Technology Trend Analysis and Prediction

Yahui Cui
New York University
New York, NY
yc3329@nyu.edu

Jiachen Zhu
New York University
New York, NY
jz3224@nyu.edu

Ranjita Rajeeva Shetty
New York University
New York, NY
rrs462@nyu.edu

Abstract—

Technology trend analysis offers a flexible instrument to understand both opportunity and competition for emerging technologies.With this project we endeavor to analyze questions posted on StackOverflow, packages information from libraries.io and academic papers posted on arxiv.org and perform exploratory analysis on these datasets to find out the technology trend in past 11 years in time series and relationship between certain technology. This analytics can be used to identify 'raising' technology in the nascent stage and also to identify the dying technology based on its adoption rate.Using this analytics one can search any word and see how the technology trend around the word has been changed over the period.The data captured from the dataset goes back to 2007 up to present day. This is where the big data analytic plays critical role to analyse such a colossal dataset. Various big data technologies can be used to effortlessly analyze hundreds of GBs of data within few seconds and expedite the process.

Keywords—Technology Trend, Prediction, Big Data

I. Introduction

With our analytics, we conducted a text mining analysis on StackOverflow dataset which is the popular Q&A site for the programmers worldwide with its database of questions and answers, Libraries.io dataset which is an open source web service that lists software development project dependencies and alerts developers to new version of the software libraries they are using and ArXiv.org dataset which is a highly-automated electronic archive and distribution server for computer science academic papers including summary and published data.

Applying mapReduce jobs, distributed machine learning and natural language processing algorithms, we got the trend of technology in time series and relationships between certain technology. Our analytic will help programmers, Tech companies, researchers, investors and students to identify and learn the trending technology and some potential technologies which may become popular in the future.

II. Motivation

Time series analysis shows the trend of technologies, indicating the core technology and the popular technology during the period we focused on. Analysis on relationships between technical topics shows the connections between certain technologies, revealing the potential needs of certain technology and the future important technology based on today's core and popular topics.Using our analytics we can identify 'raising' technology in the nascent stage and also we can also identify the dying technology based on its adoption rate.

III. Related Work

The popularity and trend of technologies can be captured through different data sources. Different kind of data set provides information of technologies on the unique perspectives. Intuitively, questions and application of technologies always indicate the popularity of a specific technology. Academic paper titles, which are always well structured, indicate the basic technology the paper focusing on or used by researchers. Thus, quantitative analysis on these data source will unveil the popularity of technologies and dependence between technologies, indicating the trend of future technology.. This kind of analysis includes predicting score if the questions relating to technologies, academic paper classification as well as software packages dependency analysis.

[1] For technology question analysis, Haifa Alharthi, Djedjiga and Olga Baysal performed the analysis of questions content on Stack Overflow. This paper talks about predicting the score of the question on StackOverflow based on sixteen significant factors associated with questions' format, content and interactions that occur in the post. Authors performed the multiple regression analysis and found that questions' length of the code, accepted answer score, number of tags and count of views, comments and answers are statistically significantly associated with the score of the questions. These analysis can help Q&A community to improve the quality and content of the shared knowledge. Each question and answers can be scored by summing up the upvotes and downvotes received. If the author of the question mark answer as an accepted which denotes that answer meets the author's need.The study used the data dump of the Stack Overflow content which was

