

# Thematic Exploration of YouTube Data: A Methodology for Discovering Latent Topics

By  
Clinton Daniel

In a study published in 2015 by Ganesan, Brantley, Pan, and Chen (Ganesan, Brantley, Pan, & Chen, 2015) researchers recognized that there is a problem with the search process when trying to visualize the correlation between large collections of documents and a given set of topics. Chaney and Blei emphasize in a 2012 study the importance of science, industry, and culture to have the ability to explore the hidden structures found within large collections of unorganized documents (Chaney & Blei, 2012). Wang and Blei published an article in 2011 citing the difficulty of finding and recommending relevant scientific research papers to communities of researchers (Wang & Blei, 2011). Finally, an article written by Roberts, Stewart, and Airoidi in 2016 discusses the popularity of statistical models and how they are used for exploring large collections of documents to measure latent linguistic, political,

and psychological variables in the social sciences (Roberts, Stewart, & Airoidi, 2016).

All of these studies can be aggregated together to describe the problem that continues to challenge researchers and information management practitioners whom are attempting to explore a latent set of topics which form a common theme within a large collection of documents. Documents are being collected from a growing number of sources that continue to offer the problem of complexity and lack of intuitive correlation. Novel research methods that include a mixture of technologies working together as a framework are needed to address these challenges found within a large collection of documents. This research method proposes a framework that can be used by researchers and practitioners to discover latent topics found within a target set of YouTube video transcript documents.

**An automated research method that uses the topic modeling algorithm called Latent Dirichlet Allocation (LDA) to discover latent topics and explore potential themes in YouTube transcript data.**

**Keywords:** Topic Modeling, Text Mining, Latent Dirichlet Allocation, LDA, YouTube Transcript Data, Research Method

## The Research Problem

Research has emerged over time within the social sciences to address the discovery of latent information found within documents (Rana, Cheah, & Letchmunan, 2016). Large collections of documents are being archived from sources on the internet such as social networks, blogs, videos, wikis, academic articles, trade journals, patents, books, and magazines. These archives can serve as a source of knowledge for research analysis and practical understanding of what is happening within a specific topic of interest. Typically, if someone wanted to search for information that can be found within one of these documents, they would start with typing key words in a search engine and then execute a search. The search engine would then propagate through an ocean of documents and metadata using a search algorithm to locate the target documents that include the specific terms. The problem with this approach is that the search process is targeted at specific terms and there is no utility for the discovery of hidden or latent information. The discovery of hidden information is often found in the form of a theme or topic.

Documents often contain a series of themes that can be formed from a clustered set of words (Blei, 2012). For instance, there may be a single post from a software development blog titled, “New CakePHP 3.4 Red Velvet. Faster. Stronger. Tastier” that discusses a new rapid application development framework to be released in the open source community. As you read through this blog post, it may not include the words “rapid application development” or “open source community”. However, these words can be used as descriptive themes that tell a story about a cluster of words found within the blog post. The challenge of the blog post observer is understanding how they can objectively structure a theme out of the set of clustered words. The thematic exploration of data could offer a novel way for practitioners and researchers to discover new knowledge within a collection of documents.

## Description

On February 16, 2017 YouTube announced in its official blog that it has automatically captioned over 1 billion videos (Kaver, 2017). These captions are made possible with a combination of Google’s automatic speech recognition (ASR) technology and YouTube’s caption system. The original intent for captioning YouTube videos was to provide more accessible content for the hearing impaired. In addition to videos displaying closed captioning, YouTube offers an ex-

ported transcript of the closed captioning text. A collection of YouTube video transcript documents may offer a unique opportunity for researchers and practitioners to perform thematic exploration of the content with the goal of discovering topics that may lead to new knowledge.

Discovering topics within a collection of documents typically involves the estimation of latent topics for a given corpus using a Topic Modeling algorithm such as LDA. Blei describes the LDA algorithm as, “a generative probabilistic model for collections of discrete data such as text corpora.” (Blei, Ng, & Jordan, Latent Dirichlet Allocation, 2003) The LDA algorithm will be applied to a collection of YouTube video transcripts that were gathered using an innovative data collection and analysis research method. This method involves the collection of YouTube video transcripts and their metadata followed by loading each individual document text into a relational database. Once the text data has been successfully loaded into a relational database, the video transcript data can be analyzed and joined to corresponding metadata with the goal of generating a targeted data set for export. The targeted dataset is identified by a preliminary analysis based

The thematic exploration of data could offer a novel way for practitioners and researchers to discover new knowledge within a collection of documents.

on a specific business or research question. Then, the video transcript’s text will be exported as a collection of separate text files from the relational database to a target directory to form the corpus.

The corpus can then be evaluated with the LDA algorithm using an R topic modeling package. R will then be used to visualize the results of the LDA algorithm for further analysis.

## Typical Protocol

The following outlines all of the activities involved in applying this research method in a practical context.

### Prerequisites for Applying the Protocol

This research method has been designed not only for a researcher, but for the information management practitioner who has experience working with data using relational databases, ETL (Extraction, Transformation, Load) workflow tools, web scraping programming languages, and statistical analysis programming packages. The following is a summary of the technologies used in this research method to serve each of these purposes.

**Relational Database:** Microsoft SQL Server (MSSQL) will be used as the research method’s relational database management system (RDBMS). MSSQL was selected as the RDBMS of choice due to its recent expansion of Machine Learning Services (MLS) that have been integrated with its

database engine (Takaki & Kess, 2017). MLS provided by MSSQL has support to build and deploy machine learning solutions, in both Python and R, that use SQL Server data. This tight integration of R, Python, and MSSQL allows the researcher and/or practitioner to build this solution and deploy it in a production environment using Microsoft SQL Server Integration Services. For this reason, MSSQL is not only a good choice for this research method but for future research that further extends this approach using these technologies. In this research method, MSSQL will be used to develop relational database tables that will store all of the YouTube transcript text and metadata. Additionally, Transact-SQL, Microsoft's proprietary extension of the Structured Query Language, scripts will be used to perform ETL and preliminary analysis operations.

