ANALYTIXLABS

Text Mining (Analytics)

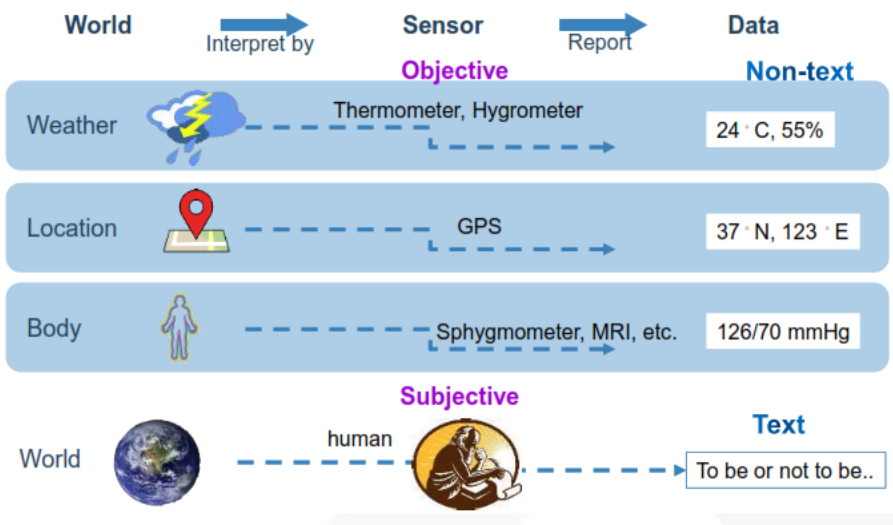
Agenda

- ✓What is Text Mining/Text Analytics?
- ✓Major Areas of Text Analytics
- ✓Text Mining Framework & process
- ✓Detailed Steps in Text Mining
- ✓Text Data processing
- ✓Concept of Natural Language Processing
- ✓Text Visualization
- ✓Role of Machine Learning in Text Analytics
- ✓Text mining Use cases

Text Mining (Analytics)

ANALYTIXLABS

Types of Data (Text vs. Non- Text)



ANALYTIXLABS

Types of Data

Structured Data

- Loadable into a spreadsheet
  - Rows & columns
  - Each cell filled, or could be filled
  - Data is consistent, uniform
- Data Mining Friendly

Un-Structured Data

- Not Structured into 'Cells'
  - Variable record length; notes, free-form survey answers
  - Text is relatively sparse, inconsistent, and not uniform
  - Also… Images, video, music, etc.

Types of Un-Structured Data

- **Weakly Structured data:** few structural cues to text based on layout or markups
  - Research papers
  - Legal memoranda
  - News stories
- **Semi structured data:** extensive format elements, metadata, field labels
  - EMAIL
  - HTML Web pages
  - Pdf files/ XML web pages /JSON Data
  - Log files

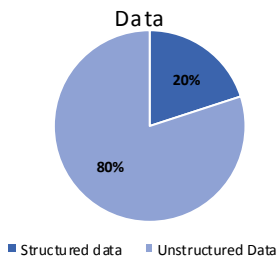
ANALYTIXLABS

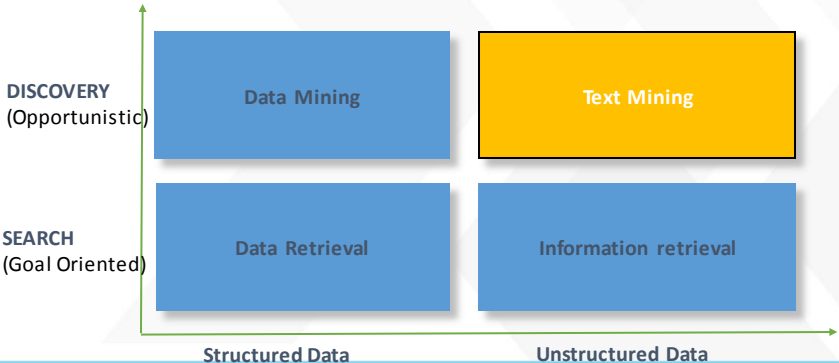
What is text mining?

Methods and tools allowing the automatic analysis of the textual information entering the enterprise

Text Mining is the discovery by computer of new, previously unknown information, by automatically extracting information from textual data (semi-structured and unstructured data).

Text mining starts by extracting key points, opinions, people, actions, events from textual sources thus enabling forming new hypotheses that are further explored by traditional BI and Data Mining methods





ANALYTIXLABS

What Text Mining is NOT..

- It is not Google!
- Text mining is not based upon understanding of document content.
- Instead it predicts the most likely meaning of a fragment of text based upon the language models.
- Text mining will generally not pick up on sarcasm, irony or other subtleties of language usage.
- Text mining tools must be tuned before use on different text types, styles or languages.

Sources of Data for Text Mining?

Sources are highly varied –

- Web sites, news & journal articles, images, video.
- Blogs, forum postings, and social media.
- E-mail, Contact-center notes and transcripts; recorded conversation.
- Surveys, feedback forms, warranty & insurance claims.
- Office documents, regulatory filings, reports, scientific papers

Data Analytics Vs. Text Analytics

Data Analytics	Text Analytics

Customer analytics
• Profiling and segmentation of customers
• Customer retention
• Profitability analysis
 	
Operational analytics
• Optimization
  • Capacity & performance
  • Workload characterization
  • Change detection
• Scheduling and optimization
 	
Financial & Risk analytics
• Financial & sales forecast
• Pricing
• Credit risk - default prediction
 	
Fraud detection and prevention
• Discover anomalous behavior
• Model various types of frauds
	Analysis of free text in customer surveys
	• Automate customer clustering/segmentation
	• Sentiment Analysis / Feedback Analysis
	• Derive insights about customers
	
	Analysis of call center transcripts
	• Improve productivity in the call centre
	• Call center performance
	• Contextual feedback on customer experience
	
	Analysis of forums, blogs and comments
	• Understand true voice of customer
	• Understand social networks
	
	Analysis of free text within the Enterprise
	• Auto-summarization of documents, reports for exec level knowledge sharing
	• Reduce costs and enhance speed of knowledge publishing

Text mining techniques: IE, Keyword extraction, Clustering, Summarization using lexical chain

ANALYTIXLABS

Data Analytics Vs. Text Analytics

Non-text data
- Numerical
  Categorical
  Relational
- Precise
- Objective

Text data
- Text
- Ambiguous
- Subjective

Text Mining

Data Mining
- Clustering
- Classification
- Association Rules
- …

Text Processing (including NLP)

Preprocessing

ANALYTIXLABS

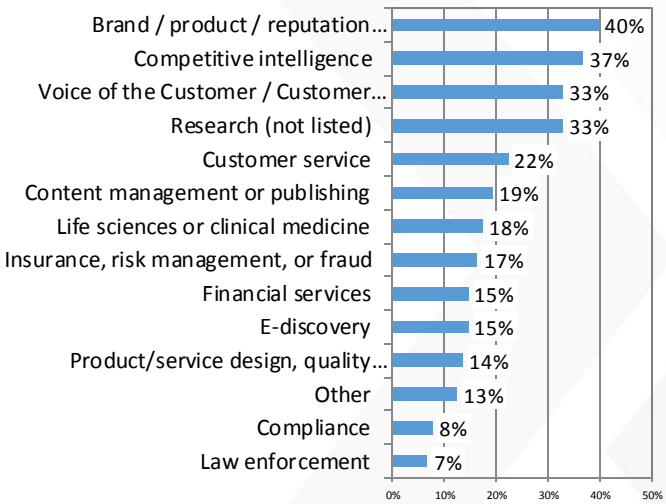
What does Text Mining bring to business?

The analysis of text information used for several general purposes:

- **Business:** Competitive business intelligence, document categorization, Voice of the employee (HR), records retention, risk analysis, website navigation
- **CRM Analytics:** Voice of the customer, Product & Service gap analysis, Churn analysis
- **Social Media Analytics:** Purchase Intent, Customer Churn Prediction, Reputational Risk
- **Marketing:** Survey analysis, market research
- **Regulatory Compliance:** Data redaction to identify and protect sensitive information
- **Log Analytics:** Failure analysis and root cause identification, Availability assurance
- **Digital Piracy:** Illegal broadcast of streaming and video content
- **Other Analytics: Fraud detection, e-discovery, warranty analysis, medical research**
- **Generate summaries and categorize documents in the most efficient way**
  - e.g., news, emails, call center/helpdesk inquiries
- **Identify hidden patterns between documents or groups of documents**
  - e.g., customer complaints, warranty claims, free form survey data
- **Increase the efficiency and effectiveness of a search process to find similar information**
- **Analyze textual information with other structured information to build models**
  - e.g., predict customer satisfaction, claim fraud, drug efficacy

ANALYTIXLABS

What are your primary applications where text comes into play?



ANALYTIXLABS

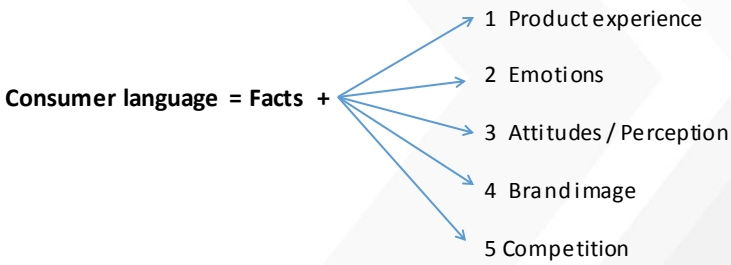
Text Mining in Banking & Financial Services

- **Identifying At-Risk Customers:** With Text Mining, you can identify those customers who are at risk of withdrawing. Identification can be done based on these users' interactions with your customer service (via email, messenger conversations, or call transcripts), forum posts, posts on social media, among other routes
- **Preempting Customer Loss through Complaint Analysis:** Text Mining makes it possible to identify customer complaints in your incoming mail and prioritizes them so that accounts in jeopardy can be addressed by your employees without delay
- **Enhancing Your Credit Scoring Models: T**ext Mining can be incorporated into your credit scoring models as a set of processing steps. These steps will detect words or phrases that you designate meaningful in your decision-making process.
- **Refining Your Lending Process and Making It More Reliable**: If your credit risk officers are precluded from making a lending decision by a lack of information on a customer, with Text Mining they will be able to probe the Web for any particulars that they deem serviceable. Results include financial reports, press releases, customer reviews, and information on the customer's Executive Board with related changes.
- **Enhancing Your Bank's Marketing Scoring Models: M**ost banks use scoring models to target their marketing campaigns based on demographics & transactional data. With Text Mining, your existing scoring models can be further enhanced to include customer-specific insights derived from your unstructured data. You might also use unstructured data from your customer's posts on social networks. In a financial market with increasing competition, your institution can manage customer churn with NLP. Text mining will enrich your organization with the resources to maintain and enhance your market share.

ANALYTIXLABS

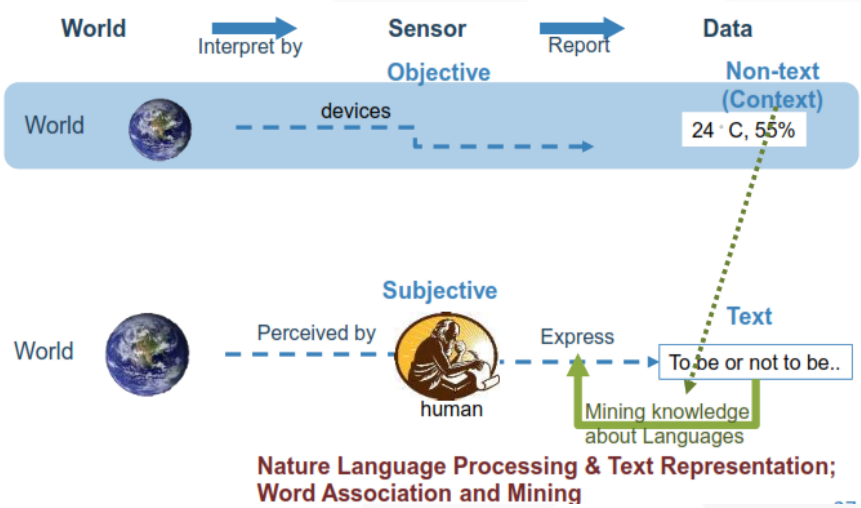
Text mining for Analyzing customer

To help discover the true value of information!