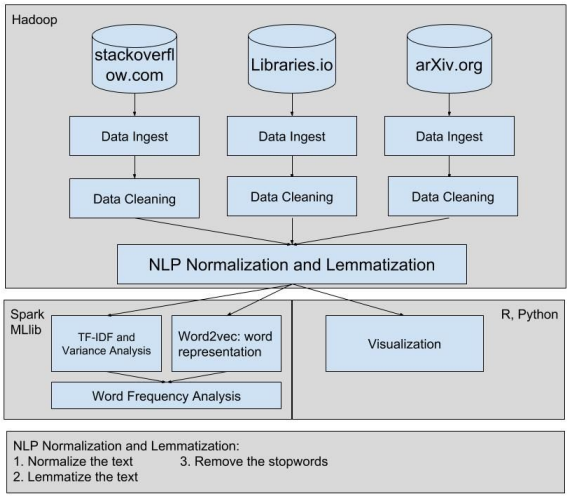
published from 2009 to 2014.The data set was obtained from the MSR challenge 2015. Out of the 8 XML files of dataset study used only two “posts” and “users”. Closed questions, community owned posts were excluded from the study. The records of accepted answers and owners were mapped to their questions. Empirical analysis was conducted on sixteen independent variables which are questioners’ reputation, the length of the question, title, number of answers, favourites ,comments, number of questions’ views, code blocks, links, length of the code, number of tags, the score of the accepted answer, time to accepted answer, the ratio of body length to paragraphs, polarity and subjectivity value with the dependent variable, the score of the question. Study showed that The greater the number of tags, the greater the number of potential users who may visit the post. Study also filtered out the question which does not have an accepted answer as a question with the good quality is expected to receive a quick and high-quality accepted answer due to the question clarity. Data analysis was performed to understand the correlation between the score of the questions and their number of answers, their accepted answers. Multiple linear regression analysis were performed using python library, Statsmodel to predict the dependent variable Y(score of the question) by knowing the value of the independent variable x1,x2,x3..etc. The coefficient of determination(R-squared) determines how well the regression line fits the data. The regression analysis demonstrated that the chosen independent variable have a significant explanatory power to predict the score of the question. There is also scope for the improvement by performing additional analysis to the prediction of the question.

[2] Other work on technology trend prediction includes Kondo’s work by identifying element technology of research articles through structured titles. In this paper, the author search the element technology based on particular words which indicates the occurring of technology words. They also partitioned the technology key works into classes including “head”, “goals” and “method” with the first two indicating the research field and the last representing research method.

[3]For research on application and software packages analysis, Decan etc. conducted an empirical comparison analysis on dependency networks evolution of different software packages. In this paper, the author applied time series analysis of the package update times, the evolution of number of dependency for each package and survival analysis on the package update frequency. Through the statistical analysis above, the author analyzed the impact and quality of each package, revealing the popularity of technology from the perspective of applications.

[4]For research on retrieving information from academic paper, Zhang etc. conducts classification of academic paper through text mining of abstract. In this paper, the authors hope to develop a system to help graduate students reduce the time of article screening and classification. The authors apply three different clustering algorithms (K-means Clustering, Hierarchical Clustering and Spectral Clustering) on the abstract of academic articles to find clusters of those articles. The paper also introduces how they collect the data and the preprocessing process. The result shows the Hierarchical Clustering and Spectral Clustering have similar results, and the result of K-means clustering is not ideal. It could result from the outlier in the dataset and K-means can't handle it properly. To conclude, it is useful to use text clustering technique to find out which articles are more relevant to one's research.

IV. DESIGN



After gathering data from all three websites, we did some data ingesting and data cleaning work using mapreduce and hdfs. During the data cleaning process, we just removed everything other than the title, summary, and the published year. Then, put all the result in HDFS for next step.

With three completed datasets, we used SparkNLP[7] to do text normalization and lemmatization, removed all the characters that are not part of a word and changed each word to its original root. Merged the result with the published year, we got cleared datasets with which we could start to do some analytics work.

The first thing we tried is TF-IDF and variance analysis. We just first calculated the TF-IDF for each word in each document using mapreduce. However, we found out that does not work. Since most of documents we have are really short, so there is a lot of non-technology words with really high TF-IDF score. We also found out variance analysis does not work either.

Then we did some Word2vec analysis on our cleaned datasets. After training the model, we could see that Word2vec could help us identify some important words and we could also use the model to analysis the relationship between two words and the change of relationships.