**ETL Workflow Tool:** Microsoft SQL Server Integration Services (SSIS) will be used as the research method's ETL workflow tool. ETL workflow tools are responsible for automating the process of extracting data from a source, transforming the data into a format that can be used for analysis, and loading the data into a destination relational database management system (or other type of destination structure). Microsoft SSIS was the ETL workflow tool of choice for this research method because of its integration with the native MSSQL services. Microsoft SSIS will be used to automate the extraction of the YouTube.SRT file and metadata.TXT file, transformation of the text data, and loading into the target MSSQL database tables.

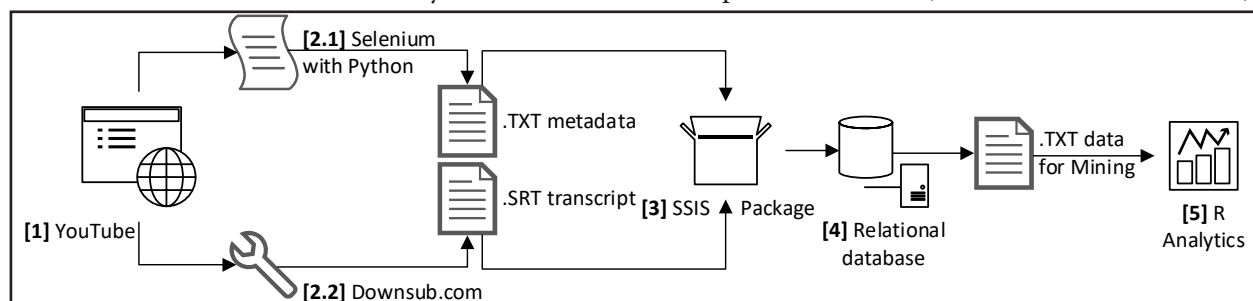
**Web Scraping Programming Language:** Selenium (What is Selenium?, n.d.) with Python (Python, n.d.) will be used to automate the browsing of YouTube.com in order to scrape the metadata about the videos. The metadata scraped from YouTube.com will be stored on the local file system as a.TXT

file for later processing. Additionally, Selenium with Python will be used to automate the process involved with Downsub.com to generate a .SRT file that includes the YouTube transcript data from a targeted video id. Selenium is a web browsing automation tool that binds with Python and includes Web driver support for many popular web browsers. Python was selected for this research method due to its potential for automation using Microsoft's new Machine Learning Services in MSSQL.

**Statistical Analysis Programming Package:** R (The R Project for Statistical Computing, n.d.) will be used to compute the topic modeling algorithm called LDA and provide output for analysis (Grun & Hornik, 2017). Additionally, R will be used to generate some of the visualizations used to better understand the output of the LDA algorithm. Several R packages were used to accomplish the analysis to include: topicmodels, tm, tidytext, reshape2, and ggplot2. R was selected for this research method because of its potential for automation using Microsoft's new Machine Learning Services in MSSQL and its easily accessibility to the topic modeling algorithms. Additionally, R packages, such as topicmodels, have been created to directly interface with the C code for LDA that was written by many of the inventors and extenders of the LDA algorithm (Grun & Hornik, 2017). Therefore, it is common to find analysis and visualizations that have been published in academic research with output from R.

## Steps in Applying the Protocol

This research method involves the following steps to screen the YouTube video content, extract captioning transcript data, extract important video metadata associated with the captioned transcript, import (transform and load) all data into a relational database, process all data (additional transformation),



**Figure 1:** This diagram illustrates the technology used in the steps involved with applying this research method protocol. [1] Includes the process of screening the YouTube video content. [2.1] Includes the process of extracting important video metadata associated with the captioned transcript using Selenium with Python. [2.2] Includes the process of extracting the captioning transcript data from Downsub.com. [3] Includes the process for import and processing all data into the SQL Server relational database using SQL Server Integration Services. [4] Includes the process for performing all additional processing of data and preliminary analysis inside a relational database (Microsoft SQL Server). [5] Includes the process for performing topic modeling and analysis of the output from the LDA algorithm using R.

and analyze data using statistical programming packages. First, you can observe a diagram that visualizes an overview of which technology is used to accomplish each of the steps involved (see Figure 1). Next, each functional step is summarized in a way that describes how the technology is used to accomplish the research method protocol.

### **[1] Screen the YouTube Video Content**

Step 1 in this method starts with a search term being used in YouTube's search engine to first find a list of YouTube videos that meet an area of interest. For instance, "CIO cloud adoption" could be used as a search set of terms to explore this research method. Next, the researcher would narrow down the results of the search by reviewing the metadata available from a specific YouTube video of interest. Upon review of the metadata, a screening of the video is required by viewing and listening to its content for at least 30 to 60 seconds. Brief audio and visual screening of the video, combined with the review of the video's metadata, validates that the content is within the context of the researcher's interest. Once a video of interest has been screened and the researcher is interested in collecting its data, then the URL will need to be copied to a temporary location for a retrieval at a later time. The following is a sample of the required format to extract the YouTube videos transcript data: <https://www.youtube.com/watch?v=b-7TiqyVCmE>. The most important part of the URL is the video ID. The video ID is identified as the string after the "v=" portion of the URL. For instance, the video ID of this URL is "b-7TiqyVCmE".

#### **[2.1] Extract Important Video Metadata Associated with the Captioned Transcript**

Step 2.1 involves Selenium with Python (Muthukadan, n.d.) programming that will be used to extract the associated YouTube video metadata. A custom Selenium with Python program will be used to extract the metadata and save it in a local text (.TXT) file. Some data will be generated and appended to the metadata by the Selenium with Python script to provide further information that is not otherwise provided by YouTube. For instance, the video ID, video URL, date and time of extraction, and YouTube search terms will be added to the generated text file. This additional information will be used for analysis at a later time. The extracted text file is in a semi-structured format and will be processed in a structured format by another process at a later time in this research method.

#### **[2.2] Extract Captioning Transcript Data**

A tool is required in step 2.2 to extract the YouTube video captioning transcript. The tool can be custom programmed with a programming language such as Python using Google's YouTube API or there are free

web-based tools available for data extraction. A web-based tool available at <http://downsub.com/> will be used to explore this research method and extract the transcript data. After navigating to <http://downsub.com/> the researcher will need to enter the YouTube URL of interest in the "Download" tool. This process can be automated using Selenium with Python to re-create the steps needed to acquire the YouTube transcript. However, the manual steps are documented here for the purposes of understanding the manual process. After clicking the "Download" button, DownSub will extract the transcript data from the YouTube video and make it available for download to your local computer as a .SRT file. These .SRT files, or SubRip files, contain the recorded subtitles and timings associated with the specific YouTube video. SubRip files can be opened with text editors to view the recorded subtitles and timings. The SubRip file extracted from DownSub is not in an optimal format for analysis because it contains HTML tags and other information that we are not interested in for analyzing the text at a later time.

