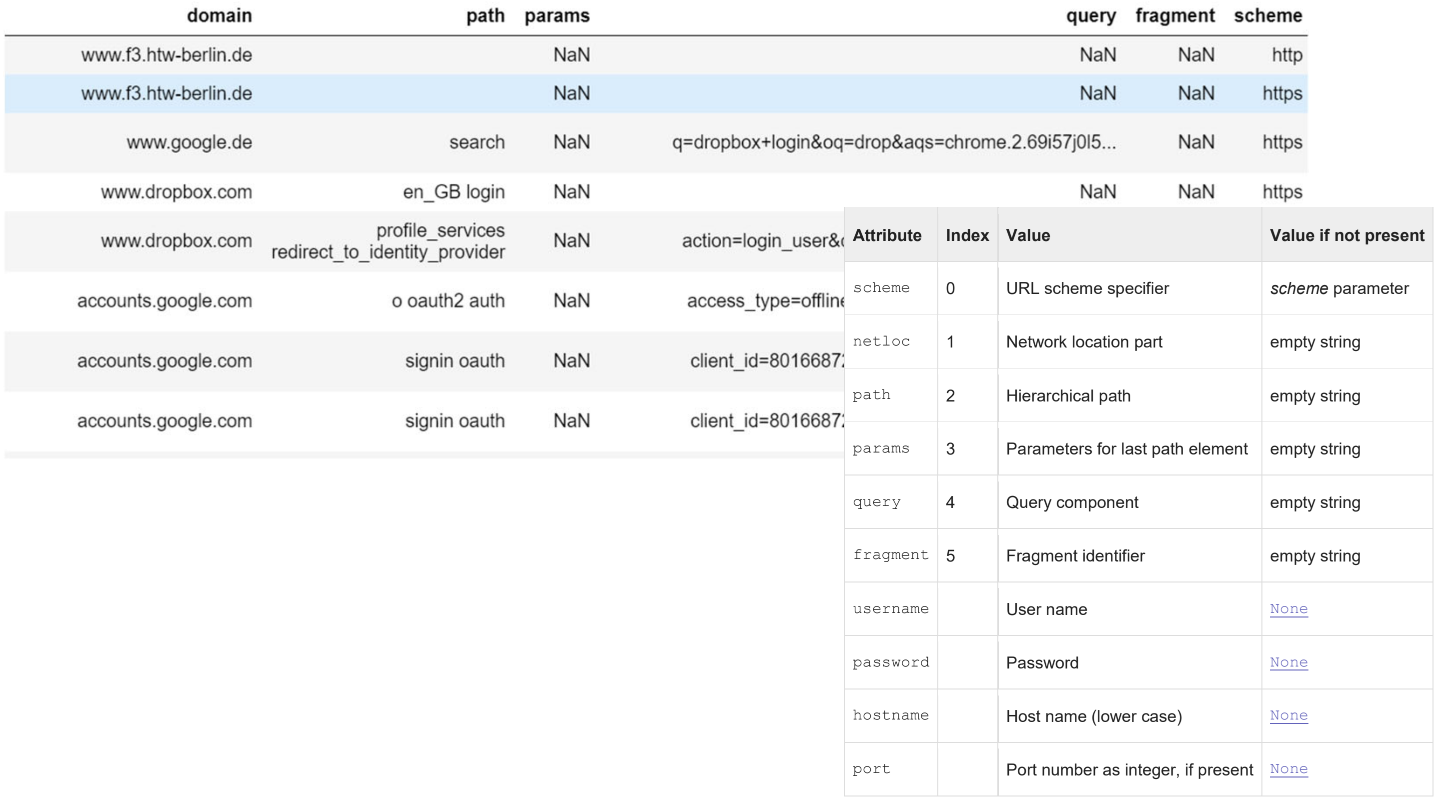
Extracting Google Chrome Data



A	B	C	D	E	F	G
id	url	title	visit_count	StudentID	visit_time	visit_duration
1	http://www.f3.htw-berlin.de/	Fachbereich 3	1	s01	45:06.4	0
2	https://www.f3.htw-berlin.de/	Fachbereich 3	1	s01	45:06.4	0
3	https://www.google.de/search?q=dropbox+login&oq=drop&aqs=chrome.2.69i5	dropbox login - Google-Suche	1	s01	46:24.5	0
4	https://www.dropbox.com/en_GB/login	Login - Dropbox	1	s01	46:30.4	0
5	https://www.dropbox.com/profile_services/redirect_to_identity_provider?actic	Anmelden ?€? Google Konten	1	s01	46:33.9	0
6	https://accounts.google.com/o/oauth2/auth?access_type=offline&client_id=801	Anmelden ?€? Google Konten	1	s01	46:33.9	0
7	https://accounts.google.com/signin/oauth?client_id=801668726815.apps.google	Anmelden ?€? Google Konten	6	s01	46:34.1	0
7	https://accounts.google.com/signin/oauth?client_id=801668726815.apps.google	Anmelden ?€? Google Konten	6	s01	46:34.4	0
7	https://accounts.google.com/signin/oauth?client_id=801668726815.apps.google	Anmelden ?€? Google Konten	6	s01	46:34.4	0
7	https://accounts.google.com/signin/oauth?client_id=801668726815.apps.google	Anmelden ?€? Google Konten	6	s01	46:34.4	0
7	https://accounts.google.com/signin/oauth?client_id=801668726815.apps.google	Anmelden ?€? Google Konten	6	s01	46:34.4	0
7	https://accounts.google.com/signin/oauth?client_id=801668726815.apps.google	Anmelden ?€? Google Konten	6	s01	46:34.6	0
8	https://accounts.google.com/signin/oauth/identifier?client_id=801668726815.a	Anmelden ?€? Google Konten	2	s01	46:34.6	0
8	https://accounts.google.com/signin/oauth/identifier?client_id=801668726815.a	Anmelden ?€? Google Konten	2	s01	47:28.4	0