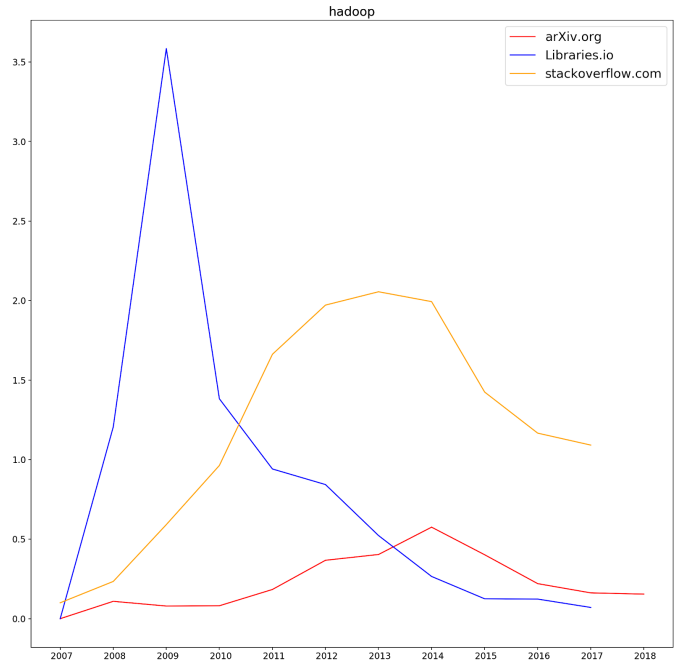
Then we did a word frequency analysis on our cleaned datasets. We applied mapreduce to count the percentage of each word appearance in each dataset for each year, and the result of word frequency analysis is used to get our final analytics results and also the visualization results.

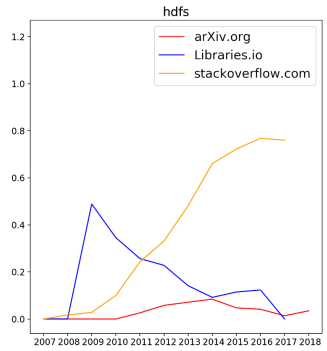
V. Results

With the datasets, we did two different analysis. One is on technology terms related to Big Data, another one is on technology terms related to Deep Learning. Also we will present our relationship analysis result based on Word2vec.

1. Technology terms related to Big Data

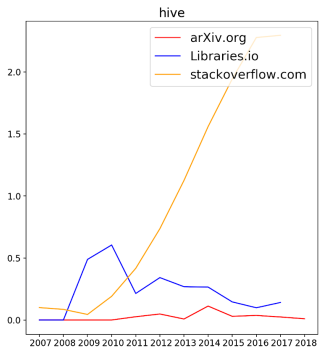
The first three term we searched is hadoop, mapreduce, hdfs.

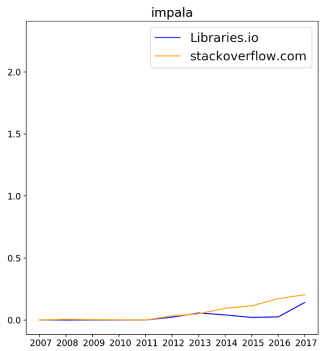




The figure shows that both hadoop and mapreduce is declining in recent years. hdfs is also declining in terms of open source libraries, but its users are keeping growing and research papers remain the same level for each year.