### **[3] Import and Process all Data into a Relational Database**

In step 3, both the captioned transcript (.SRT) and metadata (.TXT) files will need to be imported into a structured SQL Server relational database. A Microsoft SQL Server Integration Services (SSIS) package will be developed to extract and load the data from the two files into the Microsoft SQL Server database "staging" tables. Data is "staged" in tables within the database so that it can be transformed at a later step in this research method.

### **[4] Perform all Additional Processing of Data and Preliminary Analysis**

Step 4 includes the additional steps added to the Microsoft SSIS package that will be used to transform and process the data from staging tables to the two final structured tables. The structured tables are designed for ease of research analysis. These two tables, named "final\_metadata" and "final\_transcripts\_coalesce", are described as follows:

The "final\_metadata" table includes all the fields required to store the YouTube video metadata. The field summary includes: ID (primary key column), video\_id (YouTube video ID), datetime\_retrieved (date timestamp the video was extracted from YouTube), search\_terms (YouTube search terms used to find the video), video\_time\_transcribed (length of the video in seconds), video title (title of the YouTube video), video category (YouTube category), subscribe (number of subscribers for the YouTube channel), views (number of times the video has been viewed), published (date video was published to YouTube), description (YouTube video description), and youtube\_



channel (YouTube Channel).

The “**final\_transcripts\_coalesce**” table includes all of the fields required to store the YouTube transcript text. The field summary includes: ID (primary key column), video\_id (YouTube video ID), and transcript (complete text from the transcript).

A preliminary analysis can be performed on the final transcript text and metadata tables to determine the value of the data. Once the researcher has evaluated the data, the data can be processed and exported as .TXT files into a targeted local directory for further analysis by R.

### **[5] Perform Topic Modeling and Additional Analysis**

Topic modeling is performed using the LDA algorithm in R. The LDA algorithm is a *generative probabilistic model of a corpus where documents are represented as random mixtures over latent topics*. Additionally, *each topic is characterized by a distribution over words* (Blei, Ng, & Jordan, Latent Dirichlet Allocation, 2003). The LDA algorithm assumes the following generative process for each document in a corpus (Chaney & Blei, 2012):

1. For  $K$  topics, choose each topic distribution  $\beta_k$  (Each  $\beta_k$  is a distribution over the vocabulary)
2. For each document in the collection:
  - a. Choose a distribution over topics  $\theta_d$  (The variable  $\theta_d$  is a distribution over  $K$  elements)
  - b. For each word in the document
    - i. Choose a topic assignment  $z_n$  from  $\theta_d$  (Each  $z_n$  is a number from 1 to  $K$ )
    - ii. Choose a word  $w_n$  from the topic distribution  $\beta_{z_n}$  (Notation  $\beta_{z_n}$  selects the  $z_n$ th topic from Step 1)

The R package `tm` (Feinerer & Hornik, 2017), a framework for text mining applications within R, will be used to generate the objects required to mine the text included in the documents. The R package `topicmodels` (Grun & Hornik, 2017) will be used to access and execute the LDA algorithm using the YouTube text data as input. The R package `reshape2` (Wickham, 2016) will be used to restructure and aggregate the data after the LDA algorithm has been executed and prior to visualization of the results. The R package `ggplot2` (Wickham & Chang, 2016) will be used to visualize the data for analysis.

## **Example Case: Exploring Topics in CIO Discussions of Cloud Adoption**

A researcher and/or information management analyst is trying answer the question: “*What topics can be discovered in YouTube transcripts that include CIO discussions about cloud adoption?*”. To answer this question, they will apply this research method with the goal of discovering valuable insights about a phenomenon of interest.

### **STEP 1: Screen the YouTube video content**

First, the YouTube search term “CIO cloud adoption” was used to locate the initial list of videos for screening. Then the researchers screened a sample of 50 YouTube videos based on the technique recommended by the research method. The videos were screened to validate that the discussion in the video included a CIO or was targeted for CIOs within the context of cloud adoption. See Appendix A.1 for the sample list of 50 videos.

### **STEP 2.1: Extract important video metadata associated with the captioned transcript**

The Selenium with Python script was executed against all YouTube video URLs included in the sample to derive a collection of .SRT files. These .SRT files will be transformed and imported into the MSSQL Server database later in STEP 3. See Appendix A.2 for a sample of the .SRT files collected.

### **STEP 2.2: Extract captioning transcript data**

Downsub.com was used to extract each of the video transcripts included in the sample of 50 YouTube URLs. These .TXT files will be transformed and imported into the MSSQL Server database later in STEP 3. See Appendix A.2 for a sample of the .TXT files collected.

### **STEP 3: Import and process all data into a relational database**

All .TXT and .SRT files are transformed and loaded into a SQL Server relational database staging tables using Microsoft SQL Server Integration Services (SSIS). Transact-SQL programming is used within the Microsoft SSIS package to perform the transformation and loading of the .TXT and .SRT files. See Appendix A.3 for a sample of the T-SQL code used in the SSIS package.

### **STEP 4: Perform all additional processing of data and preliminary analysis**

The Microsoft SSIS package transforms the data in the staging tables and loads it into the final tables for preliminary analysis. A preliminary analysis is per-

formed on these two tables to determine the value of the content. Once the researcher is satisfied with the preliminary results of the 50 YouTube video transcripts and metadata collected, the data is exported as .TXT files to a targeted directory. Each row of data that includes the text of the YouTube transcript will be exported as a single .TXT file. These .TXT files will then be used by R for further analysis. See Appendix A.4 for a sample of the T-SQL code written to export the processed transcript text data from the SQL Server database to a target directory.