9	https://accounts.google.com/signin/v2/challenge/pwd?client_id=801668726815	Anmelden ?€? Google Konten	1	s01	47:28.4	0
10	https://accounts.google.com/CheckCookie?hl=de&checkedDomains=youtube&c	Weiterleitung	1	s01	48:08.3	0
11	https://accounts.youtube.com/accounts/SetSID?ssdc=1&sidt=ALWU2csfkg0Lpih	Weiterleitung	1	s01	48:08.3	0
12	https://accounts.google.de/accounts/SetSID?ssdc=1&sidt=ALWU2cs6YpBTfFvRyl	Weiterleitung	2	s01	48:08.3	0
12	https://accounts.google.de/accounts/SetSID?ssdc=1&sidt=ALWU2cs6YpBTfFvRyl	Weiterleitung	2	s01	48:08.3	0
13	https://accounts.google.de/accounts/SetSID	Weiterleitung	1	s01	48:08.3	0
14	https://accounts.google.com/signin/oauth/consent?authuser=0&part=AJi8hAM3WwrP8cpvAs6X8nXsDejSciGYwiNO		1	s01	48:10.2	0
15	https://accounts.google.com/signin/oauth/consent?authuser=0&part=AJi8hAM3WwrP8cpvAs6X8nXsDejSciGYwiNO		1	s01	48:10.2	0
16	https://www.dropbox.com/google/authcallback?state=ABzxVboKkpADlbt_i3lIU6SR5wh_tiDwq95u41WY6RP_1CKF-C		1	s01	48:10.2	0.029705
17	https://www.dropbox.com/	?????? - Dropbox	2	s01	48:10.6	0
17	https://www.dropbox.com/	?????? - Dropbox	2	s01	48:10.9	0

Steps to Analyse Data

Extracting Google Chrome Data



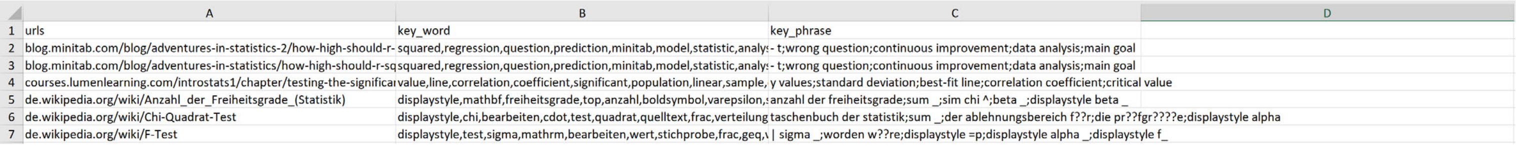
domain	path	params	query	fragment	scheme
www.f3.htw-berlin.de		NaN	NaN	NaN	http
www.f3.htw-berlin.de		NaN	NaN	NaN	https
www.google.de	search	NaN	q=dropbox+login&oq=drop&aqs=chrome.2.69i57j0l5...	NaN	https
www.dropbox.com	en_GB login	NaN	NaN	NaN	https
www.dropbox.com	profile_services redirect_to_identity_provider	NaN	action=login_user&		
accounts.google.com	o oauth2 auth	NaN	access_type=offline		
accounts.google.com	signin oauth	NaN	client_id=8016687		
accounts.google.com	signin oauth	NaN	client_id=8016687		