ANALYTIXLABS

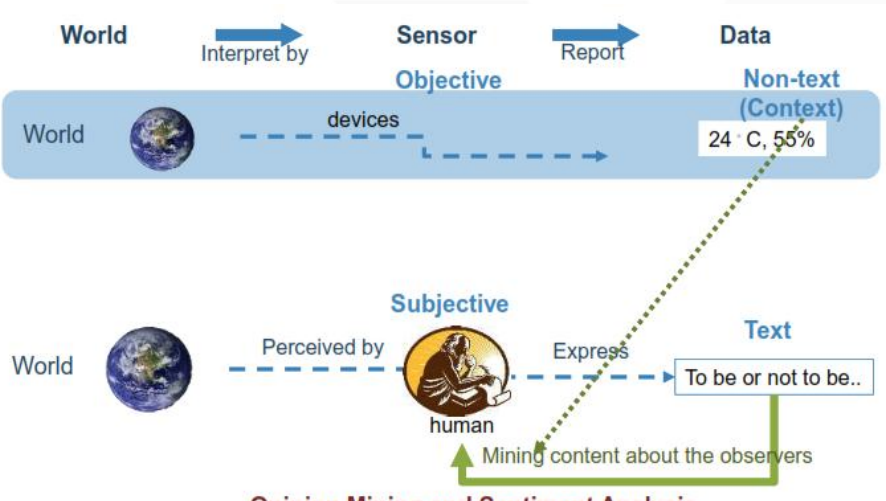
Landscape of Text Mining



Nature Language Processing & Text Representation; Word Association and Mining

ANALYTIXLABS

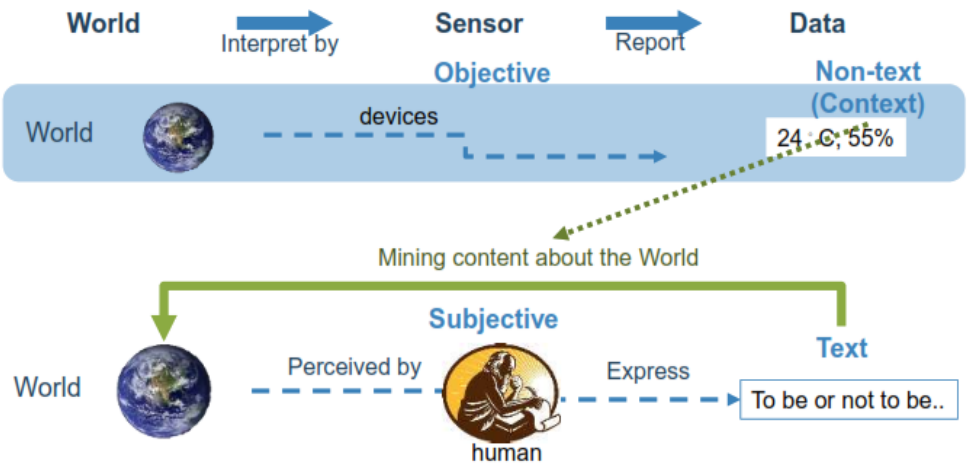
Landscape of Text Mining



Opinion Mining and Sentiment Analysis

ANALYTIXLABS

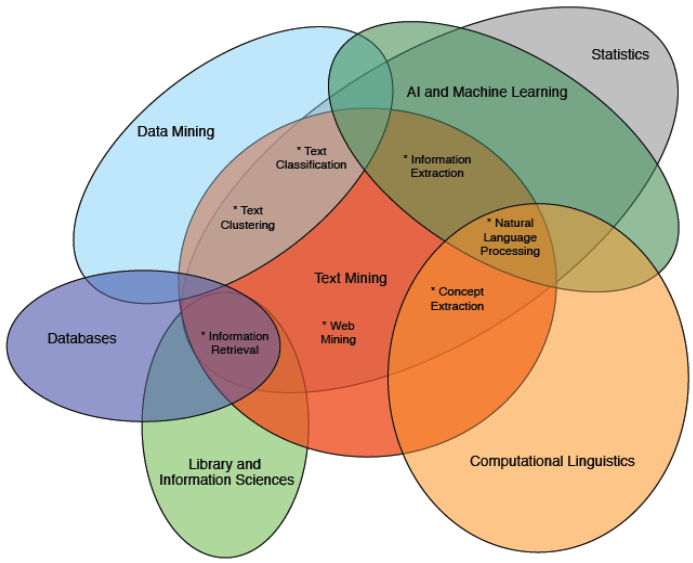
Landscape of Text Mining



Topic Mining , Contextual Text Mining

ANALYTIXLABS

Key Areas of Text Mining



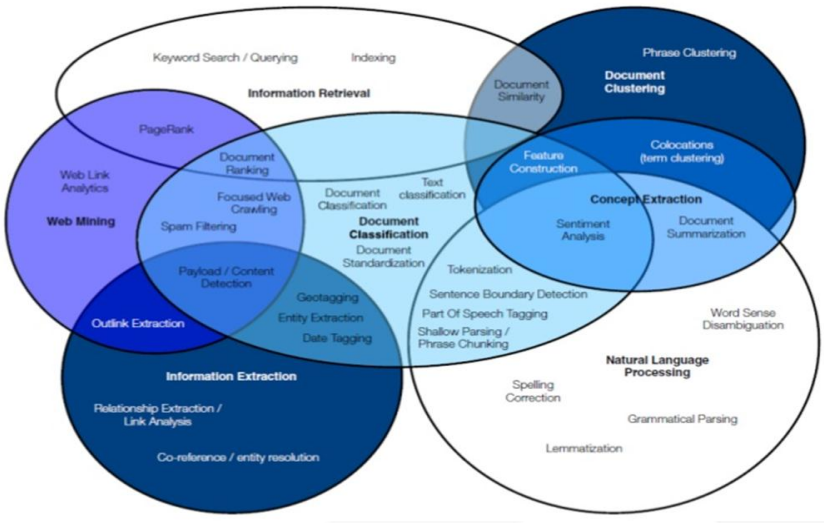
ANALYTIXLABS

Text Mining - Application areas

- **Search and Information Retrieval (IR):** Storage and retrieval of text documents, including search engines and keyword search
- **Document Clustering:** Grouping and categorizing terms, snippets, paragraphs or documents using data mining clustering methods
- **Text Categorization/Document classification:** Grouping and categorizing snippets, paragraphs, or documents using data mining classification methods, trained or labelled examples
- **Web Mining:** Data and text mining on the internet, with specific focus on the scale and interconnectedness of the web
- **Information Extraction (IE):** Identification and extraction of relevant facts and relationships from unstructured and semi structured text
- **Natural Language process (NLP):** Low-level language processing and understanding tasks (Ex: Tagging parts of speech); often used synonymously with computations linguistics
- **Concept Extraction:** Grouping of words or phrases into semantically similar groups
- **Association between Terms/Word clustering:** Discovering associations between terms
- **Text Summarization:** Summarizing large amount of textual and factual data.

ANALYTIXLABS

Major Areas of Text Analytics



ANALYTIXLABS

Text Mining in Telecom-E Commerce

- Text analytics is applied for marketing, search optimization, competitive intelligence.
  - Analyze social media and enterprise feedback to understand the Voice of the Market:
    - Opportunities
    - Threats
    - Trends
  - Categorize product and service offerings for on-site search and faceted navigation and to enrich content delivery.
  - Annotate pages to enhance Web-search findability, ranking.
  - Scrape competitor sites for offers and pricing.
  - Analyze social and news media for competitive information.

Text Mining in Telecom-E-Commerce: E-Discovery & Compliance

- Text analytics is applied for compliance, fraud and risk, and e-discovery.
  - Regulatory mandates and corporate practices dictate–
    - Monitoring corporate communications
    - Managing electronic stored information for production in event of litigation
  - Sources include e-mail ,news, social media
  - Risk avoidance and fraud detection are key to effective decision making
    - Text analytics mines critical data from unstructured sources
    - Integrated text-transactional analytics provides rich insights

Text Mining in Telecom-Online Voice of the Customer

- Text analytics is applied to enhance customer service and satisfaction.
  - Analyze customer interactions and opinions –
    - E-mail, contact-center notes, survey responses
    - Forum & blog posting and other social media

      … to ..
    - Address customer product & service issues
    - Improve quality
    - Manage brand & reputation
  - If qualitative information from text can be linked , the following become possible–
    - Link feedback to transactions
    - Assess customer value
    - Understand root causes
    - Mine data for measures such as churn likelihood

ANALYTIXLABS

Application in Insurance Domain – Improve CRM, Product

- Customers call into call centers / send e-mails / express opinions in surveys or blogs that can indicate their likes / dislikes or sentiments. Customer contact staff can recognize when a customer is at risk of leaving and take appropriate action to **reduce churn** and to **increase satisfaction level**
- Customer calling in to ask about their insurance policy and rep types in or records what the person is saying, and it could prompt a call center person to take an action, such as offering the person a certain price special at that time thus presenting an opportunity for **cross-sell/up-sell**
- Insurers can monitor the quality and effectiveness of the reps taking the phone calls by analyzing hand-written / typed notes, voice files converted to text and **improve** their **productivity**
- Analyzing unstructured data from data inside and outside an enterprise can give insights for strategic planning, **developing new products**
- Can help in achieving a clearer view of the **competitive landscape**

ANALYTIXLABS

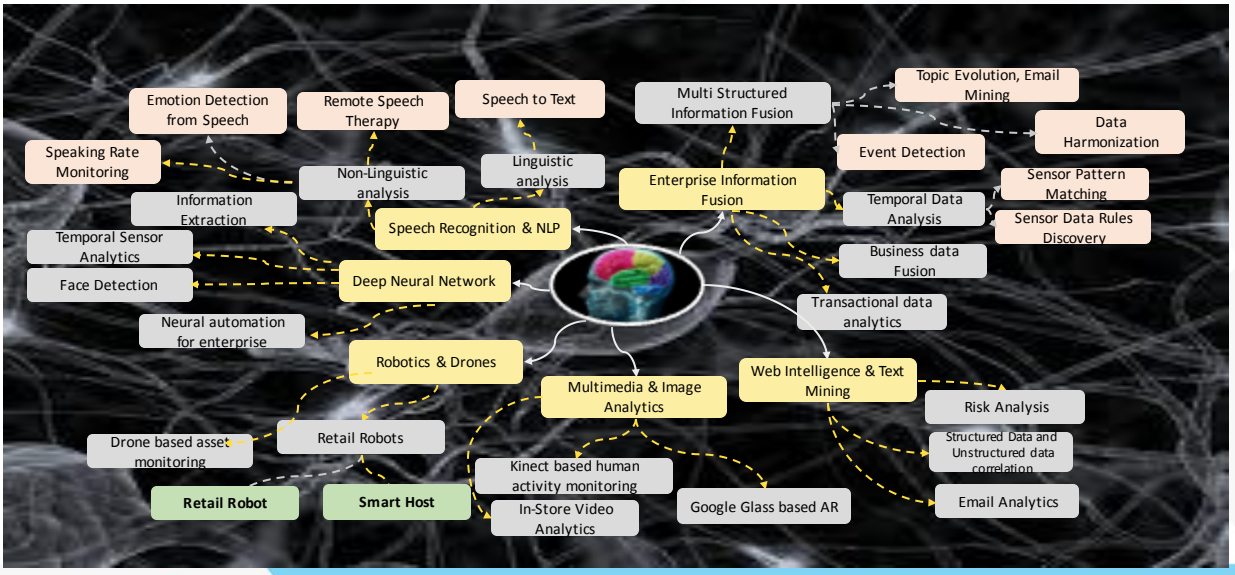
Application in Insurance Domain – Streamline Claims Process

- Conversion of text data combined with structured data can create a complete claim record providing a **360° view of all relevant claim data** for analysis
- Discover **fraud patterns** hidden in Claim Adjuster Notes, Emails, Service notes, Claimant Interviews such as "stopped for no reason", "high usage of technical terms by insured", "felt like a set up", "gap in bills"
- Detect **Fraud rings** by analyzing linkages between different entities, keywords occurring together
- Text mining logs/notes/recorded statements can help in finding patterns in missed **Subrogation opportunities**
- A focus on text can uncover inconsistent use of red flags across claim examiners and identify training opportunities for **productivity improvement**

Features of Text Data & Challenges

- ✓ High dimensionality Large number of features
- ✓ Multiple ways to represent the same concept.
- ✓ Highly redundant data.
- ✓ Unstructured data.
- ✓ Easy for humans, hard for machine. Abstract ideas hard to represent
- ✓ Huge amount of data to be processed.

Artificial Intelligence – Text Mining



ANALYTIXLABS

Typical Text Mining Steps

ANALYTIXLABS

Text Mining Models

In descriptive mining, the textual comments includes providing detailed information about the terms, phrases, and other entities in the textual collection, clustering the documents into meaningful groups, and reporting the concepts that are discovered in the clusters.

In Predictive mining, involves classifying the documents into categories and using the information that is implicit in the text for decision making.

Where does it works?

- Identify and respond of telecom customer experiences into valuable business driven strategies
- Gain a competitive advantage by monitoring the online reputation and that of competitors
- Automatically cluster and categorize call center logs to identify high-volume issues
- Monitor and forecast sentiment prior to and during a product launch
- Identify emerging issues / problematic areas before they become costly problems

ANALYTIXLABS

Text Mining Model

Descriptive Text Mining

Tabulations:

- **Frequency of words (N-gram analysis)**
- **Association Analysis**
- **Sentiment Analysis (Opinion Mining)**
- **Emotion Analysis (pattern approach)**

Visualization:

- **Wordcloud**
- **Bar graphs**
- **Heatmaps**
- **Correlations/associations - heat maps, bar graphs**
- **Scatter - topics, clustering**
- **Network analytics (SNA)**
- **Community detection**

ANALYTIXLABS

Text Mining Model

Predictive analytics:

Segmentation - Identify inherent themes

- **Topic Models(LSA/pLSA/LDA)**
- **Cluster analysis (Hierarchical/K-Means)**

Classification (Machine Learning)

- **Logistic regression**
- **KNN**
- **NB**
- **DT**
- **Ensemble learning(RF, Boosting)**
- **Linear SVM**
- **Deep Learning (RNN/CNN)**

Classification(Deep Learning)

- **Shallow Neural Networks**
- **Convolutional Neural Network (CNN)**
- **Long Short Term Model (LSTM)**
- **Gated Recurrent Unit (GRU)**
- **Bidirectional RNN**
- **Recurrent Convolutional Neural Network (RCNN)**
- **Other Variants of Deep Neural Networks**

Text Mining Analysis

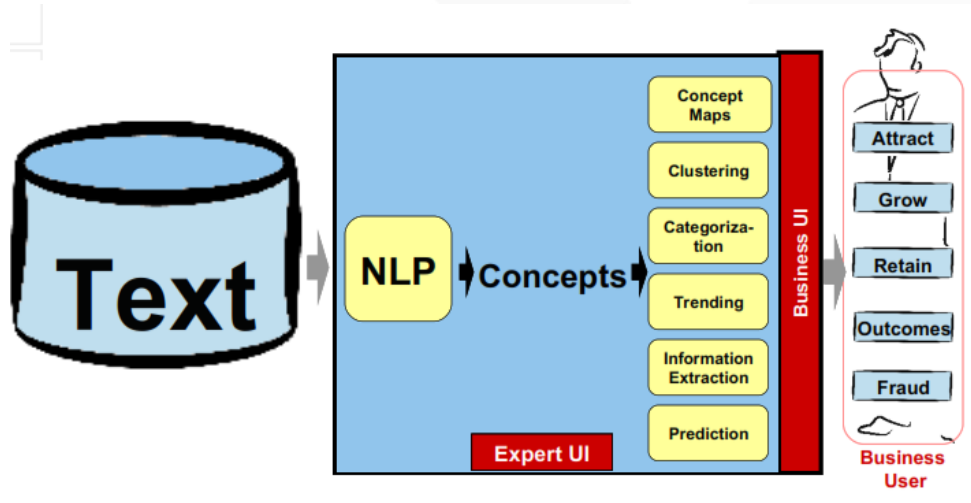
Types of Analysis

- Intent Analysis
- Sentiment analysis(Lexicon/Classification)
- Document Classification
- Text Summarization
- Segmentation - Identify inherent themes
- Social Network analysis - Community detection

Packages in python for text mining

- Packages in python for text mining?
  - textmining1.0: contains a variety of useful functions for text mining in Python.
  - NLTK: This package can be extremely useful because you have easy access to over 50 corpora and lexical resources
  - Tweepy: to mine Twitter data
  - scrappy: extract the data you need from websites
  - urllib2: a package for opening URLs
  - requests: library for grabbing data from the internet
  - Beautifulsoup: library for parsing HTML data
  - re: grep(), grepl(), regexpr(), gregexpr(), sub(), gsub(), and strsplit() are helpful functions
  - wordcloud: to visualize the wordcloud
  - Textblob: package to create blob object and perform text processing

Typical Text Mining Steps



Text mining Process

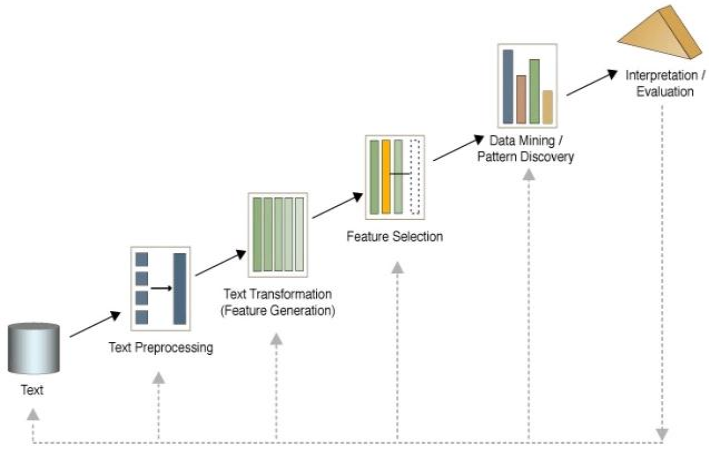
Text Pre Processing
Syntactic/Semantic Text Analytics

Feature generation
Bag of words

Feature selection
Simple Counting Statistics

Text Data mining
Classification-Supervised learning
Clustering-Unsupervised learning

Analyzing Results
Mapping/Visualization
Result interpretation



Iterative and interactive process

ANALYTIXLABS

How is text mining done?