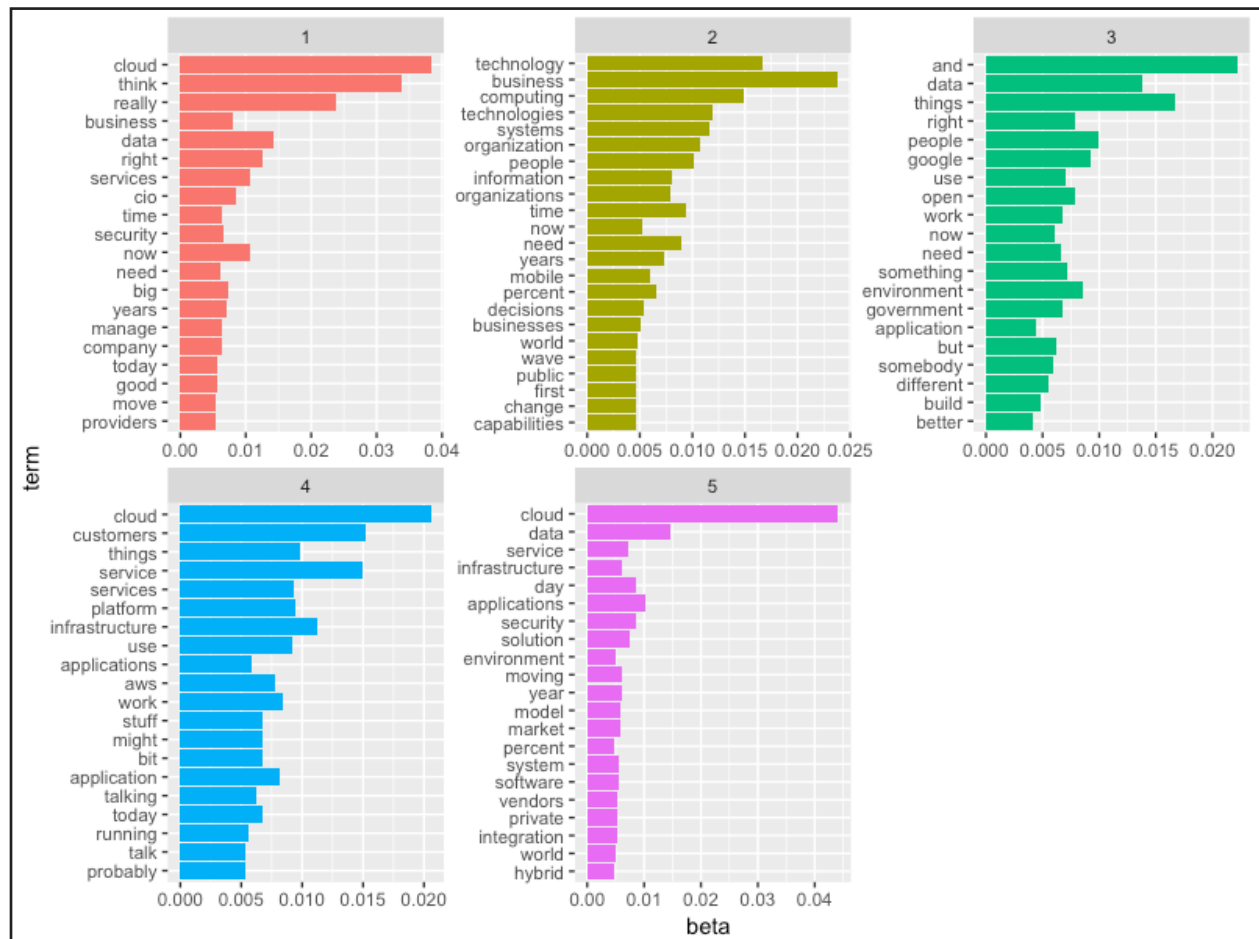
**STEP 5: Perform Topic Modeling and additional analysis**

The topicmodels R package will be used to execute the LDA model on the corpus (sample of 50 .TXT documents) and generate output. The following is the output of the LDA algorithm after it has generated 20 terms for each topic (see Figure 2).

Notice that each topic identified with a set of terms does not have a name (e.g., Topic 1, Topic 2, etc.).

	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
1	cloud	business	and	cloud	cloud
2	think	technology	things	customers	data
3	really	computing	data	service	applications
4	data	technologies	people	infrastructure	security
5	right	systems	google	things	day
6	services	organization	environment	platform	solution
7	now	people	right	services	service
8	cio	time	open	use	infrastructure
9	business	need	something	work	moving
10	big	information	use	application	year
11	years	organizations	work	aws	market
12	security	years	government	today	model
13	manage	percent	need	bit	software
14	time	mobile	but	might	system
15	company	decisions	now	stuff	integration
16	need	now	somebody	talking	private
17	today	businesses	different	applications	vendors
18	good	world	build	running	environment
19	move	change	application	probably	world
20	providers	capabilities	better	talk	percent

**Figure 2: This table shows the topics generated by the LDA algorithm. This model demonstrates a mixture of terms found within the documents with topics.**



**Figure 3: The above visualization was generated by the R package ggplot2. The visualization shows the strength of the top 20 terms for each topic. NOTE: these terms are not necessarily in the same order as seen in the table within Figure 2.**

However, when the words are collectively examined underneath each topic, they can appear to form a theme. For instance, Topic 1 in Figure 2 might represent a theme describing how CIOs manage cloud services. These latent themes can potentially tell a story that may reveal knowledge about CIOs adopting the cloud. This discovery process requires further analysis by the researcher to better understand the strength of each term in each latent topic. This can be accomplished using the `tidytext` and `ggplot2` R packages to build a visualization that illustrates the strength of the top 20 terms from each topic (see Figure 3). Visualizing the strength of the terms may help the researcher understand which terms have the highest level of influence for the specific topic.

Observing the visualization in Figure 3 reveals to the researcher that some topics have a stronger emphasis and concentration of terms than others. For instance, topics 2, 3, and 4 appear to have the strongest concentration of supported terms across the distribution of YouTube transcripts. This visualization requires the researcher to further analyze the strength of the topics as they have been generated from terms across the corpus of text documents.

Further analysis indicates the following probability strengths of the top 2 documents associated with each topic (see Figure 4):

Upon observation of the results seen in Figure 4, the researcher can go back to the MS SQL Server database and query the details of the transcript text and metadata for each of the top 2 strongest documents related to each of the topics. The output and visualizations seen by this research method in Figures 2, 3, and 4 offers substantial insight for any researcher or practitioner attempting to explore themes within the YouTube transcript data. The topic modeling package within R combined with the relational capabilities and structure of Microsoft SQL Server have demonstrated a practical approach to discovering latent topics within this example data set.

