A Comparative Analysis of Topics and Trends Between Two Online Data Science Communities: Stack Exchange vs. Reddit

Habib Karbasian

May 15, 2019

Abstract

Question and answer (Q&A) websites, such as Stack Exchange [5] and Reddit [6], provide an online platform for people to talk about different topics under a general theme and share their questions and knowledge with others. Over time, these websites turn into knowledge base of different topics that can be valuable for gaining insight into the popular topics and the trends. In this report, we present a methodology to analyze the textual content of Stack Exchange, data science community [4] and Reddit discussions. We use latent Dirichlet allocation (LDA), a statistical topic modeling technique, to automatically discover the main topics present in developer discussions. We analyze these discovered topics, as well as their trends over time, to gain insights into the online data science community. Our analysis allows us to make a number of interesting observations, including: the topics of interest range widely from jobs to python, visualization topics and the topics gaining the most popularity over time are deep learning and model selection. And the last will be the comparison of these two platforms in terms of types of the topics being shared.

1 Research questions

1.1 RQ1. What are the main discussion topics in Stack Exchange and Reddit?

Recently with the growing interest in data science, people are starting questions about this field on a variety of the topics to be able to gain knowledge ranging from fundamental to technical and advanced level to be able to fulfill their needs. Q&A websites such as Stack Exchange or Reddit are designed to help with the needs of developers facing such challenges. Identifying the major discussion topics in such knowledge sharing platforms can help us pinpoint the

major areas of interest for data science practitioners or enthusiasts. This information can be of help to companies so that they can fine tune various aspects of their products. Such improvements can be made in the visualization part or providing more relevant help toward basics of optimization. This information also assists publishers of technical books, as it points out areas in which practitioners have the most interest and questions. Moreover, this information provides data science researchers with firsthand knowledge of some of the troublesome areas that are possible future research areas with a high chance of having an impact on practice.

1.2 RQ2. How does developer interest change over time?

By analyzing the rise and fall of interests in different topics, data science practitioners will be able to assess the relative popularity of the new technologies. This will also help in identifying marketing and research opportunities and trends. For example, if interest in deep learning topic is might want to direct their attention to deep learning frameworks and challenges. The trend analysis also helps in reasoning about the rise or fall of certain topics in developer discussions. rising while interest in reinforcement learning is dropping, then companies, book publishers, and researchers.

2 Datasets

2.1 Stack Exchange: Data Science

Stack Exchange is an online platform to host variety of topics where users can exchange knowledge under different types of theme. One of the subdomains is data science that is related to technical knowledge sharing for data science challenges that people have. The website features the ability for users to ask new questions and answer existing questions, as well as to vote questions and answers up or down, based on the perceived value of the post. Users of Stack Exchange can earn reputation points and badges through various activities. For example, a person is awarded 10 reputation points for receiving an up vote on any of their answers, and receives a badge for getting voted 300 times. Stack Exchange makes its data publicly available in XML format under the Creative Commons license [3]. The dataset is divided into five XML documents: badges.xml, comments.xml, posts.xml, users.xml and votes.xml. For our purposes, we use posts.xml and comments.xml, which contain the actual text content of the posts and the comments, as well as the view count, favorite count, post type, creation date, and ID of the user who created each post and comment. The dataset spans for four years , beginning of 2015 until 2019 spanning 48 months. The dataset has 57,075 posts and comments: 27,249 (47.7%) posts including questions and answers and 29,826 (52.3%) comments.

2.2 Reddit: Data Science Subreddits

In the past few years, Reddit - a community-driven platform for submitting, commenting and rating links and text posts - has grown exponentially, from a small community of users into one of the largest online communities on the Web. It brands itself as a social news website where registered users submit content in the form of links or text posts. Users, also known as Redditors, can then vote each submission up or down to rank the post and determine its position or prominence on the sites pages. These two attributes associated with a post are referred to as upvotes and downvotes. Redditors can also comment on posts, and respond back in a conversation tree of comments. Content entries, that is the posts, are organized by areas of interest or sub-communities called subreddits, such as politics, programming, science. We used the data dump provided here [1] under public licence which was collected originally from Reddits official API [7] for submissions and comments. For the purpose of this work we decided to get the data for thses 9 relevant subreddits to data science: *1-DataIsBeautiful, 2-MachineLearning, 3-DataScience, 4-LearnMachineLearning, 5-Analytics, 6-MLQuestions, 7-BigData, 8-DeepLearning* and *9-DataMining*. To have the same time line as Stack Exchange, we limit the data for the recent four years, January 2015 until December 2018. The dataset contains 3,559,702 submissions and comments: 226,134 (6.4%) submissions and 3,333,568 (93.6%) comments. The breakdown of submissions and comments for each subreddit is shown in table 2.

3 Preprocessing

Cleaning and preprocessing the textual data is the most important part of text analysis. Hence, we cleanse the textual content of the extracted posts and comments from Stack Exchange dataset and submissions and comments from subreddits in multiple steps. First, we discard any code snippets that are present in the posts (i.e., enclosed in $< code >$ HTML tags or $ for shorter version), because source code syntax (e.g., if statements and for loops) introduces noise into the analysis phase. Next, we remove all HTML tags (e.g., $< b >$ and $< a\ href = "..." >$), since these are not the focus of our analysis. Then we remove all of URLs. After that all of the texts started with any symbol (e.g. & or #) are removed. In this step, we apply tokenization process to the text using hashtags in Stack Exchange dataset as they will not be touched by the next steps and they will be considered as one single word. (e.g. *machine learning*, *data science* or *machine learning*). So there are 453 tags used for this step. After that, we apply the Porter stemming algorithm[12], which maps words to their base form (e.g.,programming, and programmer both get mapped to program). The next step is to remove common English-language stop words such as a, the and is, which do not help to create meaningful topics [13] and get more meaningful words for each text we use lemmatizatoin technique. This lemmatization process depends on correctly identifying the intended *part of speech* and meaning of a word in a sentence. So we use just adjective, adverb,

noun and verb as part of speech. The final step is remove less frequent words from the sentences. We set the minimum threshold 10 for each word and to be able to have a better and more meaningful text understanding, we remove those sentences which has less than 5 words at the end. This help the topic modeling find a more relatable and coherent topics which will be discussed in the next chapters. At the end our two dataset after the final step of preprocessing are 51,008 including 26,856 (52.3%) posts and 24,152 (47.7%) comments for Stack Exchange dataset and 2,213,205 including 137,060 (6.2%) submissions and 2,076,145 (93.8%) comments for Reddit datasset. The breakdown of the submissions and comments for subreddits is shown in table 2. In Table 1, one example of preprocessing is shown.