Attribute	Index	Value	Value if not present
scheme	0	URL scheme specifier	*scheme* parameter
netloc	1	Network location part	empty string
path	2	Hierarchical path	empty string
params	3	Parameters for last path element	empty string
query	4	Query component	empty string
fragment	5	Fragment identifier	empty string
username		User name	None
password		Password	None
hostname		Host name (lower case)	None
port		Port number as integer, if present	None

9

Extracted Relevant Content

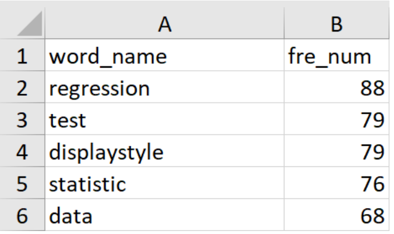
Result of Python scripts:



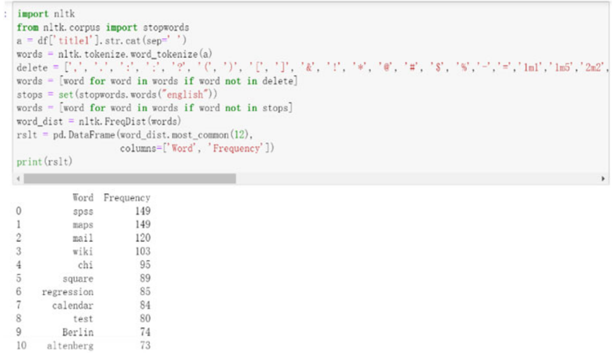
	A	B	C	D
1	urls	key_word	key_phrase	
2	blog.minitab.com/blog/adventures-in-statistics-2/how-high-should-r-	squared,regression,question,prediction,minitab,model,statistic,analy:	- t;wrong question;continuous improvement;data analysis;main goal	
3	blog.minitab.com/blog/adventures-in-statistics/how-high-should-r-sq	squared,regression,question,prediction,minitab,model,statistic,analy:	- t;wrong question;continuous improvement;data analysis;main goal	
4	courses.lumenlearning.com/introstats1/chapter/testing-the-significai	value,line,correlation,coefficient,significant,population,linear,sample,	y values;standard deviation;best-fit line;correlation coefficient;critical value	
5	de.wikipedia.org/wiki/Anzahl_der_Freiheitsgrade_(Statistik)	displaystyle,mathbf,freiheitsgrade,top,anzahl,boldsymbol,varepsilon,:	anzahl der freiheitsgrade;sum _;sim chi ^;beta _;displaystyle beta _	
6	de.wikipedia.org/wiki/Chi-Quadrat-Test	displaystyle,chi,bearbeiten,cdot,test,quadrat,quelltext,frac,verteilung	taschenbuch der statistik;sum _;der ablehnungsbereich f??r;die pr??fgr????e;displaystyle alpha	
7	de.wikipedia.org/wiki/F-Test	displaystyle,test,sigma,mathrm,bearbeiten,wert,stichprobe,frac,geq,\	| sigma _;worden w??re;displaystyle =p;displaystyle alpha _;displaystyle f_	

Extracted Data:

- frequency of keywords -> word cloud
- key word -> related key phrase



	A	B
1	word_name	fre_num
2	regression	88
3	test	79
4	displaystyle	79
5	statistic	76
6	data	68