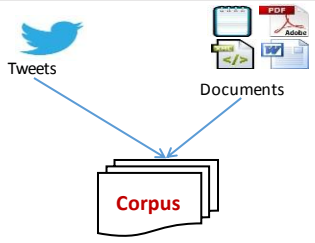
Collection	Pre-processing	Transformation (Feature generation/selection)	Using algorithms (Text/Data Mining)	Evaluation (Analyzing results)
Unstructured data for our analyses can be found in these various forms : • Text files/CSV • JSON files • HTML files • Excel files • Websites The objective is to be able to capture and store our data without compatibility issues. In most cases, the data is readily available (except for data stored on the Web, which needs to be crawled)	Once the data has been loaded, it is vital to make it ready for suitable analyses. A few generic pre-processing steps are : • Tokenizing text • POS tagging • Disambiguation • Cleanup of non-textual data (white space, numbers, symbols)	Transformation would ensure that the text corpus is enriched and homogenous. These processes include : • Stop word removal • Stemming • Transform cases A few of these processes have been briefed in the pages ahead.	Depending on the business problem at hand, we can use several of the basic machine learning algorithms to crunch our text to discover patterns. These include : • Word frequencies and word clouds • K-Means cluster analysis • Association rule generators • K nearest neighbors • Neural networks • Naïve Bayes classifiers • N-gram generators	This is an inferential step. The analytics expert has the option to decide if • another refined iteration of the algorithm is justified, or • is the current result good enough to discover unseen patterns among the text

POS Tagging: Parts of speech Tagging

ANALYTIXLABS

Typical Text Mining Steps

Step 1 – Data Assemble



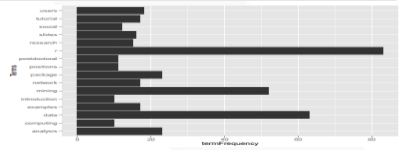
Step 2 – Data Processing

2A - Explore Corpus
2B - Convert text to lowercase
2C - Remove
- a) Numbers(if required)
- b) Punctuations
- c) English stop words
- d) Own stop words(if required)
- e) Strip whitespace
- f) Lemmatization/Stemming
- g) Sparse terms

2D - Create document term matrix

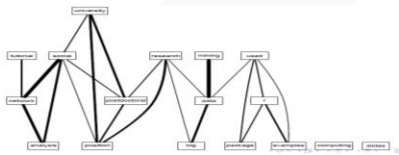
Step 3 - Visualization

Frequency Items



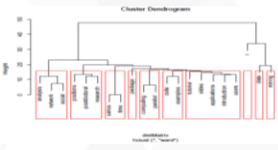
Word Cloud

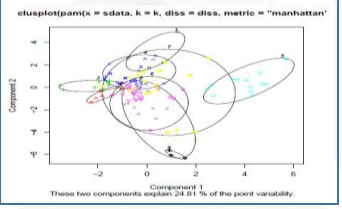
Correlation Plot



Step 4 – Run Model(s)

- Clustering
- Classification
- Sentiment Analysis





(Description about each step is on next slide)

ANALYTIXLABS

Typical Text Mining Model Steps

1) Text Mining starts with text parsing which identifies unique terms in the text variable and identifies parts of speech, entities, synonyms and punctuation.
2) Identification of target variable which describes the serious of textual inputs
3) The terms identified from text parsing are used to create a term-by-document matrix with terms as rows and documents as variables.
4) Stop lists help in reducing the number of rows in the matrix by dropping some of the terms. Stop list is a dictionary of terms that are ignored in the analysis
5) Creation of custom stop lists for getting better text mining results, those terms are deemed as to not add any value to the analysis.
6) Custom synonym data set using with the terms of telecommunication abbreviation's and joined with existing extracted cleaned synonyms.
7) Singular Value Decomposition (SVD) used to reduce the dimensionality by transforming the matrix into a lower dimensional and more compact form .
8) Clustering technique is used for text categorization. Using this technique documents are classified into groups so that documents within any one group are related and documents in different groups are not closely related. Each group or cluster is represented by a list of terms and those terms will appear in most of the documents within the group
9) Applying decision tree model for segmentation and predict the importance of variables
10) Choosing the statistically best model among the three level of approach

ANALYTIXLABS

Key Terms in Text Mining

Information Extraction
Corpus
Documents/Words/Tokenization
TF
TF-IDF
Cosine Similarity
DTM/TDF/DFM/Vectorization
BOW
n-grams
non-textual data (white spaces, numbers, punctuations etc.)
Disambiguation
POS Tagging
Stemming
Lemmatization
StopWords
Sparse terms/Rare Terms

NER
Entity
sentiment/Polarity/Opinion mining/Contextual Text Mining
Intent
Wordcloud
NLP/DNLP
LDA/LSA
topic modeling/Concepts mapping
SNA
Community detection
categorization

ANALYTIXLABS

Basics of NLP

ANALYTIXLABS

NLP vs. Text Mining

- NLP objectives
  - Understanding
  - Ability to answer
  - Immaculate
- Text Mining objectives
  - Overview
  - Know the trends
  - Accept noise

Natural Language Processing

Natural language processing (NLP) is a field of computer science, artificial intelligence, and linguistics concerned with the interactions between computers and human (natural) languages.

As such, NLP is related to the area of human–computer interaction.

Many challenges in NLP involve natural language understanding, that is, enabling computers to derive meaning from human or natural language input, and others involve natural language generation.

NLP vs. NLU vs. NLG

NLP

- Allow computers to understand, analyze and create meaning from human language.
- NLP considers the structure to human language (i.e. Words make a phrase, phrases make sentences which convey the idea or intent the user is trying to invoke).

NLU

- The ambiguous nature of human language makes it difficult for a machine to always Correctly interpret the user's requests.
- Natural Language Understanding (NLU) - how to best handle unstructured inputs that are governed by poorly defined and flexible rules and convert them into a structured form

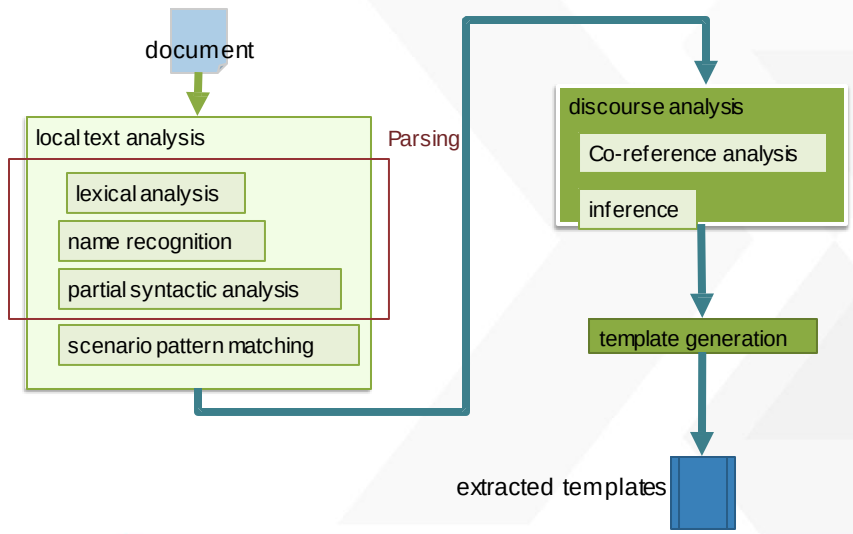
NLG

- Natural language generation (NLG) is the natural language processing task of generating natural language from a machine representation system such as a knowledge base or a logical form
- NLG may be viewed as the opposite of natural language understanding: whereas in natural language understanding the system needs to disambiguate the input sentence to produce the machine representation language, in NLG the system needs to make decisions about how to put a concept into words.

- The intelligence is from getting a natural-language understanding of intentions behind those words.
- The intelligence also combines voice technologies, artificial intelligence reasoning and contextual awareness.

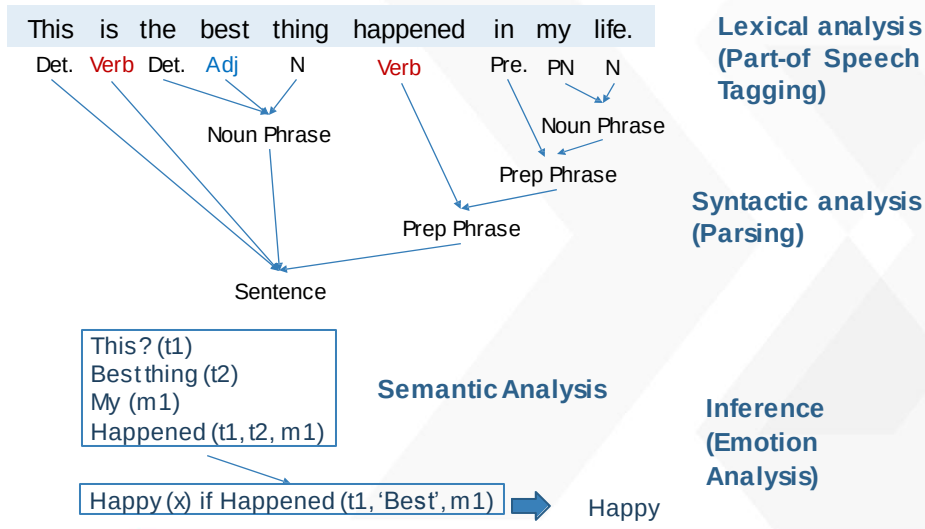
ANALYTIXLABS

Structure of Information Extraction System



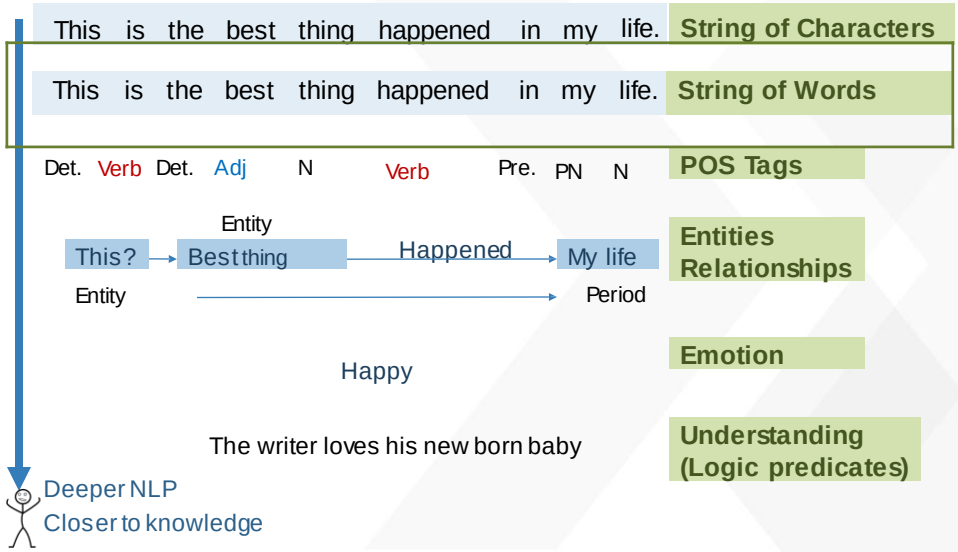
ANALYTIXLABS

Basic Concepts in NLP



ANALYTIXLABS

Basic Concepts in NLP



ANALYTIXLABS

Basic Concepts in NLP

- **Data Processing using NLP - Lower level components**
  - **Tokenization**: breaking text into tokens (words, sentences, n-grams)
  - **Stopword removal**: a/an/the
  - **Stemming and lemmatization**: root word
  - **TF-IDF**: word importance
  - **Part-of-speech tagging**: noun/verb/adjective
  - **Named entity recognition**: person/organization/location
  - **Spelling correction**: "New Yrok City"
  - **Word sense disambiguation**: "buy a mouse"
  - **Segmentation**: "New York City subway"
  - **Language detection**: "translate this page"

ANALYTIXLABS

Document Term Metrics
(Vectorization of data - Feature Engineering)

ANALYTIXLABS

Feature Engineering (Converting text data into numeric data)

Feature Engineering

- ***Term count (unigrams/bigrams/n-grams) – Number of times term t appears in a document***
- ***TF-IDF Vectors as features (unigrams/bigrams/n-grams)***
  - TF(t) = (Number of times term t appears in a document) / (Total number of terms in the document)
  - IDF(t) = log_e(Total number of documents / Number of documents with term t in it)
  - TF-IDF Vectors can be generated at different levels of input tokens (words, characters, n-grams)
    - a. Word Level TF-IDF : Matrix representing tf-idf scores of every term in different documents
    - b. N-gram Level TF-IDF : N-grams are the combination of N terms together. This Matrix representing tf-idf scores of N-grams
    - c. Character Level TF-IDF : Matrix representing tf-idf scores of character level n-grams in the corpus

Text / NLP based features

- **Word Count of the documents** – total number of words in the documents
- **Character Count of the documents** – total number of characters in the documents
- **Average Word Density of the documents** – average length of the words used in the documents
- **Punctuation Count in the Complete Essay** – total number of punctuation marks in the documents
- **Upper Case Count in the Complete Essay** – total number of upper count words in the documents
- **Title Word Count in the Complete Essay** – total number of proper case (title) words in the documents
- **Frequency distribution of Part of Speech Tags** - Noun Count, Verb Count, Adjective Count, Adverb Count, pronoun Count

ANALYTIXLABS

n-gram

Example: "defense attorney for liberty and montecito"

1-gram:
defense
attorney
for
liberty
and
montecito

2-gram:
defense attorney
for liberty
and montecito
attorney for
liberty and
attorney for

3-gram:
defense attorney for
liberty and montecito
attorney for liberty
for liberty and
liberty and montecito

4-gram:
defense attorney for liberty
attorney for liberty and
for liberty and montecito

5-gram:
defense attorney for liberty
and montecito
attorney for liberty and
montecito

Definition:

- n-gram is a contiguous sequence of n items from a given sequence of text
- The items can be syllables, letters, words or base pairs according to the application

Application:

- Probabilistic language model for predicting the next item in a sequence in the form of a (n – 1)
- Widely used in probability, communication theory, computational linguistics, biological sequence analysis

Advantage:

- Relatively simple
- Simply increasing n, model can be used to store more context

Disadvantage:

- Semantic value of the item is not considered

ANALYTIXLABS

Calculate Term Weight – TF-IDF

How frequently term appears?

Term Frequency: TF(t) = (Number of times term t appears in a document) / (Total number of terms in the document)

How **important a term is?**

DF: Document Frequency = d (number of documents containing a given term) / D (the size of the collection of documents)

To normalize take log(d/D), but often D > d and log(d/D) will give negative value. So invert the ratio inside log expression. Essentially we are compressing the scale of values so that very large or very small quantities are smoothly compared

IDF: Inverse Document Frequency IDF(t) = log(Total number of documents / Number of documents with term t in it)

Example:

Consider a document containing 100 words wherein the word **Analytics** appears 3 times

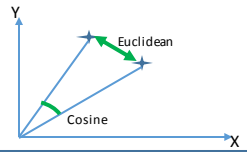
TF(Analytics) = 3 / 100 = 0.03

Now, assume we have 10 million documents and the word **Analytics** appears in one thousand of these

IDF(Analytics) = log(10,000,000 / 1,000) = 4
TF-IDF weight is product of these quantities: 0.03 * 4 = 0.12

ANALYTIXLABS

Similarity Distance Measure



$$similarity = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

- cosine value will be a number between 0 and 1
- Smaller the angel, bigger the cosine value/similarity

Example:

Text 1: statistics skills and programming skills are equally important for analytics
Text 2: statistics skills and domain knowledge are important for analytics
Text 3: I like reading books and travelling

	statistics	skills	and	programming	knowledge	are	equally	important	for	analytics	domain	I	like	reading	books	travelling
Text 1	1	2	1	1	0	1	1	1	1	1	0	0	0	0	0	0
Text 2	1	1	1	0	1	1	0	1	1	1	1	0	0	0	0	0
Text 3	0	0	1	0	0	0	0	0	0	0	0	1	1	1	1	1

The three vectors are:
T1 = (1,2,1,1,0,1,1,1,1,1,0,0,0,0,0,0)
T2 = (1,1,1,0,1,1,0,1,1,1,1,0,0,0,0,0)
T3 = (0,0,1,0,0,0,0,0,0,0,0,0,1,1,1,1,1)

Microsoft Excel 97-2003 Worksheet

Degree of Similarity (T1 & T2) = (T1 %*% T2) / (sqrt(sum(T1^2)) * sqrt(sum(T2^2))) = 77%
Degree of Similarity (T1 & T3) = (T1 %*% T3) / (sqrt(sum(T1^2)) * sqrt(sum(T3^2))) = 12%

ANALYTIXLABS

Additional Reading: *Here is a detailed paper on comparing the efficiency of different distance measures for text documents.*
URL - http://www.ijert.org/view-pdf/2373/space-and-cosine-similarity-measures-for-text-document-clustering

Problems with NLP

- Limitations of Natural Language Processing
  - Correctly identifying the role of noun phrases
  - Representing abstract concepts
  - Classifying synonyms
  - Representing the number of concepts
- Limitations of technology
  - Language specific designs are required
  - Classification speed
  - Classifying hybrid words and sentences

Text is unstructured, ambiguous, and language dependent.