Table 1: Submissions and comments for each subreddit

Original	"Error trying to do a GridSearchCV() < *p* >On the following lines of code I am getting < /*p* > < *pre* >< *code* > clf = neural_network.MLPClassifier(hidden_layer_sizes=(5, 12)) parameters =['solver': ['lbfgs'],'max_iter': [500,1000,1500], 'alpha': [1e-1,1e-2,1e-3,1e-4,1e-5,1e-6,1e-7], 'random_state':[0,1,2,3,4,5,6,7,8,9]] model = GridSearchCV(clf, param_grid=paramators, n_jobs=-1) < /*code* >< /*pre* > < *p* >On the last line I am getting the following error < /*p* > < *pre* >< *code* > ValueError: Parameter values for parameter (solver) need to be a sequence(but not a string) or np.ndarray. < /*code* >< /*pre* > < *p* >now I know from reading < /*p* > < *p* ><a href=""https://datascience.stackexchange.com/questions/13410/parameters-in-gridsearchcv-in-scikit-learn/13449"">Parameters in GridSearchCV in scikit-learn< /*a* >< /*p* > < *p* >That this means everything must be in an array but all my params are in an array so what am I doing wrong. thanks< /*p* >
Preprocessed	error, tri, gridsearchcv, follow, lines code, get, lastlin, get, follow, error, know, read, paramet, gridsearchcv, scikit learn, mean, everyth, must, array, param, array, wrong, thank

Table 2: Submissions and comments for each subreddit

Ranking	Subreddit	Submissions+Comments (%)	
		Original	Preprocessed
1	DataIsBeautiful	2,975,912 (83.60)	1,781,973 (80.52)
2	MachineLearning	307,210 (8.63)	222,370 (10.05)
3	DataScience	154,149 (4.33)	116,328 (5.26)
4	LearnMachineLearning	40,642 (1.14)	31,088 (1.40)
5	Analytics	27,794 (0.78)	21,428 (0.97)
6	MLQuestions	20,946 (0.59)	17,432 (0.79)
7	BigData	18,435 (0.52)	11,843 (0.54)
8	DeepLearning	11,262 (0.32)	8,074 (0.36)
9	DataMining	3,352 (0.09)	2,669 (0.12)

4 Topic Modeling

Researchers have developed many topic modeling techniques to account for different data types, assumptions, and goals. In this paper, we use the popular topic modeling technique called latent Dirichlet allocation (LDA)[8], as it is best suited for our research goal of finding discussion topics in natural language text documents. LDA is a statistical topic modeling technique, which means that LDA represents topics as probability distributions over the words in the corpus, and it represents documents as probability distributions over the discovered topics. LDA creates topics when it finds sets of words that tend to co-occur frequently in the documents of the corpus. Often, the words in a discovered topic are semantically related, which gives meaning to the topic as a whole. For example, the words with highest probability in a topic might be planet, space, star, and orbit (because these words tend to occur together in documents), indicating that this topic is related to astronomy. Further, LDA might tell us that a particular document contains both this astronomy-related topic as well as a mathematics-related topic. Thus, it is now easy to collect all documents related to astronomy, or about mathematics, or about both, without using any training data or manually-created tags or labels.

4.1 LDA Implementation

LDA is probabilistic in nature. Given a set of documents, LDA uses machine learning algorithms to infer the topics and topic memberships for each document. In this paper, we use the implementation of the LDA model provided by MALLET version 2.0.8 [14], which is an implementation of the Gibbs sampling algorithm [10].

4.2 Number of Topics

The number of topics, denoted K, is a user-specified parameter that provides control over the granularity of the discovered topics. Larger values of K will produce finer-grained, more detailed topics while smaller values of K will produce coarser-grained, more general topics. There is no single value of K that is appropriate in all situations and all datasets ([17] and [11]). On the other hand, there is a metric called coherence score that can give a rough estimate of how good the model has been trained. That being said, we run the both datasets with different number of topics from 2 to 100. Then we choose the highest coherence score as our final number for our optimal model for each dataset. Then we build the final model for 1000 iteration. The optimal number of topics for Stack Exchange is 32 and for Reddit is 58.

4.3 Bi-Grams

LDA can operate on either the uni-grams (i.e., single words) or ngrams (i.e., sequences of n adjacent words) in the dataset. For example, given the text compile time error, there are three uni-grams (compile, time, error) and two 2-grams (compile_time, time_error). Since 2-grams (equivalently, bi-grams) have been shown to increase the quality of text analysis [16], we include the bigrams as well to the model.

4.4 Output of LDA

The result of applying LDA to our preprocessed data is (a) a set of topics, defined as distributions over the unique words in the dataset and (b) a set of topic membership vectors, one for each post, indicating the percentage of words in the post that came from each of the K topics. As mentioned before, the highest-probable words in a topic are semantically related, which together reveal the nature, or concept, of the topic. For ease of readability, we also manually provide a short label for each topic, for example SQL for the topic that has top words query, table, sql, and row. We choose labels based on the top words in the topics, as well as examining a sample of posts that contain the topics. We also note that automated methods have recently been proposed to assign labels to topics [15].

4.5 Metrics

LDA discovers K topics, $z_1, ..., z_k$. We denote the membership of a particular topic z_k in document d_i as $\theta(d_i, z_k)$. We note that $\forall i, k : 0 \leq \theta(d_i, z_k) \leq 1$ and $\forall i : \sum_k^1 \theta(d_i, z_k) = 1$. Using this notation, we compute the following metrics of interest to help us answer our research questions. We first define a threshold, σ, to indicate whether a particular topics is in a document. Usually, a document will have between 1 and 5 dominant topics, each with memberships of 0.10 or higher [8]. These constitute the main topics in the document. However, due

to the probabilistic nature of LDA, sometimes topics are assigned small but non-zero (e.g., 0.01) memberships to a document, and are not relevant to our analysis. Thus, by using the σ threshold as a membership cutoff, we keep only the main topics in each document and discard the probabilistic errors. In this paper, we set σ to 0.10, which we found to remove noisy topic memberships while still allowing only the dominant topics to be present in each document. Then for each document or text, we normalize the weight of topics to be 1.

4.5.1 Topic Share

We define the overall share of a topic z_k across all posts as

$$share(z_k) = \frac{1}{|D|} \sum_{\substack{d_i \in D \\ \theta(d_i, z_k) \geq \sigma}} \theta(d_i, z_k)$$

where D is the set of all posts in our dataset. The share metric measures the proportion of posts that contain the topic z_k. For example, if a topic has a share metric of 10%, then 10% of all texts contain this topic. The share metric allows us to measure the relative popularity of a topic across all the texts.

4.5.2 Topic Trends Over Time

We also wish to analyze the temporal trends of topics. There are two two metrics to be discussed: 1- temporal trend by topic weight, 2- temporal trend by proportional topic weight.