Topic	Document	Probability
Topic 1	b-7TiqyVCmE.txt	0.5621845
Topic 1	lDrIngHf_-Y.txt	0.5685019
Topic 2	6xpp1QhmtdY.txt	0.4426559
Topic 2	XE_LQhNYEdM.txt	0.6492711
Topic 3	mLiv0To99E4.txt	0.7034266
Topic 3	rXCBambBILs.txt	0.5260698
Topic 4	QcRFM8H9Drg.txt	0.5320479
Topic 4	WZvGHtt21hA.txt	0.6111111
Topic 5	FBDDa5pbwCs.txt	0.6018397
Topic 5	suNccoYCM84.txt	0.6870748

Figure 4: The top 2 probability strengths for each of the 5 topics.

## Example Case: Implications

Researchers could conclude from the case demonstrated by this research method that some parameters may need to be adjusted if they are not able to determine the significance of a topic generated by the LDA algorithm. In other words, the results of the topic modeling algorithm may include topics that have no sensible value toward the researcher's interest. The researcher may have to adjust the number of topics, words, or documents evaluated by the LDA algorithm in order to acquire a result that is worthwhile for interpretation. Upon adjustment of the parameters, the researcher would then repeat this example case with multiple iterations until an interpretable set of topics is acquired. Finally, studying the impact of adjusting parameters throughout this research method example case will add practical value to any practitioner who is considering the implementation of the LDA algorithm within their applications.

## Potential Applications

Researchers have attempted to address the problem of identifying emerging technologies through the analysis of text by data mining research proposals, publications (Cozzens, et al., 2010), and patent systems (Breitzman & Thomas, 2015). These research studies demonstrate that emergence of technology can be detected by analyzing the links between clustered structures of words or terms over slices of time. As the clusters observed across time slices begin to demonstrate an increase in quantitative measures, such as the number of associated papers or patents, the technology is then identified as emerging. There is a potential to extend the results of these research projects by using these data sources for application with this research method. This research method may offer a more robust framework for analyzing the data along with a more practical approach to exploring themes within research proposals, publications, and patents.

## Learn More

Tables 1 and 2 summarize recommended literature related to this research method.

## Conclusions

A research method has been developed that includes a framework with a collection of technologies familiar within industry by information management practitioners and researchers. This framework can be used to explore themes found within YouTube video transcript data. Microsoft technology is used in collaboration with Python and R throughout this research method framework. The set of technologies was selected for this research method due to their current and future integration by Microsoft prod-

**Table 1: Recommended literature related to this research method**

Source ID	Source	Type	Author(s)	Published
1	Text Mining with R, A Tidy Approach	Book	Julia Silge and David Robinson	5/7/17
2	topicmodels: An R Package for Fitting Topic Models	Article	Bettina Grun and Kurt Hornik	4/18/17
3	Package 'topicmodels', Reference Manual	Manual	Bettina Grun and Kurt Hornik	4/18/17
4	Introduction to the tm Package, Text Mining in R	Article	Ingo Feinerer	3/2/17
5	Package 'tm'	Manual	Ingo Feinerer, Kurt Hornik, Artifex Software, Inc.	3/2/17
6	SQL Server Integration Services	Documentation	Douglas Laudenschlager and Graig Guyer	3/14/17
7	Microsoft Machine Learning Services	Documentation	Jeannine Takaki, Cody Mansfield, Barbar Kess	4/18/17

**Table 2: Publishers and URL links to recommended literature related to this research method**

Source ID	Publisher	URL	URL Retrieved
1	O'Reilly Press	<a href="http://tidytextmining.com">http://tidytextmining.com</a>	5/9/17
2	The Comprehensive R Archive Network	<a href="https://cran.r-project.org/web/packages/topicmodels/vignettes/topicmodels.pdf">https://cran.r-project.org/web/packages/topicmodels/vignettes/topicmodels.pdf</a>	5/9/17
3	The Comprehensive R Archive Network	<a href="https://cran.r-project.org/web/packages/topicmodels/topicmodels.pdf">https://cran.r-project.org/web/packages/topicmodels/topicmodels.pdf</a>	5/9/17
4	The Comprehensive R Archive Network	<a href="https://cran.r-project.org/web/packages/tm/vignettes/tm.pdf">https://cran.r-project.org/web/packages/tm/vignettes/tm.pdf</a>	5/9/17
5	The Comprehensive R Archive Network	<a href="https://cran.r-project.org/web/packages/tm/tm.pdf">https://cran.r-project.org/web/packages/tm/tm.pdf</a>	5/9/17
6	Microsoft	<a href="https://docs.microsoft.com/en-us/sql/integration-services/sql-server-integration-services">https://docs.microsoft.com/en-us/sql/integration-services/sql-server-integration-services</a>	5/9/17
7	Microsoft	<a href="https://docs.microsoft.com/en-us/sql/advanced-analytics/r/r-services">https://docs.microsoft.com/en-us/sql/advanced-analytics/r/r-services</a>	5/9/17

ucts. This tight integration allows a practicing information management firm or researcher to build and deploy integrated solutions for this research method in a production enterprise environment.

An example case has been presented to demonstrate the application of this research method using topic modeling with the LDA algorithm. This example case takes the researcher through the entire research method protocol for better understanding of its steps and applicability to answer a targeted research and/or business question. At the conclusion of this example case, the researcher can answer the targeted

question with a structured and measurable method for analysis. The results of this research method have provided output in the form of topic model generation (Figure 2), visualizations (Figure 3), and probability distributions (Figure 4). Together, these three outputs from the LDA algorithm can enable a researcher or information management practitioner to discover new knowledge in the form of a theme or decide to continue the thematic exploration process with more iterations of this research method.

Finally, this research method has the potential for other applications within industry and research. Pri-



or research studies have demonstrated the value of discovering clusters of information by data mining the text within data sources such as research proposals, publications, and patents. This research method can be used to contribute or extend the work of these prior studies by applying a novel approach that allows for the discovery of themes within these data sources.