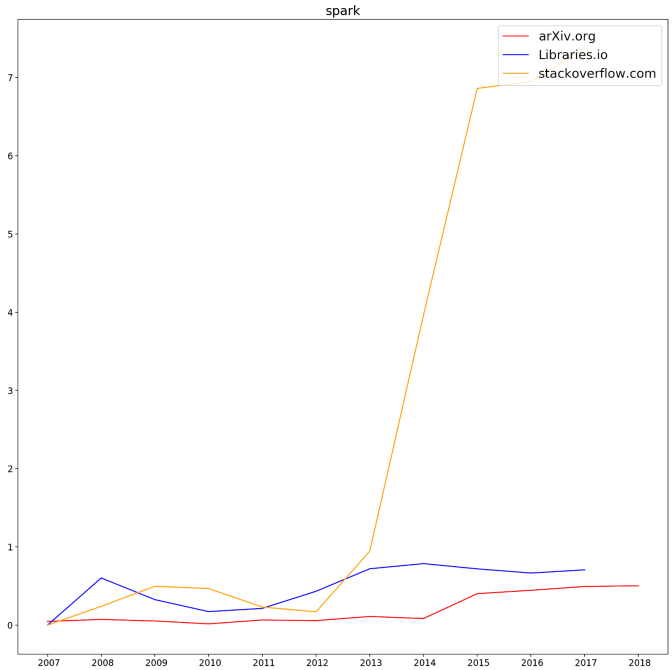
The second group terms we search is hive and impala.

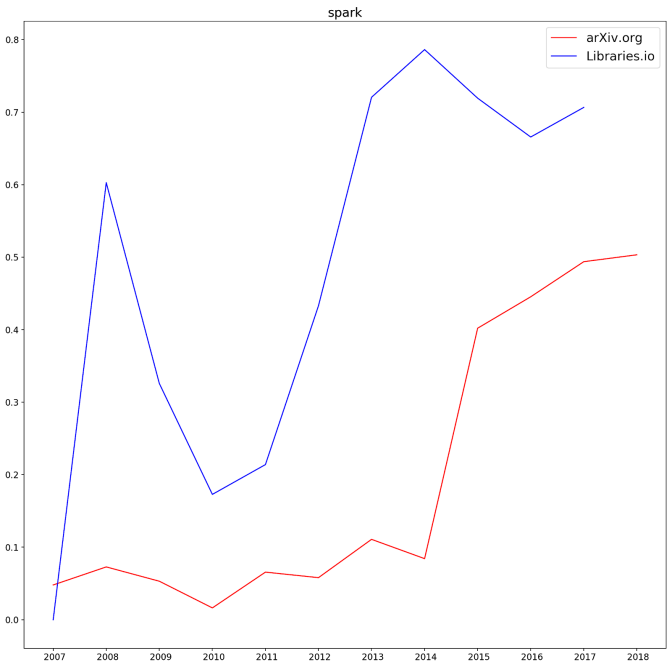


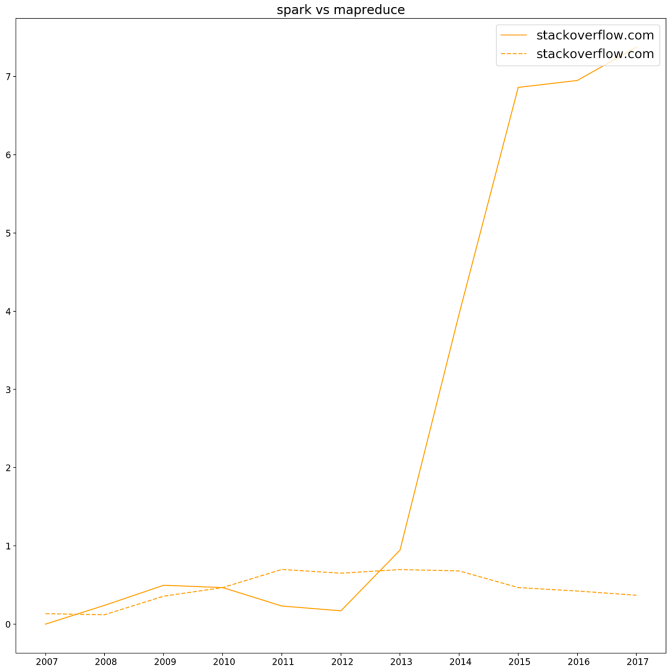


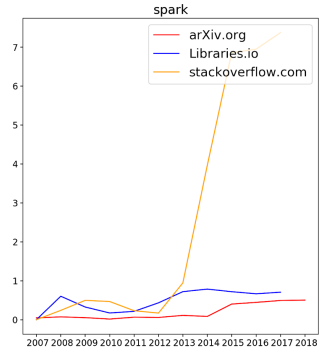
You could see that hive has the same pattern as hdfs. The number of users is still growing. On the other hand, although impala is growing, but comparing with hive, it is still not a significant technology.