4.5.2.1 Temporal Trend by Topic Weight

we define the **weight impact** of a topic z_k in month m as

$$weight\ impact(z_k, m) = \sum_{d_i \in D(m)} \theta(d_i, z_k)$$

where $D(m)$ is the set of all posts in month m. The **weight impact** metric measures the texts for one give topic compared to the other topics in that particular month in terms of the weight topic. It will be shown in the result section that this helps up understand to see which topic is gaining or losing popularity over the course of four years.

4.5.2.2 Temporal Trend by Proportional Topic Weight

we define the **proportional weight impact** of a topic z_k in month m as

$$proportional\ weight\ impact(z_k, m) = \frac{1}{|D(m)|} \sum_{d_i \in D(m)} \theta(d_i, z_k)$$

where $D(m)$ is the set of all posts in month m. The **proportional weight impact** metric measures the proportional weight of texts related to that topic

compared to the other topics in that particular month. This metric will be used that how much of the information exchange is dedicated to one particular topic regardless of the number of posts and comments. It helps us understand and compare each month altogether. It also ignores the factor of online platform popularity as well.

5 Results

We now present the results of applying our research methodology on both research datasets. These experiments were run on ARGO, a research computing cluster provided by the Office of Research Computing at George Mason University, VA [2].

5.1 RQ1. What are the main discussion topics in Stack Exchange and Reddit?

We present the 32 and 58 topics discovered by our methodology from for Stack Exchange and Reddit respectively in Tables 4 and 5. The topics are arranged in descending order according to their share metric in the Figure 1. To get the detail of weight and word count of each keyword for each topic, please refer to the Appendix. As you can see there is one topic in Stack Exchange dataset as unknown and there are 10 topics in Reddit dataset labeled *???-v* as ambiguous, unknown or spam. These are the topics that are not helping with the research questions and they should be eliminated by better preprocessing techniques. To get a better feel for the topics, we show an exmaple of *Data Science Career* from each dataset in Table 3.

5.2 RQ2. How does Developer Interest Change over Time?

Data science is a rapidly changing field, driven by both innovation and consumer demand. Data science practitioners need to be aware of the recent advances in tools and technologies. Consequently, we expect developer discussions trends to be associated with the market trends. In you will see the temporal trend for each topic in terms of number of documents in per month and proportion of documents per month for both Stack Exchange and Reddit datasets. These trendlines give us an indication of the rise or fall of interest in a particular topic. We now analyze the topics trends in more detail. First, we use the Cox Stuart trend test [9] to determine if each topics impact metric is increasing or decreasing over time, to a statistically significant degree, using the standard 95% confidence level. Briefly, the Cox Stuart trend test compares the earlier data points against the later data points in a time series to determine whether its trend is increasing or decreasing, and uses the magnitudes of the differences to determine if the trend is significant. The results of the test are shown in Table 6 and 7.

Table 3: Data Science Career Examples

StackExchange	Mathematics major for data science - So I'm a recent transfer 2nd year student from Computer Science major to Mathematics major. Though I do have a bit of an issue here. I can choose between the applied mathematics, pure mathematics and statistics concentrations. Along with this major, I'm doing a minor in Data Science with courses focused on Economics and Statistics. In the future, I'm interested in doing a master's degree in Data Science and a career in data analysis for businesses. Although I know any degree can be used to get into graduate school, I still want to which would be most beneficial for education in the future as well as opportunities for internships, research, and jobs. Thank you! **Topic Weight: 0.95**
Reddit	Data Engineer aspiring to be a data scientist evaluating my career path. I currently work as a data engineer for a large company using not so interesting tech. Think SQL, SSIS, etc... I do a substantial amount of on my own python programming and data science work for the fun of it. However, it does not seem to be enough to bridge the gap to getting hired as a data scientist or even as a data engineer in a data science shop. One option i have is to work for a actuarial firm for 2-3 years as a data engineer. This firm is using new technologies that are in demand by data science/engineering teams for processing, analyzing, and visualizing data. During that time i will also be encouraged to take and pass the actuary tests. I am also heavily considering beginning an online masters program in computer science since my BS is in a non CS engineering field. I feel that i could likely complete this in 2-3 years. My 3 questions are: -Will completing 2-3 actuary tests make me a more competitive candidate for a data science position. The tests largely involve probability, statistics, and finance. In my eyes this could be viewed as almost a stats B.S. -Are online CS masters degrees with focuses on data science from reputable institutions well respected among data science hiring managers? Ideally my resume could look like the following. Would this be competitive? -5+ Years as a data engineer (with relevant tech) -2-4 Actuary tests passed (this guy knows statistics and probability) -MS in CS with a focus in data science (perhaps at the same time setup a public portfolio) **Topic Weight: 0.99**

For Stack Exchange, 30 out of 32 topics have an increasing trend and the other two have constant trend (i.e., neither increasing nor decreasing to a significant degree) in terms of number of topics per month. *Technical guidance to a problem*, *Q/A guidelines* and *Model Selection* are among the top increasing trends. But in terms of proportion of topics per month, there are 13 increasing trends, 10 constant trends and 9 decreasing trends. *Deep Learning*, *Model Selection* and *Neural Networks* are increasing among other topics