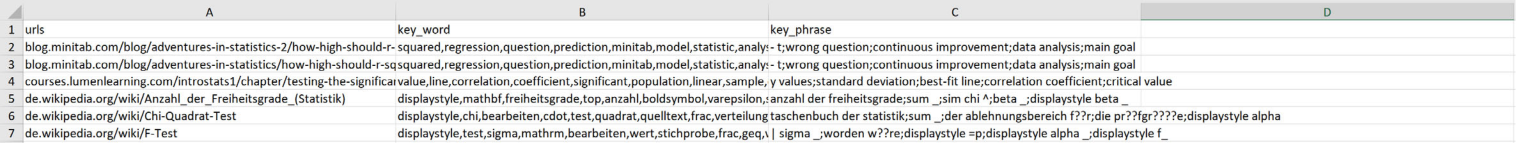


```
import nltk
from nltk.corpus import stopwords
a = df['title1'].str.cat(sep=' ')
words = nltk.tokenize.word_tokenize(a)
delete = ['.', ',', ':', ';', '?', '(', ')', '[', ']', '&', '!', '*', '@', '#', '$', '%','-','=','1m1','1m5','2m2',
words = [word for word in words if word not in delete]
stops = set(stopwords.words("english"))
words = [word for word in words if word not in stops]
word_dist = nltk.FreqDist(words)
rslt = pd.DataFrame(word_dist.most_common(12),
                    columns=['Word', 'Frequency'])
print(rslt)
```

```
          Word  Frequency
0         spss        149
1         maps        149
2         mail        120
3         wiki        103
4          chi         95
5       square         89
6   regression         85
7     calendar         84
8         test         80
9       Berlin         74
10  altenberg          73
```

Extracted Relevant Content

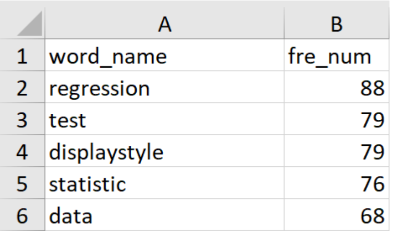
Result of Python scripts:



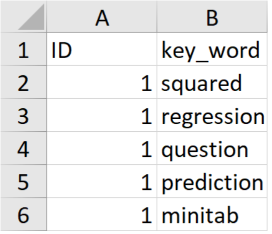
	A	B	C	D
1	urls	key_word	key_phrase	
2	blog.minitab.com/blog/adventures-in-statistics-2/how-high-should-r-	squared,regression,question,prediction,minitab,model,statistic,analy	- t;wrong question;continuous improvement;data analysis;main goal	
3	blog.minitab.com/blog/adventures-in-statistics/how-high-should-r-sq	squared,regression,question,prediction,minitab,model,statistic,analy	- t;wrong question;continuous improvement;data analysis;main goal	
4	courses.lumenlearning.com/introstats1/chapter/testing-the-significa	value,line,correlation,coefficient,significant,population,linear,sample,	y values;standard deviation;best-fit line;correlation coefficient;critical value	
5	de.wikipedia.org/wiki/Anzahl_der_Freiheitsgrade_(Statistik)	displaystyle,mathbf,freiheitsgrade,top,anzahl,boldsymbol,varepsilon,	anzahl der freiheitsgrade;sum _;sim chi ^;beta _;displaystyle beta _	
6	de.wikipedia.org/wiki/Chi-Quadrat-Test	displaystyle,chi,bearbeiten,cdot,test,quadrat,quelltext,frac,verteilung	taschenbuch der statistik;sum _;der ablehnungsbereich f??r;die pr??fgr????e;displaystyle alpha	
7	de.wikipedia.org/wiki/F-Test	displaystyle,test,sigma,mathrm,bearbeiten,wert,stichprobe,frac,geq,	| sigma _;worden w??re;displaystyle =p;displaystyle alpha _;displaystyle f_	

Extracted Data:

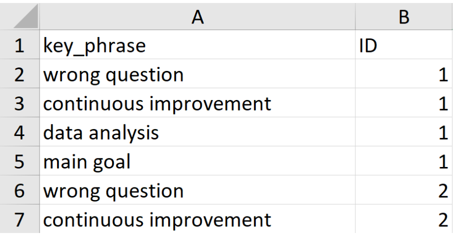
- frequency of keywords -> word cloud
- key word -> related key phrase



	A	B
1	word_name	fre_num
2	regression	88
3	test	79
4	displaystyle	79
5	statistic	76
6	data	68