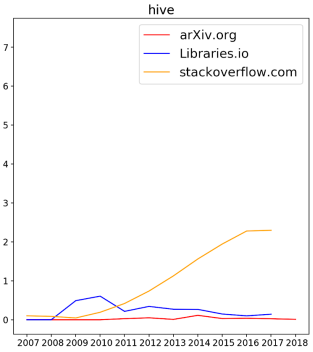
Next, we searched spark, which is a really popular technology in Big Data field.





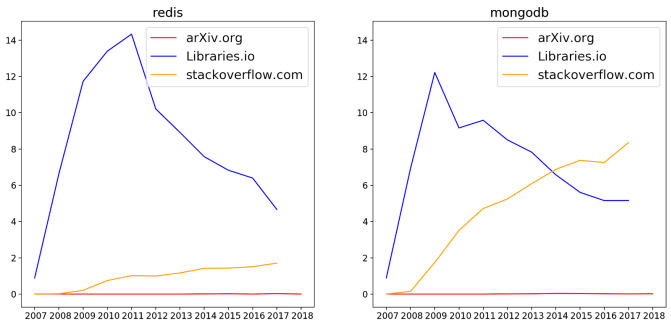


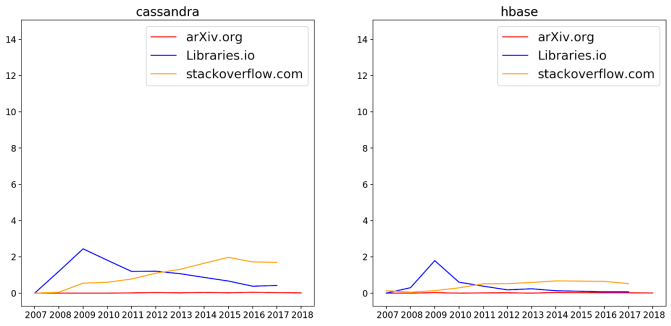




You could see in all three categories, spark is growing. Also it already surpassed mapreduce and has much more users than hive.

The last group of terms we studied is related to NoSQL database. They are redis, mongodb, cassandra and hbase.

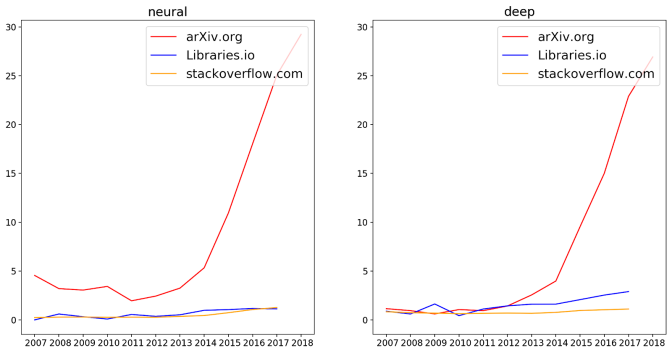




It shows that mongodb is the best in terms of the number of users. The next is redis. cassandra and hbase are less popular than the other two. By comparing the result of external database ranking[5], we could find it is matched our result.