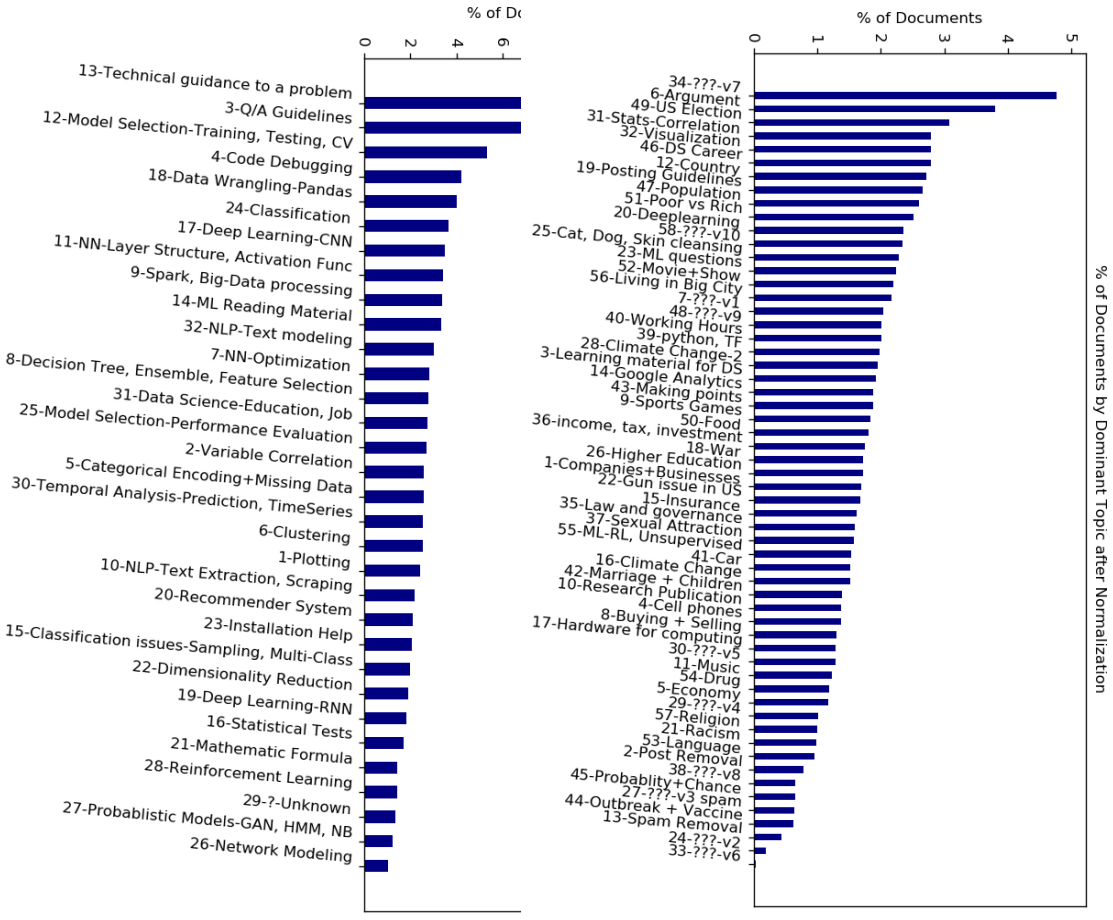
Table 4: 32 Topics of Stack Exchange

	Topics	Top LDA Keywords
1	**Plotting**	plot, data, point, valu, scale, line, normal, rang, show, exampl
2	**Variable Correlation**	model, variabl, regression, linear, linear regression, valu, correlation, fit, coeffici, predict
3	**Q/A Guidelines**	question, answer, tri, post, understand, link, comment, find, problem, add
4	**Code Debugging**	code, tri, function, error, keras, model, python, tensorflow, work, run
5	**Categorical Encoding+Missing Data**	valu, data, variabl, features, featur, categori, encoding, categor, attribut, column
6	**Clustering**	cluster, clustering, point, distance, clusters, data, k means, similar, find, similarity
7	**NN-Optimization**	weight, loss, training, gradient, error, learning rate, optim, chang, iter, valu
8	**Decision Tree, Ensemble, Feature Selection**	features, tree, featur, model, split, import, xgboost, decisiontre, random forest
9	**Spark, Big-Data processing**	data, python, r, spark, languag, packag, databas, tool, queri, write
10	**NLP-Text Extraction, Scraping**	text, extract, data, find, tag, exampl, document, search, match, dataset
11	**NN-Layer Structure, Activation Func**	layer, input, output, network, weight, neuron, neural network, activ, valu, function
12	**Model Selection-Training, Testing, CV**	model, data, training, test, train, set, accuracy, dataset, valid, cross validation
13	**Technical guidance to a problem**	problem, data, good, make, tri, work, gener, case, approach, model
14	**ML Reading Material**	paper, find, deep learning, machine learning, read, methods, book, learning, refer
15	**Classification issues-Sampling, Multi-Class**	class, data, sampl, svm, weight, tri, balanc, classifi, dataset, classifier
16	**Statistical Tests**	sampl, test, random, data, valu, number, differ, distribution, estim, error
17	**Deep Learning-CNN**	image, images, network, cnn, input, layer, filter, pixel, size, object
18	**Data Wrangling-Pandas**	column, row, data, valu, tabl, list, dataframe, index, creat, function
19	**Deep Learning-RNN**	input, lstm, sequence, output, model, rnn, network, timestep, sequenc, predict
20	**Recommender System**	user, product, item, custom, rate, recommend, base, data, purchas, time
21	**Mathematic Formula**	c, r, sum, frac, equat, function, text, theta, fracparti, alpha
22	**Dimensionality Reduction**	vector, matrix, dimens, data, pca, features, space, transform, origin, compon
23	**Installation Help**	file, run, memori, instal, gpu, orange, tri, load, comput, save
24	**Classification**	data, label, model, classification, class, problem, classifi, labels, features, classifier
25	**Model Selection-Performance Evaluation**	predict, valu, class, score, model, probability, accuracy, posit, threshold, metric
26	**Network Modeling**	graph, node, patient, age, person, citi, countri, peopl, edg, data
27	**Probablistic Models-GAN, HMM, NB**	model, distribution, probability, estim, give, observ, px, distribut, function, gaussian
28	**Reinforcement Learning**	action, state, reward, valu, agent, polici, game, move, player, reinforcement learning
29	**?-Unknown**	data, time series, detect, outlier, signal, time, frequenc, sensor, pattern, outlier
30	**Temporal Analysis-Prediction, TimeSeries**	data, time, day, predict, time series, model, event, month, year, hour
31	**Data Science-Education, Job**	data science, machine learning, data, work, project, cours, data scientist, compani, learn
32	**NLP-Text modeling**	word, document, word, sentenc, text, vector, topic, word2vec, model, term

On the other hand, for Reddit dataset, we find that 34 topics have an increasing trend, five have a decreasing trends and 19 constant trends in terms of number of topics per month. *Data Science Career*, *Machine Learning questions* and *Visualiztion* are among the most increasing trends. But in terms of proportion of topics per month, there are 22 increasing trends, 18 decreasing trends and 18 constant trends. Again *Data Science Career*, *Learning Material for Data Science* and *Machine Learning questions* are among the top increasing trends.