## References

- Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM*, 55(4), 77-84.
- Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan), 993-1022.
- Breitzman, A., & Thomas, P. (2015). The Emerging Clusters Model: A tool for identifying emerging technologies across multiple patent systems. *Research policy*, 44(1), 195-205.
- Chaney, A. J., & Blei, D. M. (2012, June 8). *Visualizing Topic Models*. Retrieved from Computer Science at Columbia University: <http://www.cs.columbia.edu/~blei/papers/ChaneyBlei2012.pdf>
- Cozzens, S., Gatchair, S., Kang, J., Kim, K. S., Lee, H. J., Ordóñez, G., & Porter, A. (2010). Emerging technologies: quantitative identification and measurement. *Technology Analysis & Strategic Management*, 22(3), 361-376.
- Feinerer, I., & Hornik, K. (2017, March 2). *tm: Text Mining Package*. Retrieved from tm: Text Mining Package: <https://cran.r-project.org/web/packages/tm/index.html>
- Ganesan, A., Brantley, K., Pan, S., & Chen, J. (2015, July 25). *LDAExplore: Visualizing topic models generated using latent Dirichlet allocation*. Retrieved from Cornell University Library: <https://arxiv.org/pdf/1507.06593.pdf>
- Grun, B., & Hornik, K. (2017, April 18). *Topic models: An R package for fitting topic models*. Retrieved from The Comprehensive R Archive Network: <https://cran.r-project.org/web/packages/topic-models/vignettes/topicmodels.pdf>
- Kaver, L. (2017, February 16). *YouTube official blog: One billion captioned videos*. Retrieved from YouTube Official Blog: <https://youtube.googleblog.com/2017/02/one-billion-captioned-videos.html>
- Muthukadan, B. (n.d.). *Selenium with Python*. Retrieved from Selenium with Python: <http://selenium-python.readthedocs.io>
- Python. (n.d.). Retrieved from Python: <https://www.python.org>
- Rana, T. A., Cheah, Y. N., & Letchmunan, S. (2016). Topic modeling in sentiment analysis: A systematic review. *Journal of ICT Research & Applications*, 10(1), 76-93.
- Roberts, M. E., Stewart, B. M., & Airoidi, E. M. (2016). A model of text for experimentation in the social sciences. *Journal of American Statistical Association*, 111(515), 988-1003.
- Takaki, J., & Kess, B. (2017, July 31). *What's new in Machine Learning Services in SQL Server*. Retrieved from Microsoft Advanced Analytics on SQL Server: <https://docs.microsoft.com/en-us/sql/advanced-analytics/what-s-new-in-sql-server-machine-learning-services>
- The R Project for Statistical Computing*. (n.d.). Retrieved from The R Project for Statistical Computing: <https://www.r-project.org>
- Wang, C., & Blei, D. M. (2011, August). Collaborative topic modeling for recommending scientific articles. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 448-456). ACM.
- What is Selenium?* (n.d.). Retrieved from Selenium-HQ Browser Automation: <http://www.seleniumhq.org>
- Wickham, H. (2016, October 22). *reshape2: Flexibly Reshape Data: A Reboot of the Reshape Package. R package version 1.4. 2*. Retrieved from reshape2: Flexibly Reshape Data: A Reboot of the Reshape Package: <https://cran.r-project.org/web/packages/reshape2/index.html>
- Wickham, H., & Chang, W. (2016). ggplot2: Create elegant data visualisations using the grammar of graphics. *R Development Core Team. R package version, 2(1)*. Retrieved from ggplot2: Create Elegant Data Visualisations Using the Grammar of Graphics: <https://cran.r-project.org/web/packages/ggplot2/index.html>

## Review

This article was accepted under the **constructive peer review** option. For further details, see the descriptions at:

<http://mumabusinessreview.org/peer-review-options/>

## Author



































**Clinton Daniel** is an Instructor in the Information Systems Decision Sciences Department at the USF Muma College of Business in Tampa, FL, teaching courses in business intelligence, business data communications, business application development and systems analysis and design. Before joining USF's faculty in 2014, Daniel worked in several information technology roles for the United States Department of Veterans Affairs, where he managed a data warehouse for seven VA medical centers and developed business intelligence solutions for the VA and its customers. Daniel earned a Master of Science in Management Information Systems from USF and a Bachelor's degree from Saint Leo University in Saint Leo, FL.