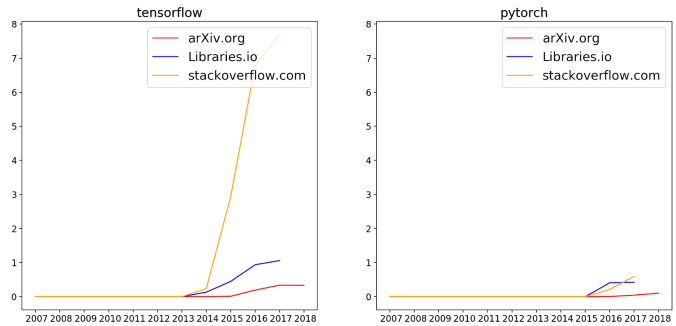
2. Technology terms related to Deep Learning

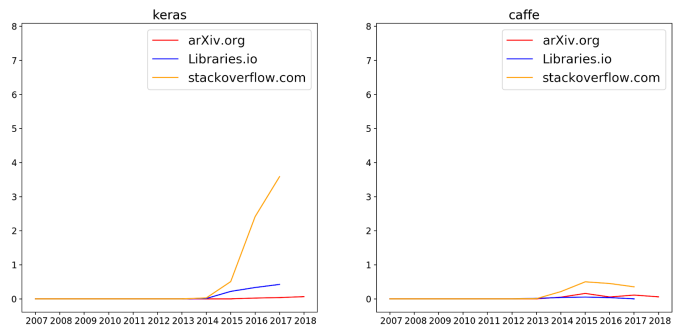
The first two terms we searched is neural and deep.



It is obviously that both of them growing a lot in terms of the number of research papers. It matches our expectation.

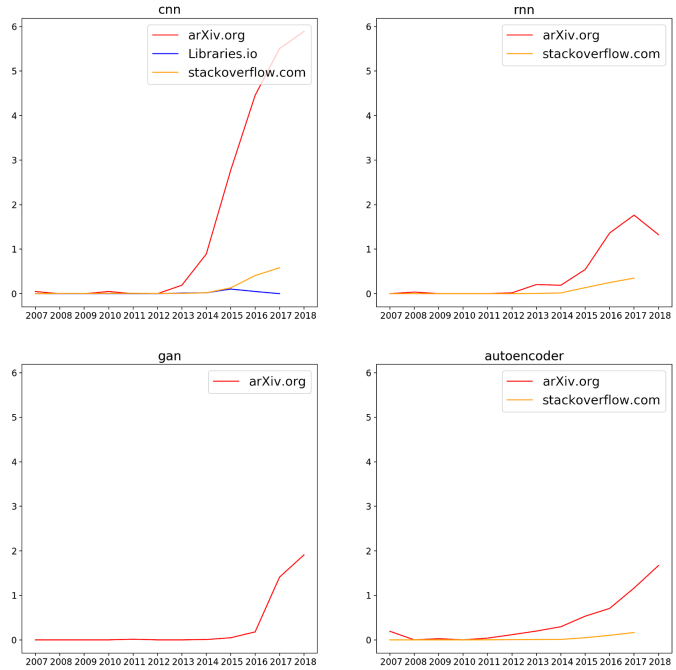
Then we searched four names of deep learning packages, tensorflow, pytorch, keras, caffe.





Tensorflow and keras are obviously the best and the second best in terms of the number of users. pytorch and caffe are less competitive. By comparing the result of external deep learning ranking[6], we could find it is matched our result.

At last, we compared four different algorithms of deep learning.



We could see CNN has the highest number of research paper. And all other three algorithms are almost same.

3. Relationships between technologies

In order to find the relations between technologies, we applied Word2vec model to get the representations of words. Word2vec model provides an efficient method to capture the semantic relationships between words in large corpus. By training Word2vec models, we are able to search the relations between technologies through distances calculated from word vectors.

We implemented Word2vec model on Stackoverflow, libraries,io and arxiv.org dataset. In order to search the trend of relations between every pair of technologies, we separated the training dataset into several parts based on year and trained unique model for each part. The results of Word2vec model are in Table 1. We searched the keywords "data", "application" and "algorithm" in our model and examined the words closely related to the keywords in each time phase. By doing this, we could find technologies closely related to the keywords and how the related technologies change over time.