Table 5: 58 Topics of Reddit

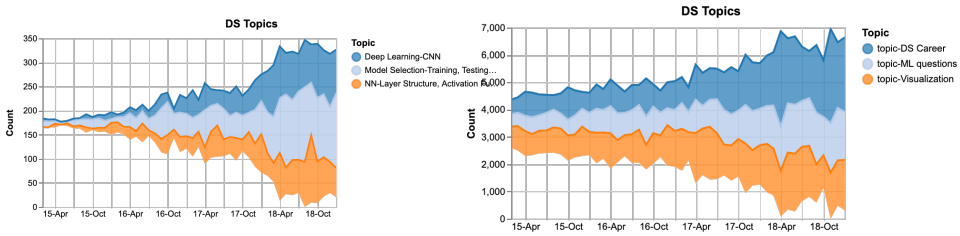
	Topics	Top LDA Keywords
1	**Companies+Businesses**	job, work, compani, pay, peopl, make, busi, money, worker, employe
2	**Post Removal**	post, reddit, subreddit, comment, postremov, feelfre, submissionremov, r
3	**Learning material for DS**	cours, learn, machine learning, good, ml, start, learning, math, understand, book
4	**Cell phones**	phone, appl, app, iphon, batteri, buy, updat, android, devic, make
5	**Economy**	chang, increas, effect, time, popul, larg, point, thing, econom, peopl
6	**Argument**	peopl, make, point, fact, gt, claim, argument, actual, believ, thing
7	**???-v1**	peopl, work, thing, good, make, lot, realli, time, tri, hard
8	**Buying + Selling**	buy, price, sell, product, valu, make, money, bitcoin, compani, market
9	**Sports Games**	game, play, team, player, game, time, sports, good, win, make
10	**Research Publication**	paper, research, work, review, publish, interest, read, author, research, result
11	**Music**	song, music, listen, play, artist, album, popular, hear, good, band
12	**Country**	countri, world, europ, popul, china, american, peopl, immigr, america, india
13	**Spam Removal**	data, concern, pleasemessag, performedautomat, questionsconcern, messagecompos
14	**Google Analytics**	site, user, data, page, google, websit, track, link, click, set
15	**Insurance**	pay, cost, money, healthcar, insur, spend, peopl, govern, tax, taxi
16	**Climate Change**	year, time, day, month, data, chang, start, averag, graph, increas
17	**Hardware for computing**	run, data, gpu, comput, internet, speed, time, server, fast, network
18	**War**	war, militari, countri, russia, fight, world, american, power, china, attack
19	**Posting Guidelines**	post, read, comment, articl, link, question, edit, find, answer, make
20	**Deeplearning**	network, model, input, layer, training, output, weight, train, gener, paper
21	**Racism**	peopl, white, black, race, racist, group, cultur, racism, american, minor
22	**Gun issue in US**	gun, peopl, kill, shoot, crime, polic, murder, death, suicid, make
23	**ML questions**	data, model, predict, problem, set, tri, good, features, dataset, test
24	**???-v2**	buttbutt, anotherplac, botbleep, linkedthread, bloopsomeon, bot, followlink
25	**Cat, Dog, Skin cleansing**	dog, wear, make, cat, guy, anim, man, time, back, put
26	**Higher Education**	school, student, colleg, year, degre, class, univers, job, work, program
27	**???-v3 spam**	originalsourc, helpfind, tri, link c, messagene, link visualization
28	**Climate Change-2**	water, energy, earth, temperatur, year, power, heat, planet, build, caus
29	**???-v4**	messag, peopl, friend, date, send, person, time, facebook, meet, talk
30	**???-v5**	data, sourc, visualization, make, visual, creat, excel, chart, map, inform
31	**Stats-Correlation**	data, number, averag, high, rate, show, popul, compar, measur, peopl
32	**Visualization**	data, color, graph, make, line, map, show, chart, point, axi
33	**???-v6**	fuck, nothingwrong, hitl, cunt, fuckfuck, fickiti, fuckiti, duck, figgidi, fuckingfuck
34	**???-v7**	peopl, fuck, shit, make, guy, thing, realli, reddit, bad, call
35	**Law and governance**	govern, state, law, peopl, power, legal, gt, system, make, rule
36	**income, tax, investment**	money, pay, make, debt, year, incom, invest, peopl, save, buy
37	**Sexual Attraction**	woman, man, male, femal, sex, gender, girl, peopl, guy, attract
38	**???-v8**	thankorigin, informationpleas, importantinform, postthread, discussionthread
39	**python, TF,**	code, python, data, work, r, run, write, file, languag, c
40	**Working Hours**	time, work, day, hour, week, spend, year, month, minut, sleep
41	**Car**	car, drive, road, driver, time, vehicl, speed, peopl, truck, traffic
42	**Marriage + Children**	kid, child, parent, famili, peopl, age, year, live, marri, young
43	**Making points**	make, point, thing, differ, peopl, realli, someth, good, understand, actual
44	**Outbreak + Vaccine**	peopl, problem, vaccin, risk, caus, issu, die, happen, thing, kill
45	**Probablity+Chance**	number, random, chanc, win, time, bet, odd, pick, probability, give
46	**DS Career**	work, job, compani, data science, data, experi, data scientist, good, project, interview
47	**Population**	state, citi, live, area, popul, map, counti, peopl, california, place
48	**???-v9**	time, start, thing, make, tri, back, year, realli, someth, end
49	**US Election**	vote, trump, elect, state, peopl, parti, presid, republican, candid, democrat
50	**Food**	eat, food, drink, good, make, beer, fat, calori, buy, tast
51	**Poor vs Rich**	peopl, make, live, life, poor, work, thing, world, good, rich
52	**Movie+Show**	movi, show, watch, episod, good, season, film, charact, netflix, make
53	**Language**	word, word, languag, english, sentenc, text, write, speak, charact, type
54	**Drug**	drug, smoke, peopl, doctor, alcohol, medic, cancer, high, addict, caus
55	**ML-RL, Unsupervised**	human, ai, problem, model, system, comput, learn, work, gener, base
56	**Living in Big City**	live, citi, hous, place, peopl, area, home, move, rent, build
57	**Religion**	peopl, religion, muslim, christian, believ, religi, god, church, islam, atheist
58	**???-v10**	realli, interest, good, great, love, make, cool, data, someth, work



(a) Stack Exchange dataset

(b) Reddit Dataset

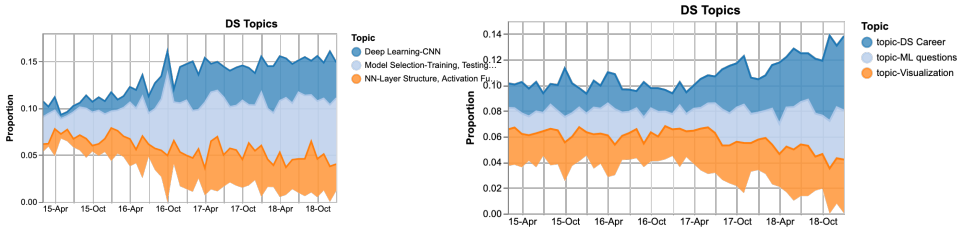
Figure 1: Topics in the descending order by the share percentage



(a) Stack Exchange

(b) Reddit

Figure 2: 3 most increasing topics in terms of number of topics per month



(a) Stack Exchange

(b) Reddit

Figure 3: 3 most increasing topics in terms of proportion of topics per month

Table 6: Result of trend for Stack Exchange in terms of number of topics *(left)* and proportion of topics *(right)* per month