The Linguistic Approach:

- Does not treat a document as a bag of words
- Removes ambiguity by extracting structured concepts

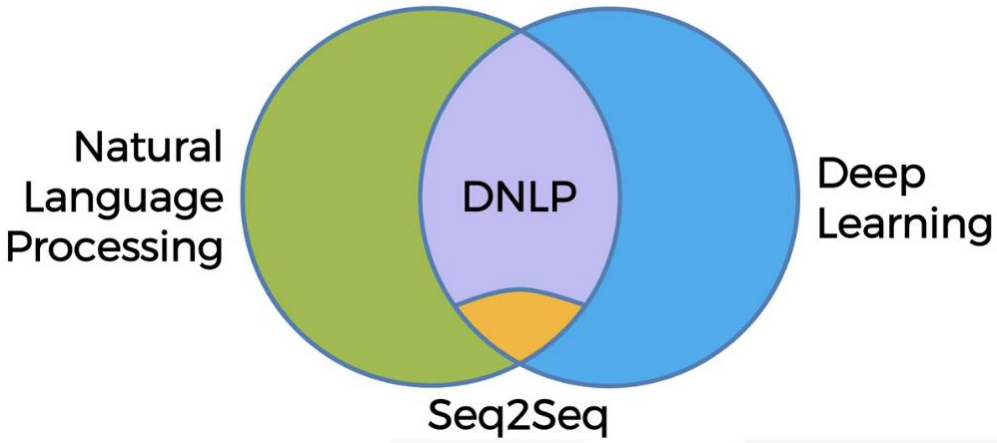
Concepts are the **DNA** of text.

ANALYTIXLABS

Types of NLP

ANALYTIXLABS

Types of Natural Language Processing



ANALYTIXLABS

Shallow NLP Technique

Definition:

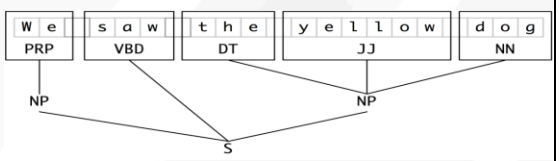
- Assign a syntactic label (noun, verb etc.) to a chunk
- Knowledge extraction from text through semantic/syntactic analysis approach

Application:

- Taxonomy extraction (predefined terms and entities)
  - Entities: People, organizations, locations, times, dates, prices, genes, proteins, diseases, medicines
- Concept extraction (main idea or a theme)

Advantage:

- Less noisy than n-grams



Disadvantage:

- Does not specify role of items in the main sentence

ANALYTIXLABS

Shallow NLP Technique

Sentence - "The driver from Europe crashed the car with the white bumper"

Concept Extraction:

1-gram	Part of Speech
the	DT – Determiner
driver	NN - Noun, singular or mass
from	IN - Preposition or subordinating conjunction
europe	NNP - Proper Noun, singular
crashed	VBD - Verb, past tense
the	DT – Determiner
car	NN - Noun, singular or mass
with	IN - Preposition or subordinating conjunction
the	DT – Determiner
white	JJ – Adjective
bumper	NN - Noun, singular or mass

- Convert to lowercase & PoS tag
- Remove Stop words
- Retain only Noun's & Verb's
- Bi-gram with Noun's & Verb's retained

Bi-gram	PoS
car white	NN JJ
crashed car	VBD NN
driver europe	NN NNP
europe crashed	NNP VBD
white bumper	JJ NN

- 3-gram with Noun's & Verb's retained

3-gram	PoS
car white bumper	NN JJ NN
crashed car white	VBD NN JJ
driver europe crashed	NN NNP VBD
europe crashed car	NNP VBD NN

Conclusion:
1-gram: Reduced noise, however no clear context
Bi-gram & 3-gram: Increased context, however there is a information loss

PoS Tagging: http://nlp.stanford.edu:8080/corenlp/process
NER Demo: http://nlp.stanford.edu:8080/ner/process

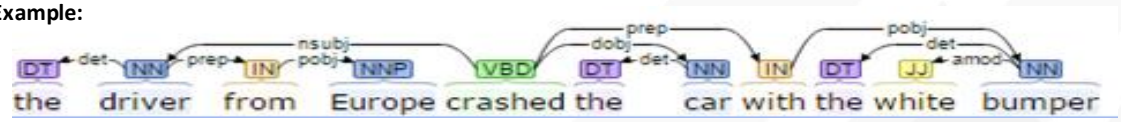
ANALYTIXLABS

Deep NLP technique

Definition:
- Extension to the shallow NLP
- Detected relationships are expressed as complex construction to retain the context
- Example relationships: Located in, employed by, part of, married to

Applications:
- Develop features and representations appropriate for complex interpretation tasks
  - Fraud detection
  - Life science: prediction activities based on complex RNA-Sequence

Example:



The above sentence can be represented using **triples (Subject: Predicate [Modifier]: Object)** without loosing the context.

Triples:
driver : crash : car
driver : crash with : bumper
driver : be from : Europe

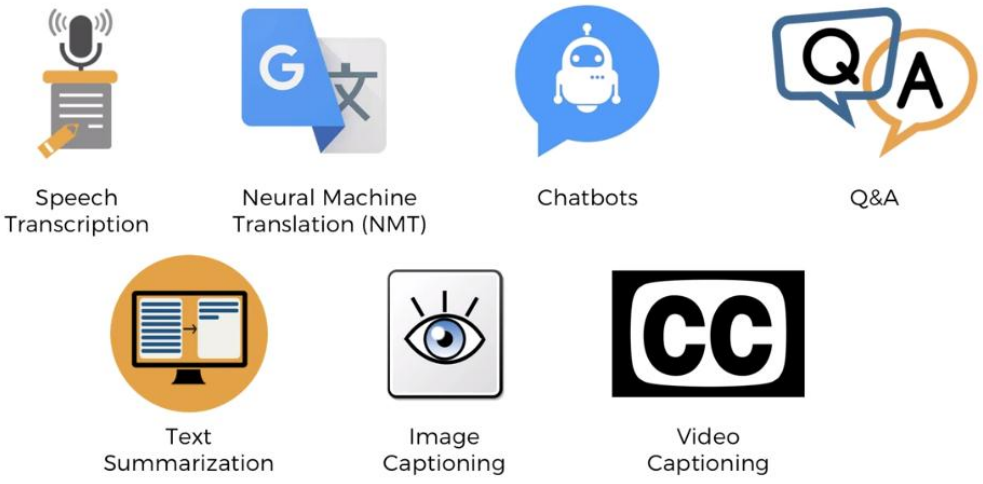
ANALYTIXLABS

Techniques - Summary

Technique	General Steps	Pros	Cons
N-Gram	- Convert to lowercase - Remove punctuations - Remove special characters	Simple technique	Extremely noisy
Shallow NLP technique	- **POS tagging** - **Lemmatization** i.e., transform to dictionary base form i.e., "produce" & "produced" become "produce" - **Stemming** i.e., transform to root word i.e., 1) "computer" & "computers" become "comput" 2) "product", "produce" & "produced" become "produc" - **Chunking** i.e., identify the phrasal constituents in a sentence , including noun/verb phrase etc., and splits the sentence into chunks of semantically related words	Less noisy than N-Grams	Provide a relatively less, computationally expensive solution for analyzing the structure of texts. Does not specify the internal structure or the role of words in the sentence
Deep NLP technique	- Generate syntactic relationship between each pair of words - Extract subject, predicate, nagation, objecct and named entity to form triples.	Context of the sentence is retained.	Sentence level analysis is too structured

ANALYTIXLABS

Deep Natural Processing - Applications



ANALYTIXLABS

Deep Natural Language Understanding & Speech Analytics

3 Aspects of Speech

Who is Speaking?	What is being spoken?	How it is spoken?
▪ How many people are speaking? ▪ What's the gender of speaker? ▪ Language of the speaker	▪ Content of the speech ▪ Does speech has some specific keywords?	▪ Assent of the speaker ▪ Tone of the speaker ▪ Emotion of the speaker

Two Approaches To Analyze speech

Speech to Text Analytics

Dependent on Automatic Speech Recognition (ASR)
Calls are analyzed on **"WHAT"** is being spoken during the call conversation.

Non Speech to Text Analytics

- Independent of Automatic Speech Recognition (ASR)
- Can handle mixed language calls
- Calls are analyzed on **"HOW"** speech is being spoken and not **"WHAT"** is being spoken.

Use Cases


Speech to Text, Text Analytics


Audio Analysis, Speaking Rate analysis

Speaker Verification / Identification


Call Centre Automation


Analysing Emotion in speech

ANALYTIXLABS

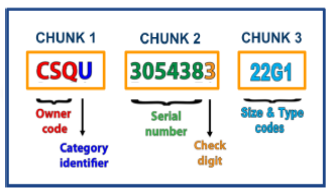
Deep Learning based Text detection from container (DL for vision)

Customer: One of the top 3 leading port operators in the world

Challenge: Automatic container number detection from the camera feed

Solution: Deep Learning based solution for reading container number

Techniques : Spatial *Transformer Networks and Connected Component Region*



WIP

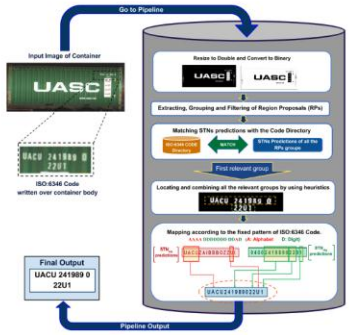
Sample Real world Images

ISO:6346 Container Code Pattern

AAAA DDDDDDD DDAD

where A represents an alphabet
D represents a digit

Deep Learning based solution



ANALYTIXLABS

Popular NLP/DNLP Models

Bag of Words Model

Training Data:

Hey mate, have you read about Hinton's capsule networks?	→	No
Did you like that recipe I sent you last week?	→	Yes
Hi Kirill, are you coming to dinner tonight?	→	Yes
Dear Kirill, would you like to service your car with us again?	→	No
Are you coming to Australia in December?	→	Yes
...	→	...

Hello Kirill, Checking if you are back to Oz. Let me know if you are around ... Cheers, V — Yes / No ?

Bag of Words Model

Training Data:

[1, 1, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 1, 0, 0, … , 2] → No

[1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 1, 0, 0, 1, 0, 0, … , 0] → Yes

[1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, … , 1] → Yes

[1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, … , 1] → No

[1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 1, 0, … , 1] → Yes

… → …

Hello Kirill, Checking if you are back to Oz. Let me know if you are around … Cheers, V

[1, 1, 0, 0, 1, 0, 2, 0, 1, 0, 0, 0, 0, 0, 0, 1, 2, 0, 0, 0, 1, 0, 0, 1, 0, 0, … , 3]

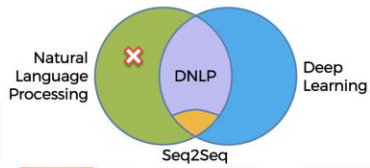
20,000 elements long

[1, 1, 0, 0, 1, 0, 2, 0, 1, 0, 0, 0, 0, 0, 0, 1, 2, 0, 0, 0, 1, 0, 0, 1, 0, 0, … , 3] → Yes / No ?

20,000 elements long

ANALYTIXLABS

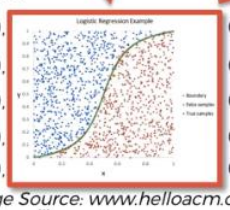
Bag of Words Model



[1, 1, 0, 0, 1, 0, 2, 0, 1, 0, 0, 0, 0, 0, 0, 1, 2, 0, 0, 0, 1, 0, 0, 1, 0, 0, … , 3] → Yes / No ?

20,000 elements long

Training Data:

[1, 1, 0, 0,  0, 0, 1, 0, 1, 0, 1, 0, 0, 1, 0, 0, … , 2] → No

[1, 1, 0, 0, 0, 0, 2, 0, 0, 0, 1, 0, 0, 1, 0, 0, … , 0] → Yes

[1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, … , 1] → Yes

[1, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, … , 1] → No

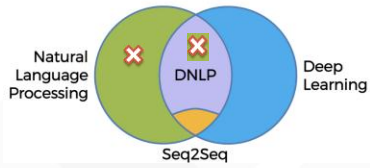
[1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, … , 1] → Yes

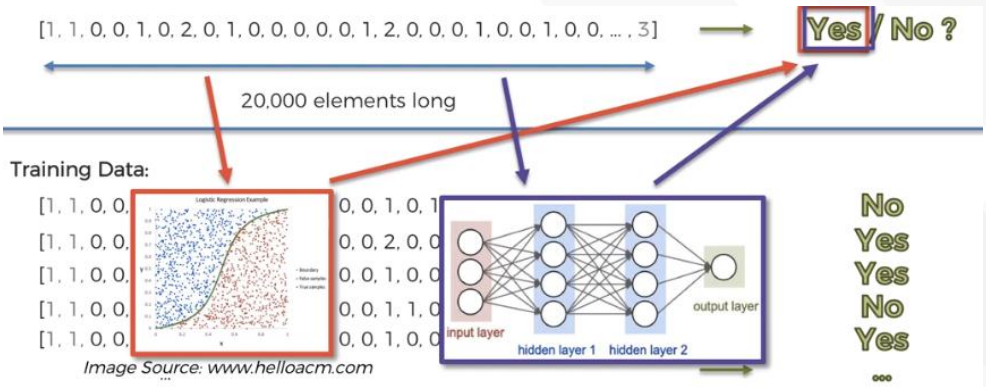
…

Image Source: www.helloacm.com

ANALYTIXLABS

Bag of Words Model





Issues with Bag of Words Model

Issues with Bag of Word Model

1. **Fixed sized input**
2. **Doesn't take word order into account**
3. **Fixed sized output**

Seq2Seq Architecture

For Chatbots, This architecture works better based on the nature of problem

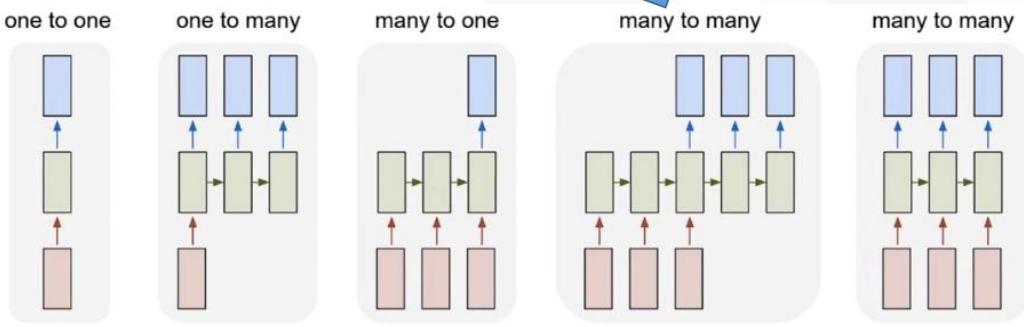
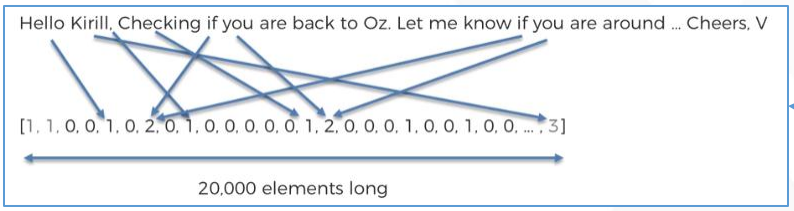


Image Source: karpathy.github.io

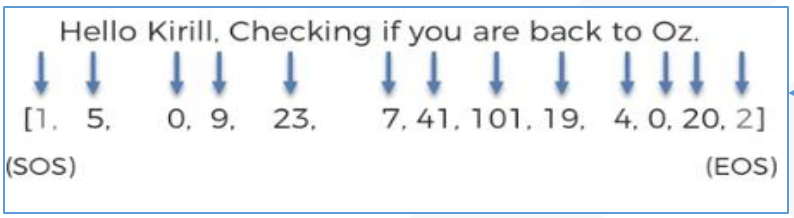
Note: Each Box represents whole layer of neurons

ANALYTIXLABS

Vectorization (Bag of words vs. Seq2Seq)