For keyword "data", we could find technologies related with data storage have always been popular. With the trend of big data, tools to deal with large data in parallel become popular in all of the datasets. The word "hdfs" is a good example. Besides new tools to store data, we found technologies related to analytic methods and application also become popular. Words including "statistics", "machine-learning" indicate the popularity of methods in data mining and analysis. Words like "gis" and "healthcare" indicate the popular fields that involves with big data.

For "application", the three datasets show the same trend of technologies related to applications. Results from all three datasets shows that application with "android" and "opensource" are always popular. In addition, the popularity of applications with "distributed" and "parallel" increases through time. Moreover, we also found applications related to microservice are becoming popular in recent years. This may indicate the future popularity of the mocroservice.

For "algorithm", we found there is divergence between academic field and industry. Since data from arXiv represent the hot area in academic field while stackoverflow and Libraries.io are more related to industrial application. We found when searching close word of "algorithm" in the model trained by arXiv dataset, algorithm is always related words including "approximation", "heuristic" and "greedy", while in model trained by Libraries.io dataset, algorithm is more close to certain data structure including "tree" and "hash". This implies that academic field may focus more on research about abstract method while industrial prefers using data structures to solve problem.

To conclude, we found the technology trend in the three datasets sometimes shows the same pattern while there are also divergences. The divergences mainly come from the difference between academic field presented by arXiv and industry fields represented by Libraries.io. The result from keyword "algorithm" is a good example of such divergence.

Table 1-1a. Results of Word2vec 2011-2014

Keyword: data				
year	2011	2012	2013	2014
arXiv	data-warehouse nosql massive data data-mining	massive data data-warehouse basket analysis DNA microarray	high-throughput output data-ingestion genomics data-warehouse	big-data bench nosql crawlers gis analytics
Libraries.io	storage distributed transaction plot retrieve	sql storage distributed query tracing trading	distributed big-ingestion packet biomedical/biological	object value persistence immutable biological
Stackoverflow	dataset information database relational blob datatable	dataset database dump csv coredata keyvalue	huge sqllite populate record tabular	json hdfs database response bulk

Table 1-1b Results of Word2vec 2015-2018

Keyword: data			
year	2015	2016	2017-2018
arXiv	analytics digest regenerate ehr (a system related to health)	volumes unlabelled/unalig-ned(machine learning) visualization analytics	cleaning voluminous/massive Hadoop big data bench edgar(a electronic system) healthcare
Libraries.io	mapped nested structures cross-filter gps snake case	kmeans taxonomy msgpack genomic profiling	key-value pandas database bench(big data bench) schema
Stackoverflow	dataset forecast hierarchical datatype database	statistics information historical in memory entire	realtime historical machine learning regression millions parquet statistics dataset

Table 1-2a. Results of Word2vec 2011-2014

Keyword: application				
year	2011	2012	2013	2014
arXiv	aneka gridcertlib gaming telecommunication architecture	aspnet android mlearning micro controller	adaptor pilot data-ingestion metamodel parallelizing	sociology android mcl aerospace javascript
Libraries.io	opensource database deployment scheduling docker engine	development multifronted architecture web machine	scheduling recommendation high performance golanger container docker	splatform/testflight/micro paas opensource engine frontend
Stackoverflow	webapplication vmware kit opensource platform	develop webapp project sso android	intranet docker jee webapp mono	opensource platform android webapp development

Table 1-2b Results of Word2vec 2015-2018

Keyword: application			
year	2015	2016	2017-2018
arXiv	browser android just-in-time banking mlearning automotive	cado javascript open source android ecosystem	cloudsim/cloudbased microservices/microsimulation blocksci high performance docker
Libraries.io	snapshots workflow websites	cocoapods(manager for Obj-C app) app workflow labstack echo	crawling rocket micro service/framework workflow chatbot webapp full stack
Stackoverflow	webapplication service android weblogic platform	webapplication clickonce platform Android desktop app	webapp microservices glassfish jetty gae tomcat startup kestrel
			intranet Development