Id	Topics	%difference
12	Technical guidance to a problem	⇑ 19.73
2	Q/A Guidelines	⇑ 17.28
11	Model Selection-Training, Testing, CV	⇑ 14.97
16	Deep Learning-CNN	⇑ 13.39
3	Code Debugging	⇑ 11.61
10	NN-Layer Structure, Activation Func	⇑ 10.33
6	NN-Optimization	⇑ 8.98
23	Classification	⇑ 8.68
17	Data Wrangling-Pandas	⇑ 8.18
18	Deep Learning-RNN	⇑ 7.16
24	Model Selection-Performance Evaluation	⇑ 7.09
4	Categorical Encoding+Missing Data	⇑ 6.80
27	Reinforcement Learning	⇑ 5.36
7	Decision Tree, Ensemble, Feature Selection	⇑ 5.22
1	Variable Correlation	⇑ 5.12
29	Temporal Analysis-Prediction, TimeSeries	⇑ 5.12
31	NLP-Text modeling	⇑ 4.89
14	Classification issues-Sampling, Multi-Class	⇑ 4.75
22	Installation Help	⇑ 4.63
0	Plotting	⇑ 4.55
13	ML Reading Material	⇑ 4.17
20	Mathematic Formula	⇑ 3.77
28	?-Unknown	⇑ 3.23
5	Clustering	⇑ 3.20
21	Dimensionality Reduction	⇑ 3.11
9	NLP-Text Extraction, Scraping	⇑ 3.02
15	Statistical Tests	⇑ 2.27
26	Probablistic Models-GAN, HMM, NB	⇑ 1.73
19	Recommender System	⇑ 1.63
25	Network Modeling	⇑ 1.32
30	Data Science-Education, Job	— 0.37
8	Spark, Big-Data processing	— −0.6

Id	Topics	%difference
16	Deep Learning-CNN	⇑ 0.024
11	Model Selection-Training, Testing, CV	⇑ 0.021
10	NN-Layer Structure, Activation Func	⇑ 0.016
3	Code Debugging	⇑ 0.016
18	Deep Learning-RNN	⇑ 0.014
6	NN-Optimization	⇑ 0.013
23	Classification	⇑ 0.008
27	Reinforcement Learning	⇑ 0.007
24	Model Selection-Performance Evaluation	⇑ 0.006
4	Categorical Encoding+Missing Data	⇑ 0.006
14	Classification issues-Sampling, Multi-Class	⇑ 0.003
20	Mathematic Formula	⇑ 0.003
28	?-Unknown	⇑ 0.001
25	Network Modeling	⇓ −0.003
5	Clustering	⇓ −0.007
9	NLP-Text Extraction, Scraping	⇓ −0.009
12	Technical guidance to a problem	⇓ −0.009
13	ML Reading Material	⇓ −0.011
19	Recommender System	⇓ −0.011
2	Q/A Guidelines	⇓ −0.014
30	Data Science-Education, Job	⇓ −0.026
8	Spark, Big-Data processing	⇓ −0.037
7	Decision Tree, Ensemble, Feature Selection	— 0.001
0	Plotting	— 0.001
22	Installation Help	— 0
17	Data Wrangling-Pandas	— 0
21	Dimensionality Reduction	— −0.001
15	Statistical Tests	— −0.001
29	Temporal Analysis-Prediction, TimeSeries	— −0.001
26	Probablistic Models-GAN, HMM, NB	— −0.002
1	Variable Correlation	— −0.002
31	NLP-Text modeling	— −0.005

6 Discussion

Our analysis of Stack Exchange and Reddit allows us to gain some insight into the thought process of people on a basic and advanced level. We find that the topics on Reddit have a wide variedty of topics as opposed to Stack Exchange. It can come from the way these two are maintained and supervised. The posting guidlines in Reddit are very flexible and loose but on the other hand, StackExchange enforces a strict set of guidelines as you can see in Figure 1, the second dominant topics in the entire datasets is *Q/A Guidelines*. Therefore Reddit is a good platform for people who would like to general questions and learn data

science as a beginner while Stack Exchange's users are advanced enough to skip this part and share more detailed and leading-edge Q/A. To prove this point, *Data Science learning material* in Reddit is very popular and has an increasing trend. But *Deep Learning* and *Neural Network* related topics are more popular in Stack Exchange. Another interesting observation is that people are more willing to share their questions and experience about *Data Science Career* on Reddit rather than Stack Exchange where the trend is decreasing. This makes Reddit a good source of knowledge base for career advice and recommendation.

Table 7: Result of trend for Reddit in terms of number of topics *(left)* and proportion of topics *(right)* per month

Id	Topics	%difference
45	DS Career	⇑ 241.05
22	ML questions	⇑ 155.17
31	Visualization	⇑ 149.21
2	Learning material for DS	⇑ 144.95
19	Deeplearning	⇑ 141.45
38	python, TF	⇑ 136.92
37	???-v8	⇑ 123
46	Population	⇑ 118.3
55	Living in Big City	⇑ 98.98
49	Food	⇑ 97.08
26	???-v3 spam	⇑ 85.38
27	Climate Change-2	⇑ 79.6
10	Music	⇑ 78.92
57	???-v10	⇑ 78.64
39	Working Hours	⇑ 76.6
54	ML-RL, Unsupervised	⇑ 75.43
9	Research Publication	⇑ 75.05
47	???-v9	⇑ 63.41
18	Posting Guidelines	⇑ 62.06
44	Probablity+Chance	⇑ 55.81
28	???-v4	⇑ 55.7
29	???-v5	⇑ 55.29
6	???-v1	⇑ 53.07
16	Hardware for computing	⇑ 51.07
15	Climate Change	⇑ 46.93
52	Language	⇑ 46
8	Sports Games	⇑ 34.56
13	Google Analytics	⇑ 26.87
40	Car	⇑ 26.29
42	Making points	⇑ 23.68
1	Post Removal	⇑ 11.93
23	???-v2	⇑ 4.23
4	Economy	⇑ 3.6
24	Cat, Dog, Skin cleansing	⇑ 2.04
25	Higher Education	⇓ −6.42
21	Gun issue in US	⇓ −7.1
12	Spam Removal	⇓ −49.11
20	Racism	⇓ −58.49
48	US Election	⇓ −111.13
53	Drug	— 62.67
7	Buying + Selling	— 46.73
41	Marriage + Children	— 20.94
5	Argument	— 18.9
30	Stats-Correlation	— 17.32
3	Cell phones	— 11.89
33	???-v7	— 10.2
32	???-v6	— 1.24
51	Movie+Show	— −0.68
43	Outbreak + Vaccine	— −8.5
14	Insurance	— −18.02
11	Country	— −18.45
34	Law and governance	— −18.7
0	Companies+Businesses	— −22.32
35	income, tax, investment	— −22.9
17	War	— −25.72
50	Poor vs Rich	— −33.12
36	Sexual Attraction	— −33.58
56	Religion	— −57.29