## Appendix

### A.1: Sample list of 50 YouTube URLs after searching “CIO cloud adoption” and visual/audio screening

[https://www.youtube.com/watch?v=s7y\\_qv-g85s](https://www.youtube.com/watch?v=s7y_qv-g85s)  
<https://www.youtube.com/watch?v=ZzEMLem3sKI>  
[https://www.youtube.com/watch?v=lDr1ngHf\\_-Y](https://www.youtube.com/watch?v=lDr1ngHf_-Y)  
[https://www.youtube.com/watch?v=XE\\_LQhNYEdM](https://www.youtube.com/watch?v=XE_LQhNYEdM)  
<https://www.youtube.com/watch?v=b-7TiqyVCmE>  
<https://www.youtube.com/watch?v=g2UtKieUNyY>  
<https://www.youtube.com/watch?v=-yMmIenKvAQ>  
[https://www.youtube.com/watch?v=Pk\\_dVTYQe00](https://www.youtube.com/watch?v=Pk_dVTYQe00)  
<https://www.youtube.com/watch?v=BgNQ9oEF12c>  
<https://www.youtube.com/watch?v=-rw655rZM-0>  
<https://www.youtube.com/watch?v=6xpp1QhmtdY>  
<https://www.youtube.com/watch?v=ZRDXEtwbndg>  
<https://www.youtube.com/watch?v=qunt2S7Kn-U>  
[https://www.youtube.com/watch?v=pr7xHII\\_yn8](https://www.youtube.com/watch?v=pr7xHII_yn8)  
[https://www.youtube.com/watch?v=xiP1bA\\_ro5U](https://www.youtube.com/watch?v=xiP1bA_ro5U)  
<https://www.youtube.com/watch?v=WDpwb2LZ9Zs>  
<https://www.youtube.com/watch?v=mLiv0To99E4>  
<https://www.youtube.com/watch?v=9ZHwePUCNbo>  
<https://www.youtube.com/watch?v=4xZ6puzwuWE>  
<https://www.youtube.com/watch?v=suNccoYCM84>  
<https://www.youtube.com/watch?v=dbVMkxZ8yew>  
<https://www.youtube.com/watch?v=i1yW6vWCpgk>  
<https://www.youtube.com/watch?v=xn3O13NwQc4>  
<https://www.youtube.com/watch?v=prHwxxxAKGk>  
<https://www.youtube.com/watch?v=C3J3jjwDDoY>  
<https://www.youtube.com/watch?v=JZzGg0ir3N8>  
<https://www.youtube.com/watch?v=Y4TDbvdWISM>  
<https://www.youtube.com/watch?v=FBDDa5pbwCs>  
<https://www.youtube.com/watch?v=QcRFM8H9Drg>  
<https://www.youtube.com/watch?v=XGL-MmwJT1Q>  
<https://www.youtube.com/watch?v=ARYsEWIQYQs>  
[https://www.youtube.com/watch?v=e7\\_rkVNYNJE](https://www.youtube.com/watch?v=e7_rkVNYNJE)  
<https://www.youtube.com/watch?v=XXT1QgNE3Ks>  
<https://www.youtube.com/watch?v=KBjxpta9Ci8>  
<https://www.youtube.com/watch?v=K4AIS1GeHYs>  
<https://www.youtube.com/watch?v=1ZFmSMqKDq4>  
<https://www.youtube.com/watch?v=SsyJHup48qA>  
<https://www.youtube.com/watch?v=0E90-ExySb8>  
<https://www.youtube.com/watch?v=43yJe9sukZw>  
<https://www.youtube.com/watch?v=rXCBambBLLs>  
<https://www.youtube.com/watch?v=VBVR53D-ojI>  
[https://www.youtube.com/watch?v=UvG\\_sjnRY0g](https://www.youtube.com/watch?v=UvG_sjnRY0g)  
<https://www.youtube.com/watch?v=SV7CHyZCiKo>  
[https://www.youtube.com/watch?v=PEBSXC\\_fYYg](https://www.youtube.com/watch?v=PEBSXC_fYYg)  
<https://www.youtube.com/watch?v=Whbed3dAxiU>  
<https://www.youtube.com/watch?v=fmN8bZWUwig>  
<https://www.youtube.com/watch?v=5WpornYIHkU>  
<https://www.youtube.com/watch?v=IsTYdhHuEB8>  
<https://www.youtube.com/watch?v=WZvGHtt21hA>  
<https://www.youtube.com/watch?v=WNUq66W8Y3U>

**A.2: Sample of .SRT and .TXT files from the 50 each collected based on the YouTube URLs**

Name	Date Modified	Size	Kind
 WNUq66W8Y3U.txt	May 2, 2017, 12:31 AM	3 KB	Plain Text Document
 WNUq66W8Y3U.srt	May 2, 2017, 12:30 AM	53 KB	Subrip Subtitle File
 lsTYdhHuEB8.txt	May 2, 2017, 12:21 AM	3 KB	Plain Text Document
 lsTYdhHuEB8.srt	May 2, 2017, 12:20 AM	42 KB	Subrip Subtitle File
 4xZ6puzwuWE.txt	May 1, 2017, 11:32 PM	3 KB	Plain Text Document
 4xZ6puzwuWE.srt	May 1, 2017, 11:31 PM	3 KB	Subrip Subtitle File
 UvG_sjnRYOg.txt	May 1, 2017, 10:49 PM	2 KB	Plain Text Document
 UvG_sjnRYOg.srt	May 1, 2017, 10:48 PM	5 KB	Subrip Subtitle File
 Whbed3dAxiU.txt	May 1, 2017, 10:38 PM	3 KB	Plain Text Document
 Whbed3dAxiU.srt	May 1, 2017, 10:37 PM	20 KB	Subrip Subtitle File
 fmN8bZWUwig.txt	May 1, 2017, 9:42 PM	2 KB	Plain Text Document
 fmN8bZWUwig.srt	May 1, 2017, 9:42 PM	11 KB	Subrip Subtitle File
 5WpornYIHkU.txt	May 1, 2017, 9:21 PM	3 KB	Plain Text Document
 5WpornYIHkU.srt	May 1, 2017, 9:20 PM	13 KB	Subrip Subtitle File
 SsyJHup48qA.txt	May 1, 2017, 9:12 PM	3 KB	Plain Text Document
 SsyJHup48qA.srt	May 1, 2017, 9:11 PM	4 KB	Subrip Subtitle File
 SV7CHyZCiKo.txt	May 1, 2017, 9:02 PM	2 KB	Plain Text Document
 SV7CHyZCiKo.srt	May 1, 2017, 8:59 PM	5 KB	Subrip Subtitle File
 QcRFM8H9Drg.txt	May 1, 2017, 7:20 PM	3 KB	Plain Text Document
 QcRFM8H9Drg.srt	May 1, 2017, 7:19 PM	69 KB	Subrip Subtitle File
 XGL-MmwJT1Q.txt	May 1, 2017, 6:48 PM	2 KB	Plain Text Document
 XGL-MmwJT1Q.srt	May 1, 2017, 6:46 PM	5 KB	Subrip Subtitle File
 e7_rkVNYNJE.txt	May 1, 2017, 6:26 PM	3 KB	Plain Text Document
 e7_rkVNYNJE.srt	May 1, 2017, 6:25 PM	8 KB	Subrip Subtitle File
 1ZFmSMqKDq4.txt	May 1, 2017, 6:08 PM	2 KB	Plain Text Document
 1ZFmSMqKDq4.srt	May 1, 2017, 6:07 PM	20 KB	Subrip Subtitle File
 PEBSXC_fYYg.txt	May 1, 2017, 5:32 PM	3 KB	Plain Text Document
 PEBSXC_fYYg.srt	May 1, 2017, 5:31 PM	8 KB	Subrip Subtitle File
 rXCBambBILs.txt	May 1, 2017, 4:55 PM	4 KB	Plain Text Document
 rXCBambBILs.srt	May 1, 2017, 4:54 PM	92 KB	Subrip Subtitle File
 OE90-ExySb8.txt	May 1, 2017, 4:33 PM	3 KB	Plain Text Document
 OE90-ExySb8.srt	May 1, 2017, 4:30 PM	14 KB	Subrip Subtitle File