Table 1-3a. Results of Word2vec 2011-2014

Keyword: algorithm				
year	2011	2012	2013	2014
arXiv	approximation linea- time non-iteration subquadratic quasipolynomial	heuristic approximation rounding distance block-coordinate	greedy local-search bellman approximation	memetic iterative minplus maxlog
Libraries.io		solver stemming neural	hash tree	stemming huffman aprior) knn union find
Stackoverflow	quicksort complexity lineartime computation	polynomial dijkstra classification similarity	greedy quicksort tree	neighbor huffman quicksort iterative

Table 1-3b Results of Word2vec 2015-2018

Keyword: algorithm			
year	2015	2016	2017-2018
arXiv	approximation greedy	approximation delta-coloring	leaps-and-bounds ptas) rmgd
Libraries.io	heapsort levenshtein backpropagation hashing mergesort weighted bandit	aho-corasick genetic hashing levenshtein convex classification	eucleadian matrix geometric embedded
Stackoverflow	mergesort shortest approximation recurrence	euclidean greedy longest approximation	subsequence traversal euclidean nonlinear

VI. Future Work

1. N-gram implementation and wordcount

Due to time constraints, our model only supports one word searching at this time. In the future, we plan to conduct n-gram algorithms to parse the dataset more times and find pattern about technologies represented by phrases including “deep

learning", "neural network". This work will improve our model accuracy and support more general applications.

2. Retrain Word2vec model

We will retrain the Word2vec model using the newest version of our stemmed dataset to get a more accurate representation of the words in each dataset. Combined the results from N-gram, we plan to search the words or phrases with highest frequency and to analyze the related technologies trend.

VII. Conclusion

In Word2vec model, we found that the three datasets sometimes shows the same technology trend while also shows divergences. The divergences mainly come from the difference between academic field and industry fields represented by Libraries.io. The result from keyword "algorithm" is a good example of such divergence.

From the two example (the Big Data example and the Deep Learning example) based on our word frequency analysis model, we could find out that our analytics model have great potential to analyse technology trend in most fields. It is really easy to understand, and has great accuracy, which match most of ranking indices. More importantly, it is based on opendata, everyone could get the data and do their own analysis on it.

Acknowledgment

Thanks NYU HPC for providing distribute computation environment.

Thank StackOverflow, Libraries.io and arXiv.org for providing data source.

References

1. H. Alharthi, D. Outioua and O. Baysal, "Predicting Questions' Scores on Stack Overflow," *2016 IEEE/ACM 3rd International Workshop on CrowdSourcing in Software Engineering (CSI-SE)*, Austin, TX, 2016, pp. 1-7. https://ieeexplore.ieee.org/document/7809391/
2. Y. Zhang and Y. Wan, "How to find valuable references? Application of text mining in abstract clustering," *2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, Guilin, 2017, pp. 2201-2205. https://ieeexplore.ieee.org/document/8393112/
3. Kondo, Tomoki, et al. "Technical trend analysis by analyzing research papers' titles." *Language and Technology Conference*. Springer, Berlin, Heidelberg, 2009.
4. Decan, Alexandre, Tom Mens, and Philippe Grosjean. "An empirical comparison of dependency network evolution in seven software packaging ecosystems." *Empirical Software Engineering* (2018): 1-36.
5. Db-engines.com. (2018). DB-Engines Ranking - popularity ranking of database management systems. [online] Available at: https://db-engines.com/en/ranking [Accessed 6 Aug. 2018].
6. The Data Incubator. (2018). Ranking Popular Deep Learning Libraries for Data Science. [online] Available at: https://blog.thedataincubator.com/2017/10/ranking-popular-deep-learning-libraries-for-data-science/ [Accessed 6 Aug. 2018].
7. Nlp.johnsnowlabs.com. (2018). John Snow Labs Spark-NLP. [online] Available at: https://nlp.johnsnowlabs.com/index.html [Accessed 6 Aug. 2018].