Id	Topics	%difference
45	DS Career	⇑ 0.015
37	???-v8	⇑ 0.011
2	Learning material for DS	⇑ 0.009
22	ML questions	⇑ 0.009
38	python, TF	⇑ 0.009
19	Deeplearning	⇑ 0.008
26	???-v3 spam	⇑ 0.008
31	Visualization	⇑ 0.008
49	Food	⇑ 0.005
9	Research Publication	⇑ 0.004
54	ML-RL, Unsupervised	⇑ 0.004
44	Probablity+Chance	⇑ 0.003
27	Climate Change-2	⇑ 0.003
57	???-v10	⇑ 0.003
47	???-v9	⇑ 0.003
29	???-v5	⇑ 0.003
16	Hardware for computing	⇑ 0.002
28	???-v4	⇑ 0.002
52	Language	⇑ 0.002
15	Climate Change	⇑ 0.002
6	???-v1	⇑ 0.002
32	???-v6	⇑ 0
4	Economy	⇓ −0.002
43	Outbreak + Vaccine	⇓ −0.002
30	Stats-Correlation	⇓ −0.004
34	Law and governance	⇓ −0.005
0	Companies+Businesses	⇓ −0.005
35	income, tax, investment	⇓ −0.006
14	Insurance	⇓ −0.006
36	Sexual Attraction	⇓ −0.006
21	Gun issue in US	⇓ −0.006
56	Religion	⇓ −0.007
17	War	⇓ −0.007
12	Spam Removal	⇓ −0.007
20	Racism	⇓ −0.007
5	Argument	⇓ −0.007
11	Country	⇓ −0.007
50	Poor vs Rich	⇓ −0.008
33	???-v7	⇓ −0.01
48	US Election	⇓ −0.014
55	Living in Big City	— 0.005
46	Population	— 0.004
10	Music	— 0.004
39	Working Hours	— 0.004
53	Drug	— 0.002
7	Buying + Selling	— 0.001
18	Posting Guidelines	— 0
23	???-v2	— 0
42	Making points	— −0.001
1	Post Removal	— −0.001
8	Sports Games	— −0.001
41	Marriage + Children	— −0.001
40	Car	— −0.001
13	Google Analytics	— −0.002
3	Cell phones	— −0.003
25	Higher Education	— −0.004
51	Movie+Show	— −0.004
24	Cat, Dog, Skin cleansing	— −0.004

A Topics Histogram

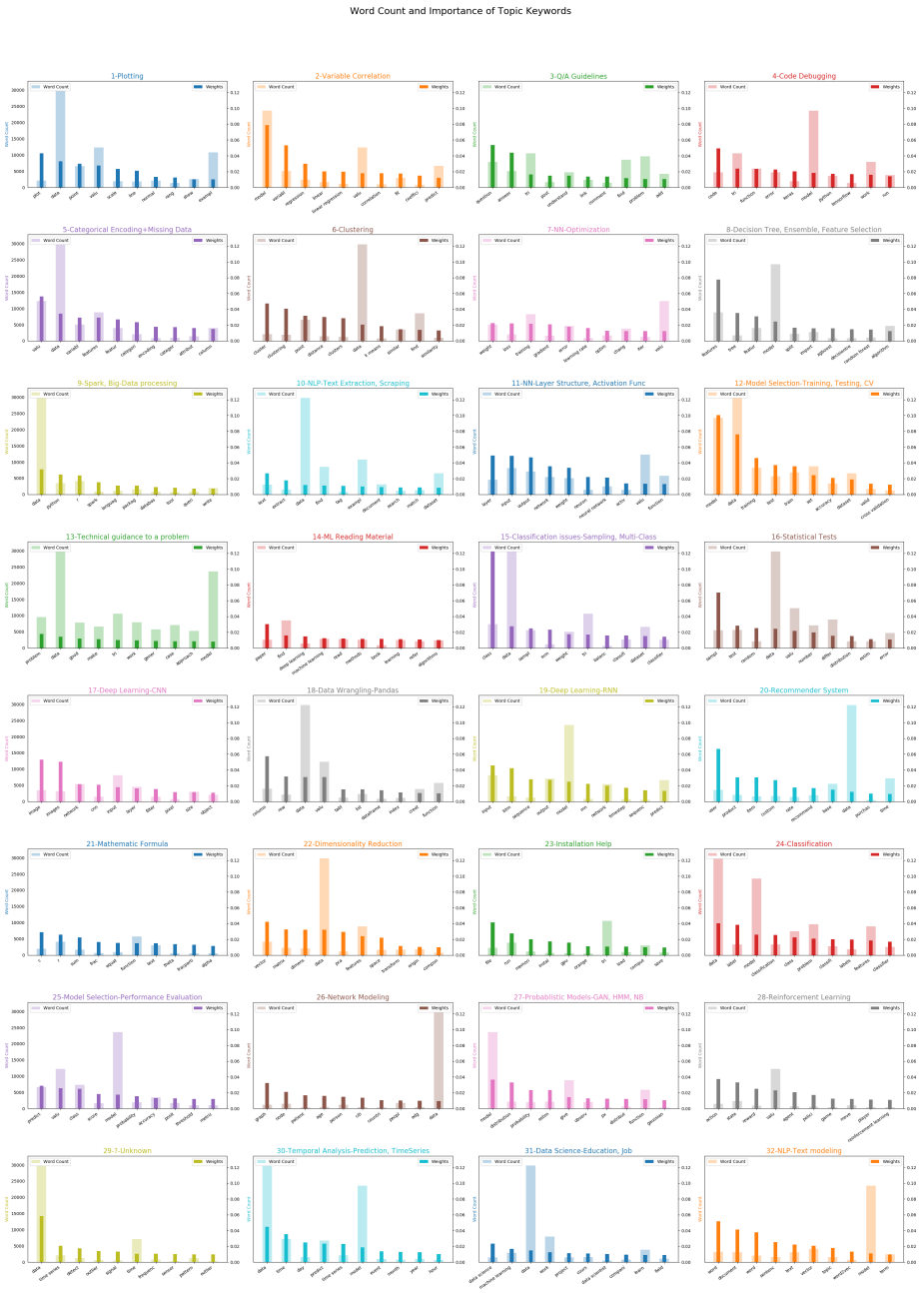


Figure 4: Stack Exchange dataset, weight and word count of top keywords in each topic

Figure 5: Reddit dataset, weight and word count of top keywords in each topic

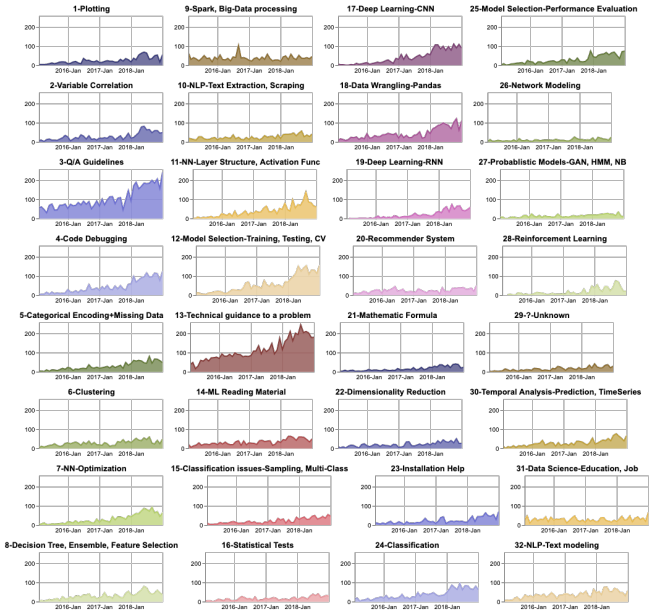


Figure 6: Stack Exchange dataset, temporal graphs for each topic in terms of number of documents per month