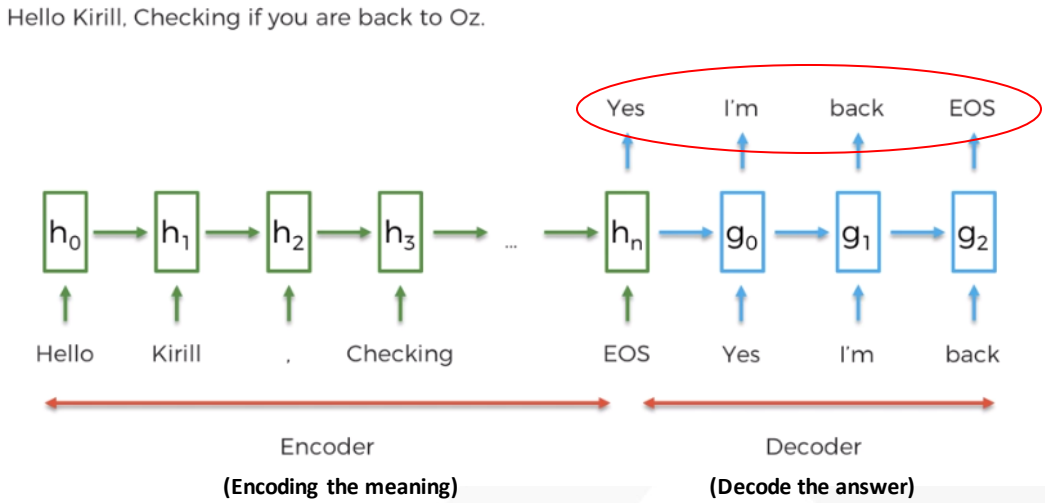
Based on count vectorizer



Based on Position

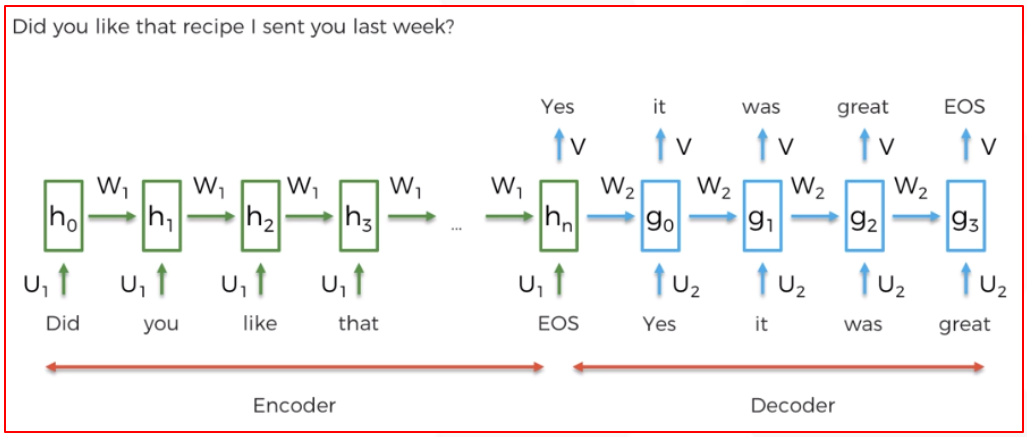
ANALYTIXLABS

Seq2Seq (Encoding vs. Decoding)



(Encoding the meaning) **(Decode the answer)**

Seq2Seq Training

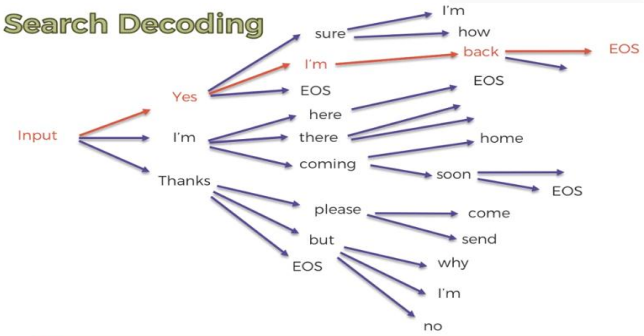


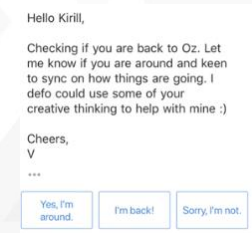
(Encoding the meaning) **(Decode the answer)**

Greedy Decoding vs. Beam Search Decoding

Greedy Decoding

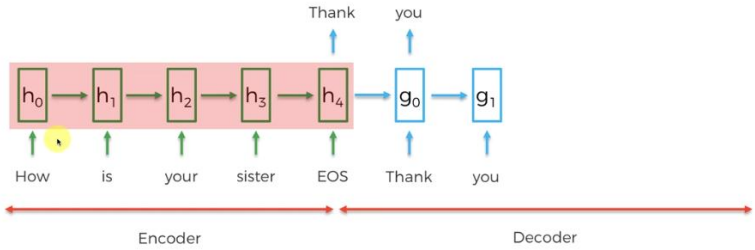
Beam Search Decoding

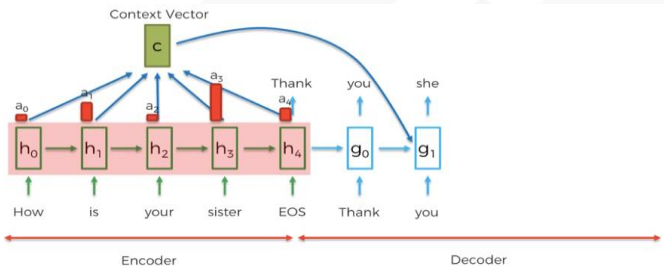




ANALYTIXLABS

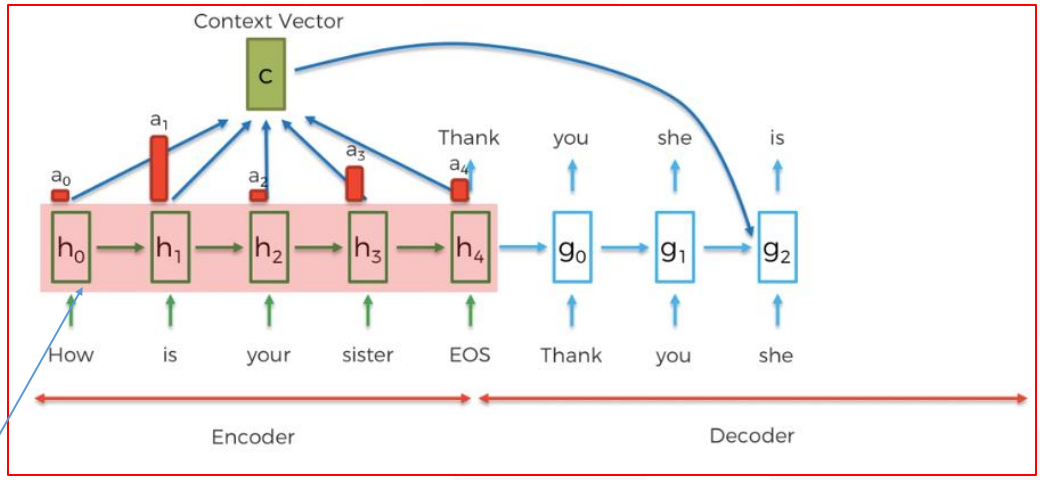
Attention Mechanism





ANALYTIXLABS

Attention Mechanism (Global vs. Local Attention)



Global Attention

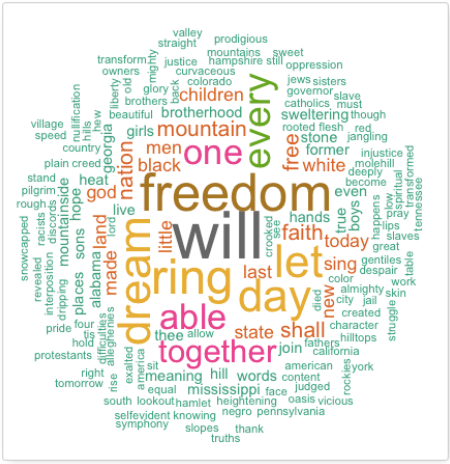
ANALYTIXLABS

Types of Analysis (Descriptive & Predictive)

ANALYTIXLABS

Word clouds

- A **word cloud** is a **text mining** method that allows us to highlight the most frequently used keywords in a paragraph of texts.
- It is also referred to as a **text cloud** or **tag cloud**.
- A **text mining** package (tm) and **word cloud generator** package (**wordcloud**) are available in R for helping us to analyze texts and to quickly visualize the keywords words as a **word cloud**.



ANALYTIXLABS

Word clouds

3 reasons you should use word clouds to present your text data

- **Tag cloud** is a powerful method for **text mining** and, it add simplicity and clarity. The most used keywords stand out better in a word cloud
- **Word clouds** are a potent communication tool. They are easy to understand, to be shared and are impactful
- **Word clouds** are visually engaging than a table data

Who is using word clouds ?

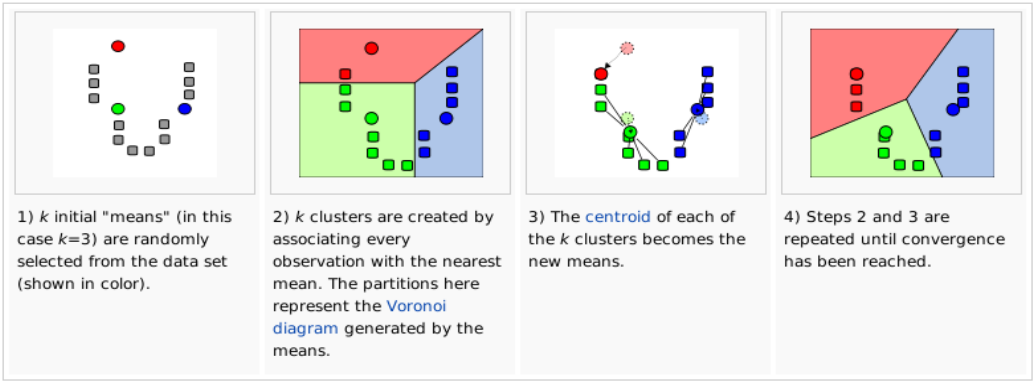
- Researchers : for reporting qualitative data
- Marketers : for highlighting the needs and pain points of customers
- Educators : to support essential issues
- Politicians and journalists
- social media sites : To collect, analyze and share user sentiments

ANALYTIXLABS

K-Means clustering

- unsupervised learning
- group **n** documents into **k** clusters

Demonstration of the standard algorithm



1) *k* initial "means" (in this case *k*=3) are randomly selected from the data set (shown in color).

2) *k* clusters are created by associating every observation with the nearest mean. The partitions here represent the Voronoi diagram generated by the means.

3) The centroid of each of the *k* clusters becomes the new means.

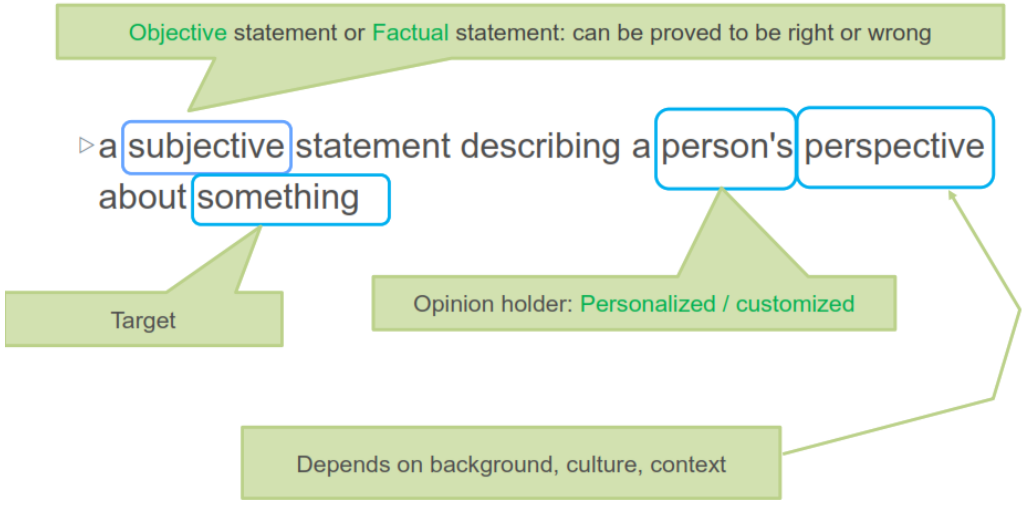
4) Steps 2 and 3 are repeated until convergence has been reached.

ANALYTIXLABS

Opinion Mining (Sentiment Analysis)

ANALYTIXLABS

Opinion



ANALYTIXLABS

Opinion Representation

- Opinion holder: user
- Opinion target: object
- Opinion content: keywords?
- Opinion context: time, location, others?
- Opinion sentiment (emotion): positive/negative, happy or sad

ANALYTIXLABS

Sentiment Analysis

✓Sentiment analysis refers to the use of natural language processing, text analysis and computational linguistics to identify and extract subjective information in source materials.

✓Generally speaking, sentiment analysis allows companies the ability to measure how positive or negative a person feels about their product and service.

✓Companies look at:

Reviews/Surveys
Complaints/Fees/Prices
Password recovery
Technical issues

Sentiment Analysis

Methods:

- ✓Scaling Systems (-10/+10)
- ✓Subjectivity/Objectivity Identification
- ✓Feature/Aspect Based Analysis

Risk:

- ✓Losing shifting and subjective human dynamics
- ✓Computers can not tell the context of a statement.
- ✓*Ex: sarcasm, slang, double négatives*

Sentiment Analysis

- Input: An opinionated text object
- Output: Sentiment tag/Emotion label
  - Polarity analysis: {positive, negative, neutral}
  - Emotion analysis: happy, sad, anger
- Naive approach:
  - Apply classification, clustering for extracted text features

Sentiment Analysis – Text Features

- Character *n*-grams
  - Usually for spelling/recognition proof
  - Less meaningful
- Word *n*-grams
  - n should be bigger than 1 for sentiment analysis
- POS tag *n*-grams
  - Can mixed with words and POS tags
    → E.g., “adj noun”, “sad noun”

Topic Modeling

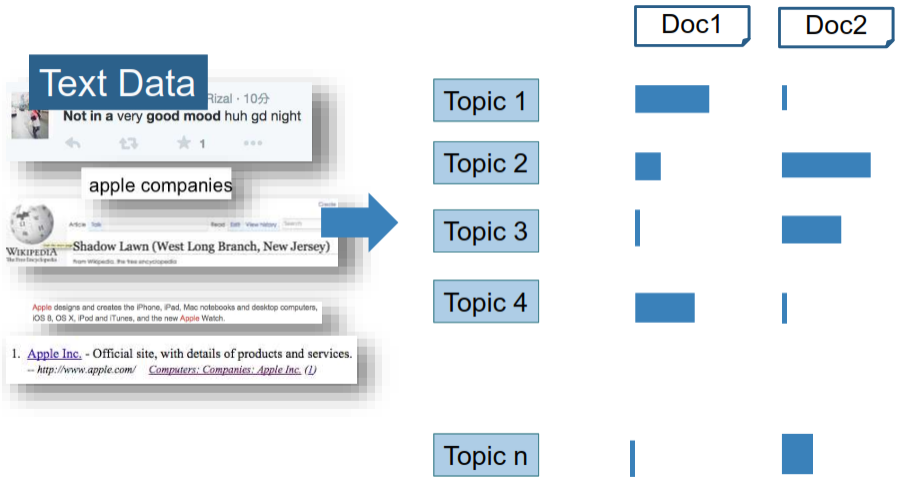
ANALYTIXLABS

Topic Modeling - Motivation

- Topic: key idea in text data
  - Theme/subject
  - Different granularities (e.g., sentence, article)
- Motivated applications, e.g.:
  - Hot topics during the debates in 2016 presidential election
  - What do people like about Windows 10
  - What are Facebook users talking about today?
  - What are the most watched news?