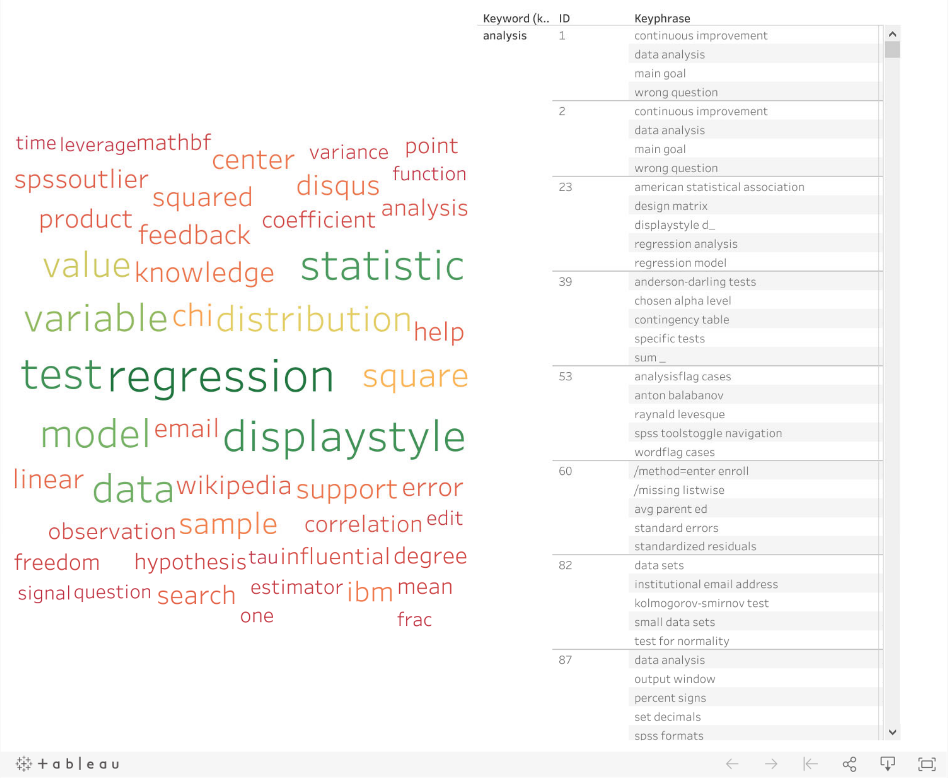


	A	B
1	ID	key_word
2	1	squared
3	1	regression
4	1	question
5	1	prediction
6	1	minitab



	A	B
1	key_phrase	ID
2	wrong question	1
3	continuous improvement	1
4	data analysis	1
5	main goal	1
6	wrong question	2
7	continuous improvement	2

Identifying Trending Topics - VISUALIZATION



Keyword (k..	ID	Keyphrase
analysis	1	continuous improvement
		data analysis
		main goal
		wrong question
	2	continuous improvement
		data analysis
		main goal
		wrong question
	23	american statistical association
		design matrix
		displaystyle d_
		regression analysis
		regression model
	39	anderson-darling tests
		chosen alpha level
		contingency table
		specific tests
		sum _
	53	analysisflag cases
		anton balabanov
		raynald levesque
		spss toolstoggle navigation
		wordflag cases
	60	/method=enter enroll
		/missing listwise
		avg parent ed
		standard errors
		standardized residuals
	82	data sets
		institutional email address
		kolmogorov-smirnov test
		small data sets
		test for normality
	87	data analysis
		output window
		percent signs
		set decimals
		spss formats

https://public.tableau.com/profile/anuj.dixit#!/vizhome/shared/J7PW25C8G

Further Improvements

Possible improvements

- implementing a black list of pages visited helps to block not relevant content, e.g., weather, news pages, ...
- company structure can provide additional context:
  searches based on department structure or job descriptions should be similar
- After tracking down certain content, similarity between pages can help to provide relevant similar pages to look at
- Twitter introduced also an algorithm to identify trending topics.
  - Get trends near a location
  - Get locations with trending topics
  - Basically the algorithms works as follows:
    - extract keywords
    - determine the number of occurrences in tweeds
    - number of occurrences -> time series
    - if slope in time series is large -> arising / trending topic