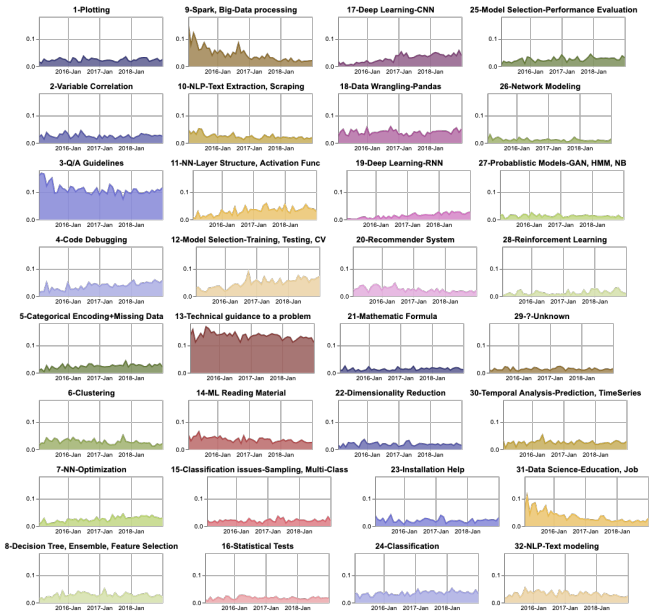


Figure 7: Stack Exchange dataset, temporal graphs for each topic in terms of proportion of documents per month

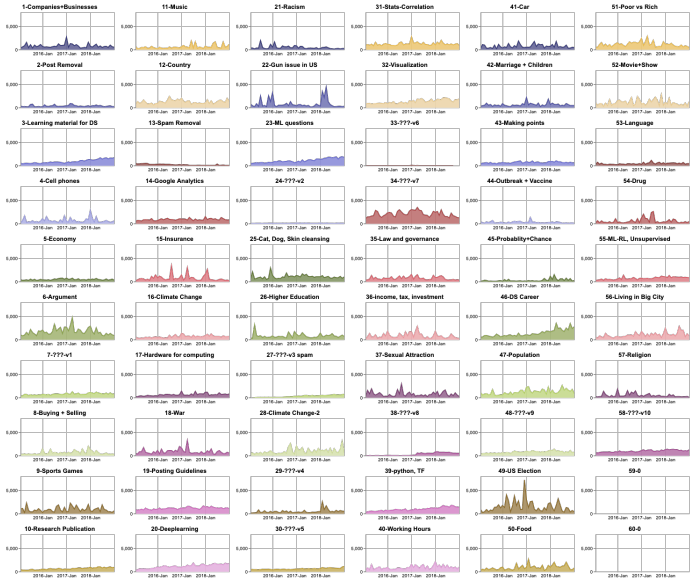


Figure 8: Reddit dataset, temporal graphs for each topic in terms of number of documents per month

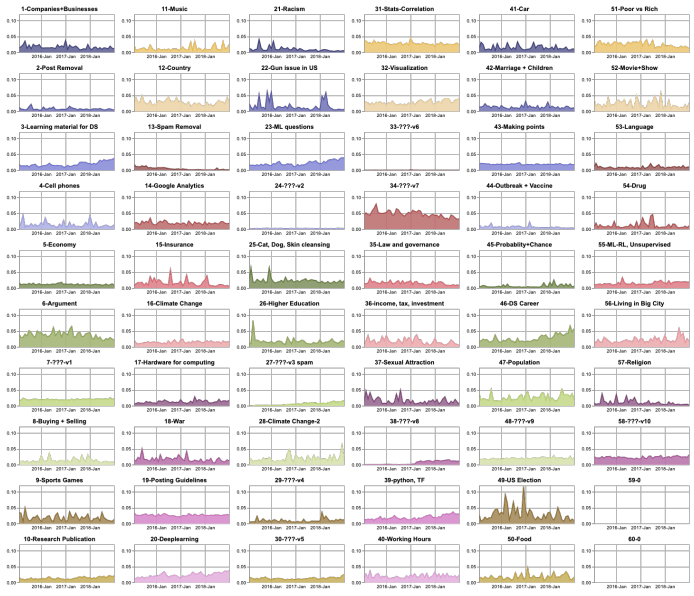


Figure 9: Reddit dataset, temporal graphs for each topic in terms of proportion of documents per month

References

[1] http://files.pushshift.io/reddit/.

[2] http://orc.gmu.edu.

[3] https://archive.org/download/stackexchange.

[4] https://datascience.stackexchange.com.

[5] https://www.stackexchange.com.

[6] http://www.reddit.com.

[7] http://www.reddit.com/dev/api.

[8] Blei, D. M., Ng, A. Y., and Jordan, M. I. Latent dirichlet allocation. *Journal of machine Learning research 3*, Jan (2003), 993–1022.

[9] Cox, D. R., and Stuart, A. Some quick sign tests for trend in location and dispersion. *Biometrika 42*, 1/2 (1955), 80–95.

[10] Geman, S., and Geman, D. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. In *Readings in computer vision*. Elsevier, 1987, pp. 564–584.

[11] Grant, S., and Cordy, J. R. Estimating the optimal number of latent concepts in source code analysis. In *2010 10th IEEE Working Conference on Source Code Analysis and Manipulation* (2010), IEEE, pp. 65–74.

[12] Jones, K. S. *Readings in information retrieval*. Morgan Kaufmann, 1997.

[13] Manning, C., Raghavan, P., and Schütze, H. Introduction to information retrieval. *Natural Language Engineering 16*, 1 (2010), 100–103.

[14] McCallum, A. K. Mallet: A machine learning for language toolkit.

[15] Mei, Q., Shen, X., and Zhai, C. Automatic labeling of multinomial topic models. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining* (2007), ACM, pp. 490–499.

[16] Tan, C.-M., Wang, Y.-F., and Lee, C.-D. The use of bigrams to enhance text categorization. *Information processing & management 38*, 4 (2002), 529–546.

[17] Wallach, H. M., Murray, I., Salakhutdinov, R., and Mimno, D. Evaluation methods for topic models. In *Proceedings of the 26th annual international conference on machine learning* (2009), ACM, pp. 1105–1112.