ANALYTIXLABS

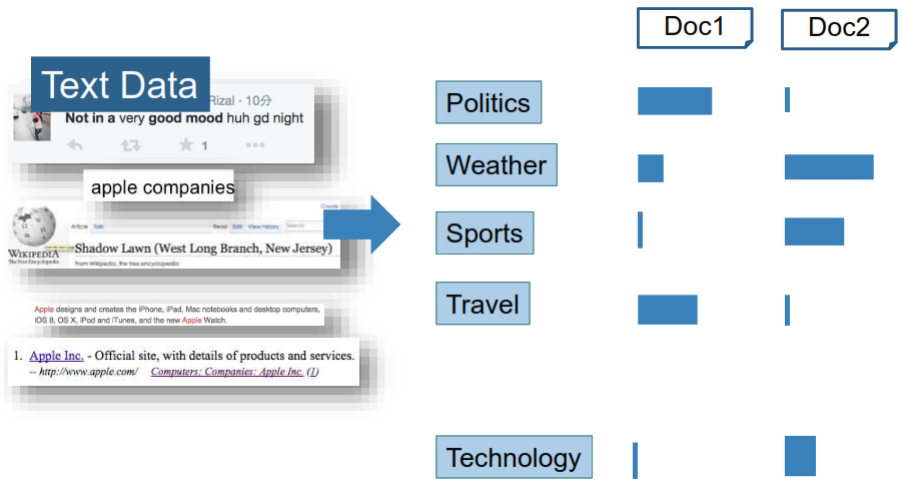
Tasks of Topic Mining



Definition of Topic Modeling

- Input
  - A collection of N text documents $S = \{d_1, d_2, d_3, \ldots d_n\}$
  - Number of topics: k
- Output
  - k topics: $\{\theta_1, \theta_2, \theta_3, \ldots \theta_n\}$
  - Coverage of topics in each d_i: $\{\mu_{i1}, \mu_{i2}, \mu_{i3}, \ldots \mu_{in}\}$
- How to define topic θ_i?
  - Topic=term (word)?
  - Topic= classes?

Tasks of Topic Mining (Terms as topic)

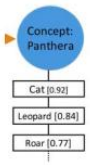


Problems with "Terms as topic"

- Not generic
  - Can only represent simple/general topic
  - Cannot represent complicated topics
    - → E.g., "uber issue": political or transportation related?
- Incompleteness in coverage
  - Cannot capture variation of vocabulary
- Word sense ambiguity
  - E.g., Hollywood star vs. stars in the sky; apple watch vs. apple recipes

Improved ideas for Topic Modeling

- Idea1 (Probabilistic topic models): topic = word distribution
  - E.g.: Sports = {(Sports, 0.2), (Game 0.01), (basketball 0.005), (play, 0.003), (NBA,0.01)…}
  - √: generic, easy to implement
- Idea 2 (Concept topic models): topic = concept
  - Maintain concepts (manually or automatically)
    → E.g., ConceptNet



ANALYTIXLABS

Probable Approaches for Topic Modeling

- Bag-of-words approach:
  - Mixture of unigram language model
  - Expectation-maximization algorithm
  - Probabilistic latent semantic analysis
  - Latent Dirichlet allocation (LDA) model
- Graph-based approach :
  - TextRank (Mihalcea and Tarau, 2004)
  - Reinforcement Approach (Xiaojun et al., 2007)
  - CollabRank (Xiaojun er al., 2008)

ANALYTIXLABS

Topic Modeling - LDA

- I eat fish and vegetables.
- Dog and fish are pets.
- My kitten eats fish.

Topic 1

0.268	fish
0.210	pet
0.210	dog
0.147	kitten

Sentence 1: 14.67% Topic 1, 85.33% Topic 2
Sentence 2: 85.44% Topic 1, 14.56% Topic 2
Sentence 3: 19.95% Topic 1, 80.05% Topic 2

Topic 2

0.296	eat
0.265	fish
0.189	vegetable
0.121	kitten

ANALYTIXLABS

NLP Related Approach

- Find and classify all the named entities in a text.
- What's a named entity?
  - A mention of an entity using its name.
    → *Kansas Jayhawks*
  - This is a subset of the possible mentions...
    → *Kansas, Jayhawks, the team, it, they*
- Find means identify the exact span of the mention
- Classify means determine the category of the entity being referred to

ANALYTIXLABS

Named Entity Recognition Approach

- As with partial parsing and chunking there are two basic approaches (and hybrids)
  - Rule-based (regular expressions)
    - → Lists of names
    - → Patterns to match things that look like names
    - → Patterns to match the environments that classes of names tend to occur in.
  - Machine Learning-based approaches
    - → Get annotated training data
    - → Extract features
    - → Train systems to replicate the annotation

ANALYTIXLABS

Rule Based Approaches

- Employ regular expressions to extract data
- Examples:
  - Telephone number: (\d{3}[-.\ ()]){1,2}[\dA-Z]{4}.
    - → 800-865-1125
    - → 800.865.1125
    - → (800)865-CARE
  - Software name extraction: ([A-Z][a-z]*\s*)+
    - → Installation Designer v1.1

ANALYTIXLABS

Contextual Text Mining

ANALYTIXLABS

Context

- Text usually has rich context information
  - Direct context (meta-data): time, location, author
  - Indirect context: social networks of authors, other text related to the same source
  - Any other related text
- Context could be used for:
  - Partition the data
  - Provide extra features

ANALYTIXLABS

Contextual Text Mining

- Query log + User = Personalized search
- Tweet + Time = Event identification
- Tweet + Location-related patterns = Location identification
- Tweet + Sentiment = Opinion mining

- Text Mining +Context → Contextual Text Mining

ANALYTIXLABS

Generative model of Text

$P(word \mid Topic)$

$P(Topic \mid Document)$

Generation

Dog eat and fish kitten pets I are My vegetables

I eat fish and vegetables. Dog and fish are pets. My kitten eats fish.

Analyze Model

Topic 1

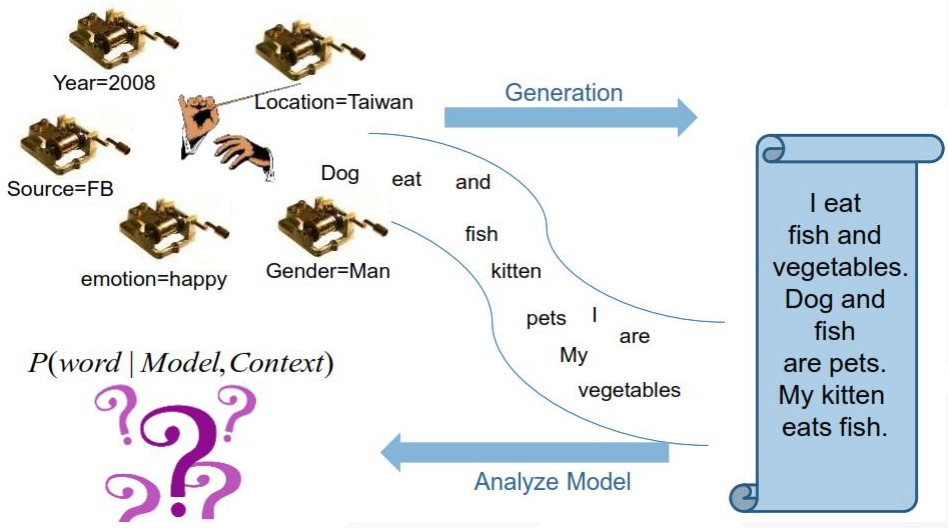
0.268	
0.210	
0.210	
0.147	

Topic 2

0.296	eat
0.265	fish
0.189	vegetable
0.121	kitten

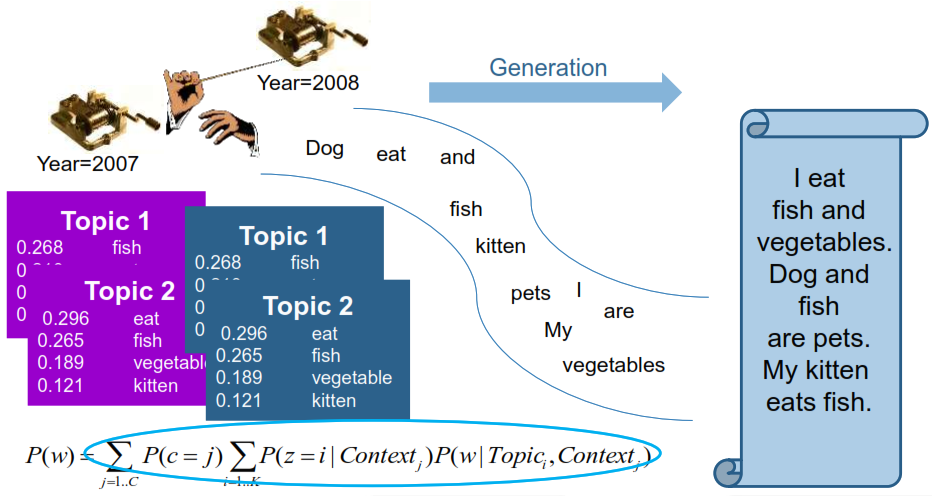
ANALYTIXLABS

Contextualized Model of Text



ANALYTIXLABS

Naïve Contextual topic Model



How do we estimate it? → Different approaches for different contextual data and problems

ANALYTIXLABS

Different Approaches

- An extension of PLSA model ([Hofmann 99]) by
  - Introducing context variables
  - Modeling views of topics
  - Modeling coverage variations of topics
- Process of contextual text mining
  - Instantiation of CPLSA (*context*, *views*, *coverage*)
  - Fit the model to text data (EM algorithm)
  - Compare a topic from different views
  - Compute strength dynamics of topics from coverages
  - Compute other probabilistic topic patterns

ANALYTIXLABS

Use Case of Airlines

ANALYTIXLABS

Objective

Text Analytics helps to identify potential business opportunities for one of leading Manufacture Client

Airlines
- American
- Lufthansa

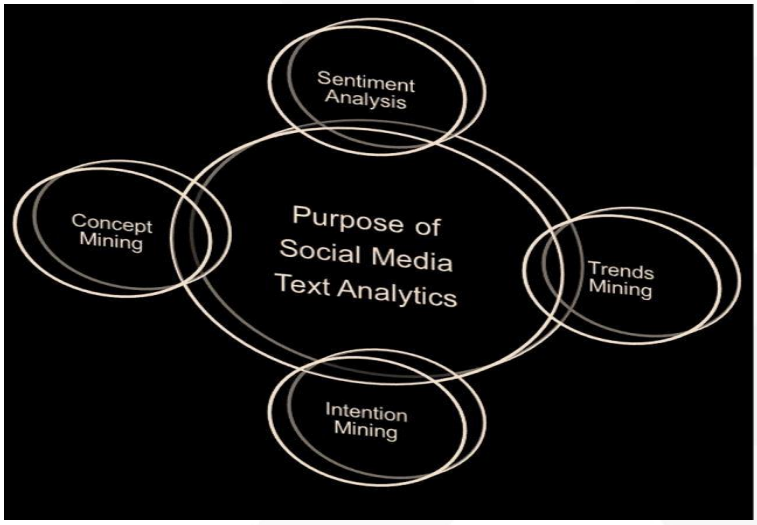
Data Sources
- Press releases
- News articles
- Facebook
- Twitter

Keywords
- Technology
- Digital
- Responsibilities
- Vision
- Innovation
- Goal

- Extracted ~100 news articles for American & Lufthansa airlines from various sources.
- Official websites of respective airlines has been used during data collection.
  - http://hub.aa.com/en/nr/pressrelease/fleet
  - https://www.facebook.com/AmericanAirlines
  - https://twitter.com/americanair
  - https://www.lufthansagroup.com/en/press/news-releases/press-releases.html
  - https://www.facebook.com/lufthansa/
  - http://airwaysnews.com/
  - http://aviationblog.dallasnews.com/
  - http://edition.cnn.com/

ANALYTIXLABS

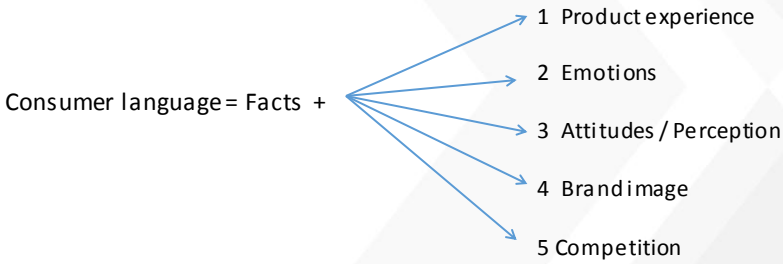
Purpose of Social Media Text Analytics



ANALYTIXLABS

Why do we need to analyze consumer language?

To help discover the true value of information!



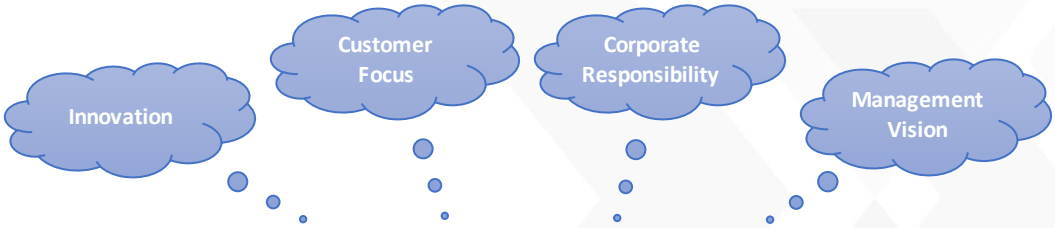
ANALYTIXLABS

Word Cloud

Most frequently used words were "Technology, Customers, Services, Digital, Strategic, opportunities, Security, Quality, Growth, etc.." **by American & Lufthansa** in multiple forums

ANALYTIXLABS

Topic Modeling



Topic-1	Topic-2	Topic-3	Topic-4
technology	services	strategic	operations
security	digital	sustainability	financial
climate	customer	corporate	maintenance
safety	entertainment	opportunities	vision
research	baggage	responsibility	management
noise	economy	environment	organization

ANALYTIXLABS

Topic - 1

American Airlines:

- Aircraft upgrades: Modern, more efficient aircraft. (600+)
- Safety and Security: new aircraft technology - **Rockwell Collins' MultiScanThreatTrack weather radar.**
- Reducing **greenhouse emissions, Flying smarter**

Lufthansa Airlines:

- "Innovation hub" established in Berlin, closer to the start-up and digital technology scene.
  - 500 million euros to be invested in innovations by 2020
  - **Fuel efficiency, Noise protection and Climate research project IAGOS, Flying Lab, Zero-G arm**
- Invests massively in ecological sustainability of flight operations:
  - Biggest fleet renewal program and invests in **highly efficient and quiet aircraft.** by 2025.
  - New engine technology, the 85 decibel noise contour of an A320neo at take-off. **Ordered a total of 116 aircraft** of this type.
  - **"vortex generators"** under the wings.

Innovation
technology
security
climate
safety
research
noise

Topic - 2

American Airlines:

- **Premium Economy**
  - Boeing 787-9, which is expected to enter service in late 2016.
  - In Airbus A350, which arrives in 2017.
  - Boeing 777-300ERs, 777-200ERs, 787-8s and Airbus A330s over the next three years.
- **T-link software** – Improve Baggage handling
- **Baggage Reroute Tool** help's to manage baggage when customers are rerouted.