### A.3: Sample Transact-Structure Query Language (T-SQL) code

Used in the Microsoft SQL Server Integration Services (SSIS) to transform and load the YouTube transcript text extracted from the .TXT files into a MS SQL Server database table called youtube.metadata transformed. NOTE: There are more steps that include SQL code in the SSIS package. However, this code is valuable, so that it is understood how the transformation and loading process is occurring from a .TXT file.

```

/*
#####
INSERT Transformed Metadata into Staging table
#####
*/
INSERT INTO youtube.metadata_transformed
(video_id,video_url,datetime_retrieved,search_terms,video_time_transcribed,video_title,youtube_category,sub-
scribe,views,published,description,youtube_channel)
(
SELECT
-- Get video_id
DISTINCT (SELECT metadata FROM youtube.metadata_staging WHERE ID = 1) AS video_id,
-- Get video_url
(SELECT metadata FROM youtube.metadata_staging WHERE ID = 2) AS video_url,
-- Get datetime_retrieved
CAST((SELECT metadata FROM youtube.metadata_staging WHERE ID = 3) AS datetime) AS datetime_retrieved,
-- Get search_terms
(SELECT metadata FROM youtube.metadata_staging WHERE ID = 4) AS search_terms,
-- Get video_time_transcribed; rarely, but occasionally there will be a video > 1 hour. Manual applica-
tion of variable below this line if needed.
(SELECT (CAST(LEFT(SUBSTRING(metadata,8,6), CHARINDEX(':',SUBSTRING(metadata,8,6),0)-1) AS int) * 60 +
CAST(RIGHT(SUBSTRING(metadata,8,6), LEN(SUBSTRING(metadata,8,6)) - CHARINDEX(':',SUBSTRING(metadata,8,6),0)) AS
int)) FROM youtube.metadata_staging WHERE metadata LIKE @video_time_transcribed) AS video_time_transcribed,
-- Get video_title
(SELECT metadata FROM youtube.metadata_staging WHERE ID = @video_title_id) AS video_title,
-- Get youtube_category
(SELECT @youtube_category) AS youtube_category,
-- Get subscribe
(SELECT @subscribe) AS subscribe,
-- Get views
(SELECT @views) AS views,
-- Get published
(SELECT CAST(SUBSTRING(metadata,14,15) AS date) FROM youtube.metadata_staging WHERE metadata LIKE 'Pub-
lished on %') AS published,
-- Get description
(SELECT @AllData) AS description,
-- Get youtube channel
(SELECT metadata FROM youtube.metadata_staging WHERE ID = @youtube_channel) AS youtube_channel

FROM youtube.metadata_staging
)

```

### A.3 (Continued)

Sample T-SQL code used in the Microsoft SSIS Package to transform and load the YouTube metadata extracted from the.SRT files in a MS SQL Server database table called youtube.raw\_data\_staging. NOTE: There are more steps that include SQL code in the SSIS package. However, this code is valuable so that it is understood how the transformation and loading process is occurring from a.SRT file.

```
UPDATE youtube.raw_data_staging
-- This update will replace several of the tags that were left in the srt file
-- The removal these tags will make it easier to perform the text analysis
SET raw_transformed =
    REPLACE(REPLACE(REPLACE(REPLACE(raw_data,'</font>',''), '<font color="#E5E5E5">',''), '<font color="#CCCCCC">',''), '<font color="#FFFFFF">','')

SELECT raw_transformed FROM youtube.raw_data_staging
WHERE raw_transformed !=''
AND raw_transformed NOT LIKE '%00:%'
AND raw_transformed LIKE '%[^0-9]%'
```

## A.4: Code used to export all of the YouTube transcript text data from a SQL Server database table

This includes the complete Microsoft T-SQL code used to export all of the YouTube transcript text data from a SQL Server database table called youtube.final\_transcripts\_coalesce. The T-SQL code uses a LOOP to generate each of the individual .TXT files for form a collection (corpus) in a targeted directory called “C:\test”.

```
-- Setup variables that will be used to generate text files
DECLARE @OutputFile NVARCHAR(100),
@FilePath NVARCHAR(100),
@bcpCommand NVARCHAR(1000)
DECLARE @GetMinID int;
DECLARE @GetMaxID int;
DECLARE @Video_id varchar(50);
/*
#####
This query will need to be modified each time to generate
the corpus of text files
*/
SELECT final_trco.id
INTO #Get_ids
FROM youtube.final_transcripts_coalesce AS final_trco
INNER JOIN youtube.final_metadata AS final_md ON final_trco.video_id = final_md.video_id
ORDER BY final_trco.id
/*
#####
*/
-- Populate Set_ids temp table with Ids from Get_ids temp table
SELECT final_trco2.id
INTO #Set_ids
FROM youtube.final_transcripts_coalesce AS final_trco2
WHERE final_trco2.id IN(SELECT #Get_ids.id FROM #Get_ids)

-- Get the MIN ID from the Set_ids temp table
SET @GetMinID = (SELECT MIN(#Set_ids.id) FROM #Set_ids)
-- Get the MAX ID from the Set_ids temp table
SET @GetMaxID = (SELECT MAX(#Set_ids.id) FROM #Set_ids)

-- Execute WHILE loop to generate all text files
WHILE @GetMinID <= @GetMaxID
BEGIN
    SET @Video_id = (SELECT video_id FROM youtube.final_transcripts_coalesce WHERE ID = @GetMaxID)
    -- Generate the text file
    SET @bcpCommand = 'bcp "USE youtube; SELECT transcript FROM youtube.final_transcripts_coalesce WHERE ID
= ' + CAST(@GetMaxID AS nvarchar(100)) + ', "queryout '
    SET @FilePath = 'C:\test\'
    SET @OutputFile = @Video_id + '.txt'
    SET @bcpCommand = @bcpCommand + @FilePath + @OutputFile + ' -c -t, -T -S'+ @@servername
    exec master..xp_cmdshell @bcpCommand

    -- Get rid of ID that has already been generated as a text file
    DELETE FROM #Set_ids WHERE #Set_ids.id = @GetMaxID
    SET @GetMinID = (SELECT MIN(#Set_ids.id) FROM #Set_ids)
    SET @GetMaxID = (SELECT MAX(#Set_ids.id) FROM #Set_ids)
END

-- Drop all temp tables
DROP TABLE #Get_ids
DROP TABLE #Set_ids
```