Lufthansa Airlines:

- The **Innovation Hub**, to ensure identifying future customer needs and trends at an early stage and participates in shaping them.
- BoardConnect - wireless service for customers to use their own devices to access entertainment.
- **Big Data and Analytics Technology:**
  - Location-based services
  - Electronic baggage receipt(**RIMOWA** Electronic Tag & Lufthansa app)
  - **SMILE program** – Surpass My Individual Lufthansa Experience
- **Flight planning app,** enables passengers to plan travel (provide information about airport traffic and security) **Coming Soon**

Customer Focus
services
digital
customer
entertainment
baggage
economy

ANALYTIXLABS

Topic - 3

American Airlines:

- Environmental Performance:
  - Join the EPA's Climate Leaders program, committed to a **30% reduction in greenhouse gas** intensity ration by 2025 and will work with Climate Leaders to set a mid-range goal to help meet this long-range target.
  - Employee-led **Fuel Smart fuel** conservation program
- Corporate Citizenship
  - Donations towards Education, Kids, Partnership & programs
  - The Police Athletic League of Philadelphia (PAL) announced today a $180,000 gift from American Airlines

Lufthansa Airlines:

- Comprehensive sustainability agenda with following files of entrepreneurial responsibility:
  - Economic sustainability
  - Corporate Governance and compliance
  - **Climate and environmental responsibility**
  - Social responsibility
  - Product responsibility
  - Corporate citizenship

Corporate Responsibility
strategic
sustainability
corporate
opportunities
responsibility
environment

Topic - 4

American Airlines:

- Premium Economy - New Planes and Retrofit Plans
- **Adding up to 2 new aircraft to the fleet each week**. The new build planes are replacing older retiring aircraft.
- AA Tulsa (Maintenance & Engineering Base)
  - American's entire program of **fuel efficiency** and range increasing winglets conversion to its 737, 757, and 767 fleet all occurred in house.
  - They accomplished this with a **45% reduction** in the OSHA injury rate from 2009.
- CEO's Vision of Labor Peace.

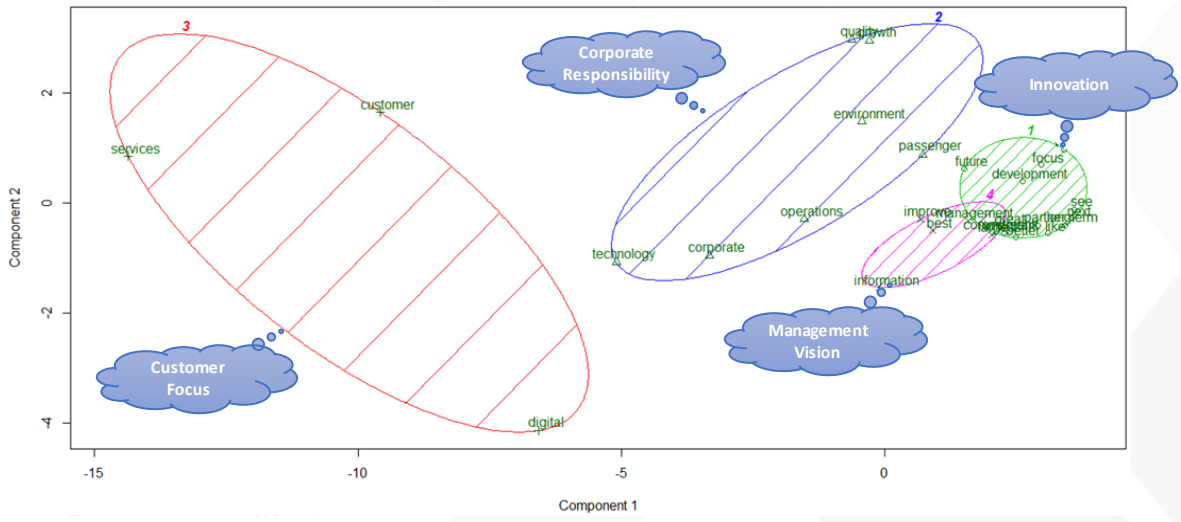
Lufthansa Airlines:

- **" 7 to 1 – Our Way Forward"** strategic agenda was initiated in 2014
- **Airline Business technology award** - 2013 in MRO (maintenance, repair and overhaul) services.
  - The **"Taxibot"** - pilot-controlled tow-tractor
  - Improve **fuel efficiency,** testing new aircraft **paint** with a sharkskin-inspired riblet texture

Management Vision
operations
financial
maintenance
vision
management
organization

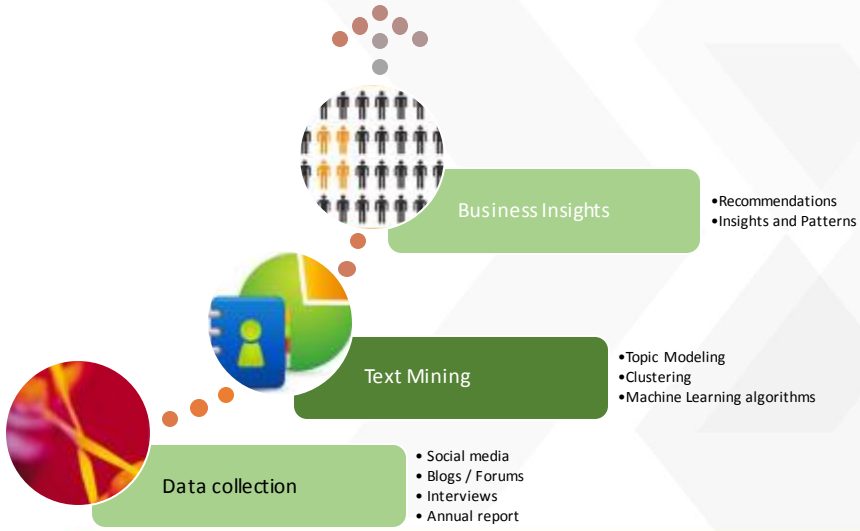
ANALYTIXLABS

Clustering



ANALYTIXLABS

Next Steps



ANALYTIXLABS

Example: Survey sentiment analysis

ANALYTIXLABS

Example: Survey sentiment analysis and clustering

Objective	A major US retail bank conducted a diagnostic around the workplace technologies that they are using through a collection of surveys from an internal employee satisfaction survey. The task was to find out the themes on workplace technologies (e.g. lotus notes, video conferencing, Wi-Fi, OS etc.) and the effect of the technology on productivity.
Analysis Input	An excel file showing the results of the survey along with the comments
Analysis Process	▪ The input of the text file was given directly to the tropes software. Tropes has the option of customizing and adding our own scenarios, but for this particular case, it is not required. ▪ The process included - sentence and proposition Hashing, ambiguity solving (with respect to the words of the text), detection of episodes, detection of the most characteristic parts of text, layout and display of the result. ▪ During the process, the software will: — assign all the significant words to the above categories — analyze their distribution into subcategories (Word categories, Equivalent classes, see below) — examine their occurrence order, both within the propositions (Relations, Actant and Acted) throughout the text
Analysis Output	Through the use of word counts for various relations and scenarios, a table was compiled showing the themes in the technology environment

ANALYTIXLABS

Customer Complaint Classification (Intent Analysis)

ANALYTIXLABS

Customer Complaints Prioritization

Current Project Objective:
Using historical unstructured data of Customer complaints, classify the complaints in different categories (NAC, LIC, AER) and provide matched case resolutions through Text mining and Machine Learning Algorithms

Proposed Project Benefits:
- Reduce Manual intervention
- Minimized Time, Effort & Cost (quantification can be done once the date is available)
- Focused resolution efforts by accurate classification of customer complaints

Our understanding of requirement:

1. Huge volumes –Thousands of customer complaints received each day
2. Unstructured text format of data
3. Lot of time is being spent to identify NAC/LIC - Not A Complaint or Low Impact Complaint
4. Prioritization of customer complaints to meet SLAs is a challenge

Solution offered:

1. Use of historical data
2. Text mining functions (Corpus, Data cleaning, DTM, Cosine similarity, NLP etc.)
3. Develop machine learning algorithm on the customer complaints text data
4. Train the ML model to get the highest level of accuracy

CUSTOMER COMPLAINTS PRIORITIZATION (PROJECT EXECUTED):

- ✓ Use of historical data
- ✓ Text mining functions (Corpus, Data cleaning, DTM, Cosine similarity, NLP etc)
- ✓ Developed a completely different and cutting edge complaint prioritization framework
- ✓ Applied sophisticated machine learning techniques such as Random Forest, Support Vector Machine, Neural Networks and Generalized Linear Model

Benefits Delivered:
- **5% reduction in complaints redressal cost within 2 months**
- **66% reduction in number of customer complaints; improved customer satisfaction**

ANALYTIXLABS

Business context

- Large installed base of customer's medical equipment across several healthcare centers
- Thousands of customer complaints received each day; shortage of Field Engineers to address them on time
- Ineffective framework to prioritize customer complaints and focus Field Engineers' efforts
- Additional challenge in analysis due to unstructured text format of complaints records

ANALYTIXLABS

What we observed

- Large volume of varied un-structured customer feedback
- Current sentiment analysis methodology ineffective in dealing with variety and scale of data
- Poor customer satisfaction
- High cost to resolve complaints

ANALYTIXLABS

What we did

- Developed a completely different and cutting edge complaint prioritization framework
- Applied sophisticated machine learning techniques such as Random Forest, Support Vector Machine, Neural Networks and Generalized Linear Model

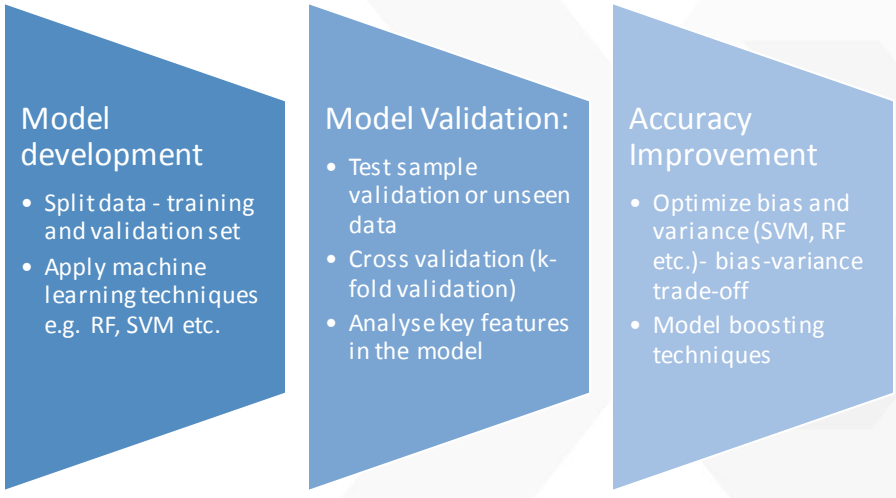
ANALYTIXLABS

Data Preparation and Exploration

Data Size :
~1GB unstructured data
Tools used:
R & Excel

ANALYTIXLABS

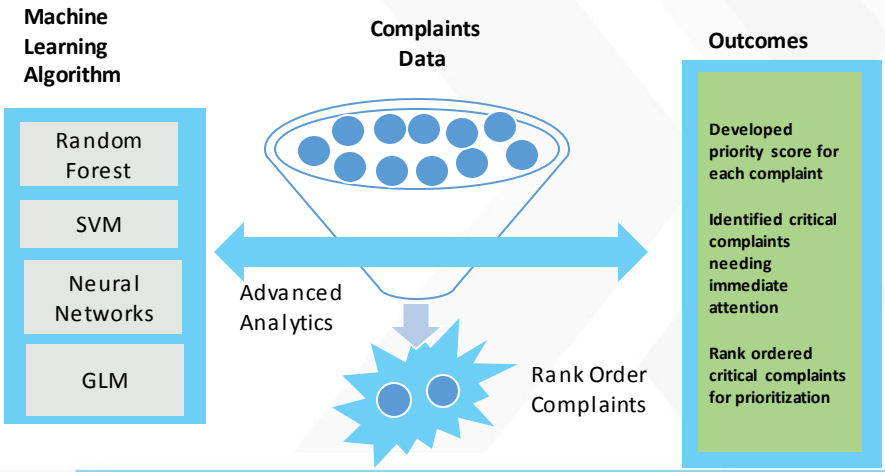
Model Development & Performance Improvement



ANALYTIXLABS

The method

- Machine learning algorithm with multiple techniques deployed

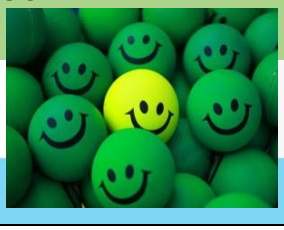


ANALYTIXLABS

Benefits Delivered

- Focused resolution efforts by prioritizing attention on the most critical complaints
- Minimized time & effort on mundane/ less critical concerns
- Thereby reduced the number of Field Engineers employed

- **5% reduction in complaints redressal cost within 2 months**
- **66% reduction in number of customer complaints; improved customer satisfaction**



ANALYTIXLABS

We also prescribed...

- Current model developed based on complaints data for two equipment modalities only
- Replicate the model across other modalities
    - separate the signal from noise given the nature of text complaints
    - focus and gain additional benefit by customizing the model for each equipment modality

ANALYTIXLABS

Appendix

ANALYTIXLABS

Topic Modeling (with more details)

ANALYTIXLABS

Introduction to topic modeling

Original problem: How to synthesize the information in a large collections of documents?

Example:

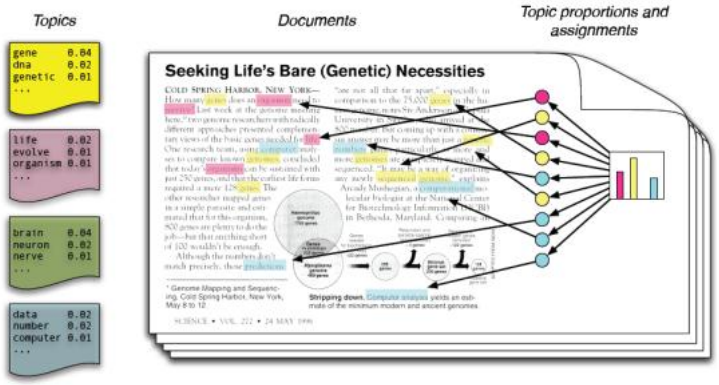
D1: modem the steering linux. modem, linux the modem. steering the modem. linux!
D2: linux; the linux. the linux modem linux. the modem, clutch the modem. petrol.
D3: petrol! clutch the steering, steering, linux. the steering clutch petrol. clutch the petrol; the clutch.
D4: the the the. clutch clutch clutch! steering petrol; steering petrol petrol; steering petrol!!!!

Typically done using some clustering algorithm to find groups of similar documents.

ANALYTIXLABS

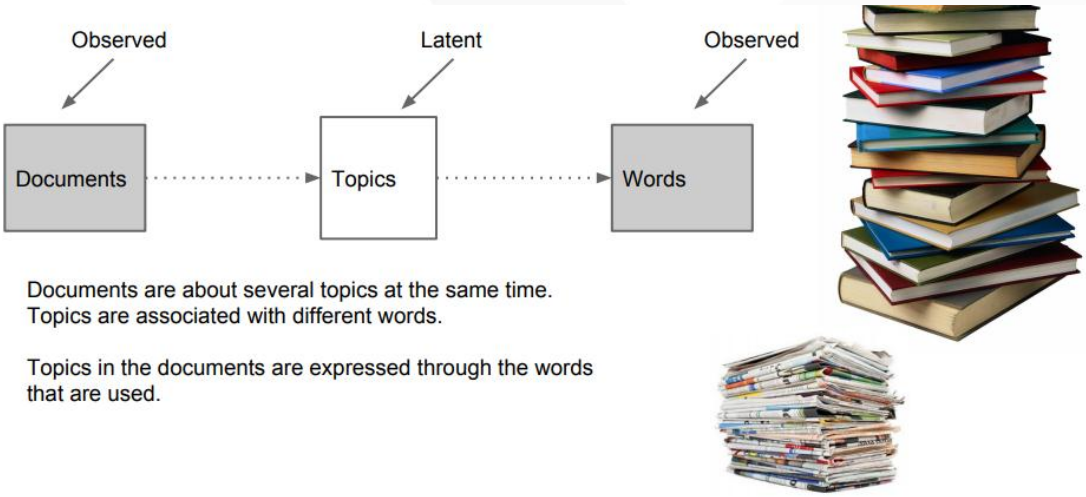
Alternative is Topic Modeling

- Find several latent topics of themes that are "link" between the documents and words.
- Topics explain why certain words appear in a given document.
- Preprocessing is very much the same for all topic modeling methods (LSA, pLSA, LDA)



ANALYTIXLABS

Goal of Topic Modeling

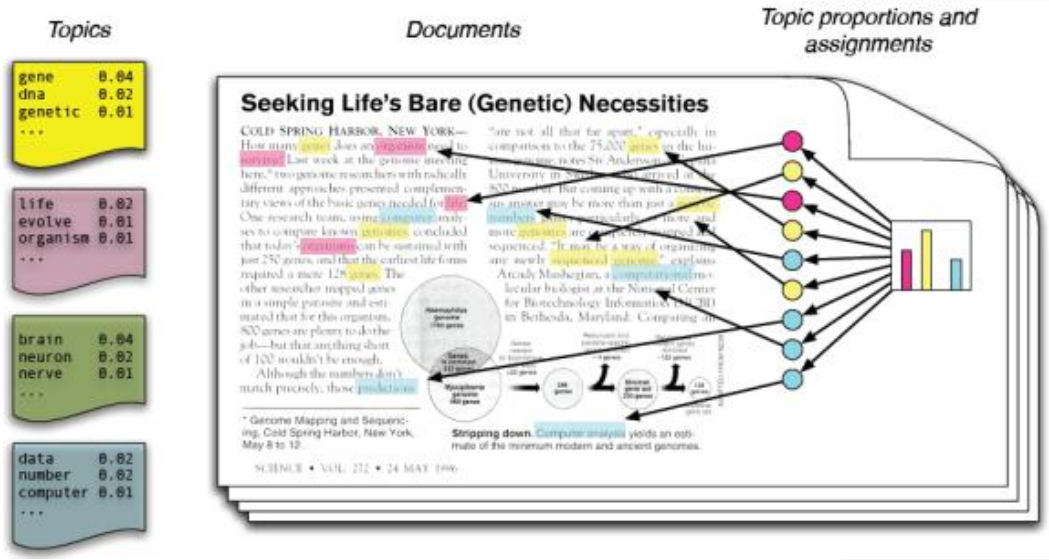


Documents are about several topics at the same time.
Topics are associated with different words.

Topics in the documents are expressed through the words that are used.

ANALYTIXLABS

Goal of Topic Modeling



ANALYTIXLABS

Example of Topic Modeling: New York times articles

LDA analysis of 1.8M New York Times articles:

music band songs rock album jazz pop song singer night	book life novel story books man stories love children family	art museum show exhibition artist artists paintings painting century works	game knicks nets points team season play games night coach	show film television movie series says life man character know
theater play production show stage street broadway director musical directed	clinton bush campaign gore political republican dole presidential senator house	stock market percent fund investors funds companies stocks investment trading	restaurant sauce menu food dishes street dining dinner chicken served	budget tax governor county mayor billion taxes plan legislature fiscal

ANALYTIXLABS

Example: New York times articles (Topics and explanation)

Topics	Explanation
black* price* worth* white* goods* yard* silk* made* lot* week ladies wool* inch* ladles* sale* prices* pair* suits* fine*	Reflects discussion of the market and sales of goods, with some words that relate to cotton and others that reflect other goods being sold alongside cotton (such as wool).
state* people* states* bill* law* made united* party* men* country* government* county* public* president* money* committee* general* great question*	Political language associated with the political debates that dominated much of newspaper content during this era. The association of the topic "money" is particularly telling, as economic and fiscal policy were particularly important discussion during the era.
clio worth mid city alie fort lino law lour lug thou hut fur court dally county anil tort iron	Noise and words with no clear association with one another.
tin inn mid tint mill* till oil* ills hit hint lull win hut ilia til ion lot lii foi	Mostly noise, with a few words associated with cotton milling and cotton seed.
texas* street* address* good wanted houston* office* work city* sale main* house* apply man county* avenue* room* rooms* land*	These topics appear to reflect geography. The inclusion of Houston may either reflect the city's importance as a cotton market or (more likely) the large number of newspapers from the collection that came from Houston.
worth* city* fort* texas* county* gazette tex* company* dallas* miss special yesterday night time john state made today louis*	These topics appear to reflect geography in north Texas, likely in relation to Fort Worth and Dallas (which appear as topics) and probably as a reflection that a large portion of the corpus of the collection came from the Dallas/Ft. Worth area.
houston* texas* today city* company post* hero*	These topics appear to an unlikely subject identified by

ANALYTIXLABS

Example: New York times articles

Period	Topics	Explanation
1865-1901	texas* city* worth* houston* good* county* fort* state* man* time* made* street* men* work* york today company great people	These keywords appear to be related to three things: (1) geography (reflected in both specific places like `Houston` and `Fort Worth` and more general places like `county`, `street`, and `city`), (2) discussions of people (`men` and `man`) and (3) time (`time` and `today`).
1892	texas* worth* gazette* city* tex* fort* county* state* good* march* man* special* made* people* time* york men days feb	As with the 1865-1901 set, these keywords also appear to be related to three things: (1) geography, (2) discussions of people and (3) time.
1893	worth* texas* tin* city* tube* clio* time* alie* man* good* fort* work* made street year men county state tex	As with the 1865-1901 set, these keywords also appear to be related to three things: (1) geography, (2) discussions of people and (3) time.
1929-1930	tin* texas* today* county* year* school* good* time* home* city* oil* man* men* made* work* phone night week sunday	As with the 1865-1901 set, these keywords also appear to be related to three things: (1) geography, (2) discussions of people and (3) time. The time discussion here appears to be heightened, and the appearance of economic issues for Texas (`oil`) makes sense in the context of the onset of the Great Depression in 1929-30.

Table 7: Main topics for years of interest for the main set

ANALYTIXLABS

Types of topic modeling

Vector-based techniques:

- Latent Semantic Analysis (LSA) (a.k.a Latent Semantic Indexing - LSI): Finding smaller (lower-rank) matrices that closely approximate DTM

Probabilistic techniques

- **Probabilistic Latent Semantic Analysis (pLSA):** Finding topic-word and topic-document associations that best match dataset and specified number of topics K
- **Latent Dirichlet Allocation (LDA)** Finding topic-word and topic-document associations that best match dataset and specified number of topics that come from Dirichlet distribution with given dirichlet priors.
  - Whole bunch of LDA extensions

ANALYTIXLABS

Software for topic modeling

R packages

a. lsa package
b. lda package
c. topicmodels package
d. stm package

Python libraries

a. Gensim - Topic Modelling for Humans
b. LDA Python library

ANALYTIXLABS

Steps for Topic modeling

Data Preprocessing: Remove all non-alphanumerical characters etc.
Tokenization: Find all the words exists in the corpus. That defines our vocabulary
Vectorization: Create document term metrics (count, tf-idf, cosine similarity etc…)
Topic Modeling: Using different algorithms like LSA, pLSA, LDA etc.

ANALYTIXLABS

Steps for Topic modeling

Example:

D1: modem the steering linux. modem, linux the modem. steering the modem. linux!
D2: linux; the linux. the linux modem linux. the modem, clutch the modem. petrol.
D3: petrol! clutch the steering, steering, linux. the steering clutch petrol. clutch the petrol; the clutch.
D4: the the the. clutch clutch clutch! steering petrol; steering petrol petrol; steering petrol!!!!

	D1	D2	D3	D4
linux	3	4	1	0
modem	4	3	0	1
the	3	4	4	3
clutch	0	1	4	3
steering	2	0	3	3
petrol	0	1	3	4

$$\mathbf{t}_i^T \rightarrow \begin{bmatrix} x_{1,1} & \dots & x_{1,n} \\ \vdots & \ddots & \vdots \\ x_{m,1} & \dots & x_{m,n} \end{bmatrix} \quad \mathbf{d}_j \downarrow$$

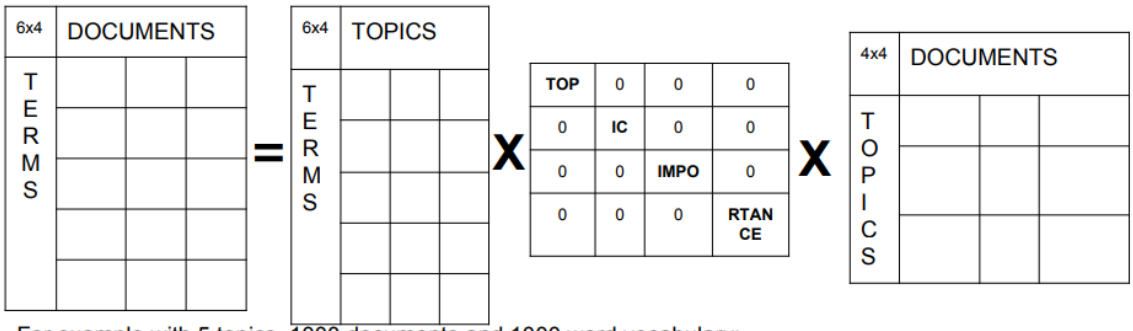
Can we somehow make matrices smaller?
Two topics => two rows in matrix

ANALYTIXLABS

Latent Semantic Analysis (LSA)

Nothing more than a **singular value decomposition (SVD)** of document-term matrix:

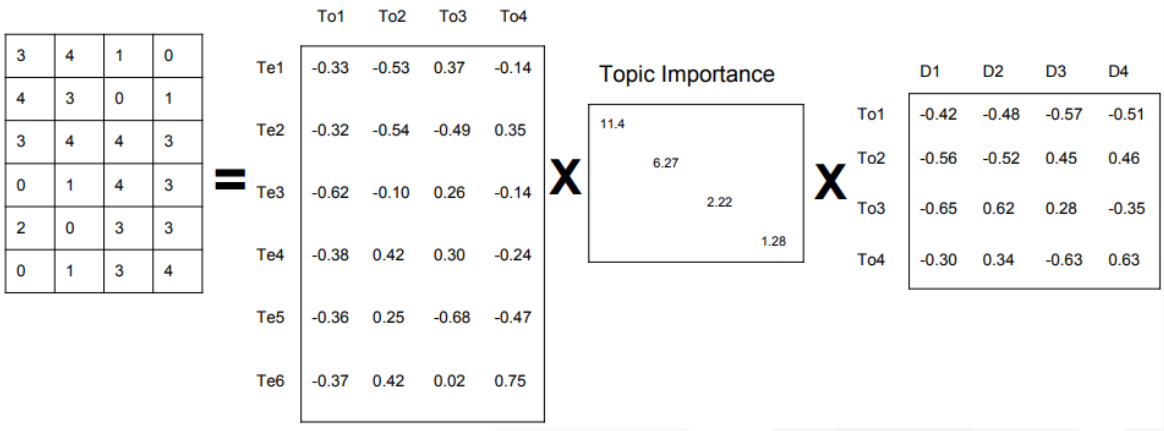
Find three matrices U, Σ and V so that: $X = U\Sigma V^t$



For example with 5 topics, 1000 documents and 1000 word vocabulary:
Original matrix: 1000 x 1000 = 10^6
LSA representation: 5x1000 + 5 + 5x1000 ~ 10^4
-> 100 times less space!

ANALYTIXLABS

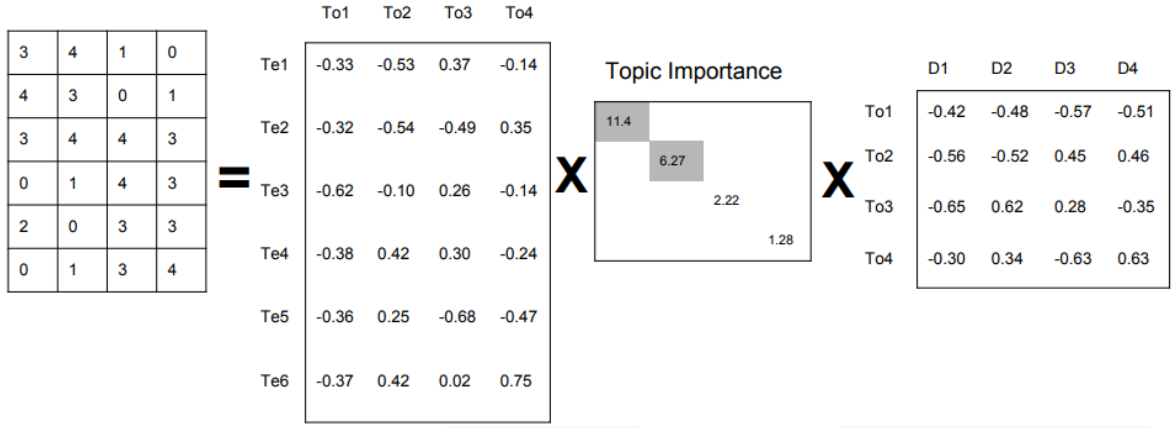
LSA – Our Example



ANALYTIXLABS

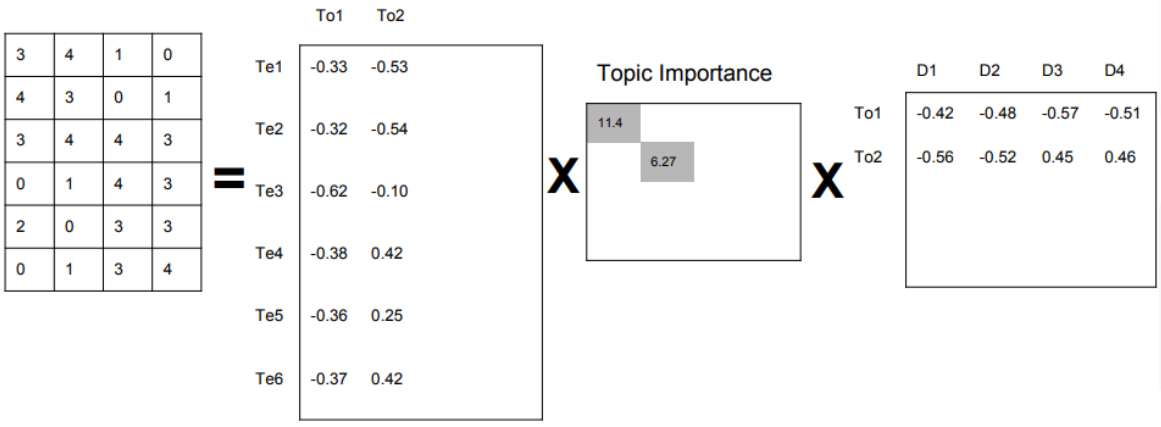
LSA – Our Example

First two topics much more important than other two topics!



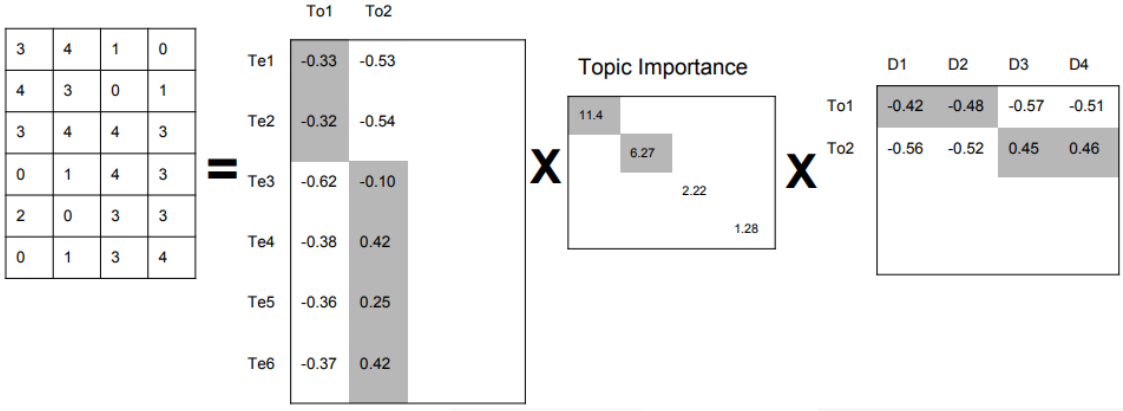
LSA – Our Example

We can drop columns/rows for the topics we are not interested.



LSA: Our Example

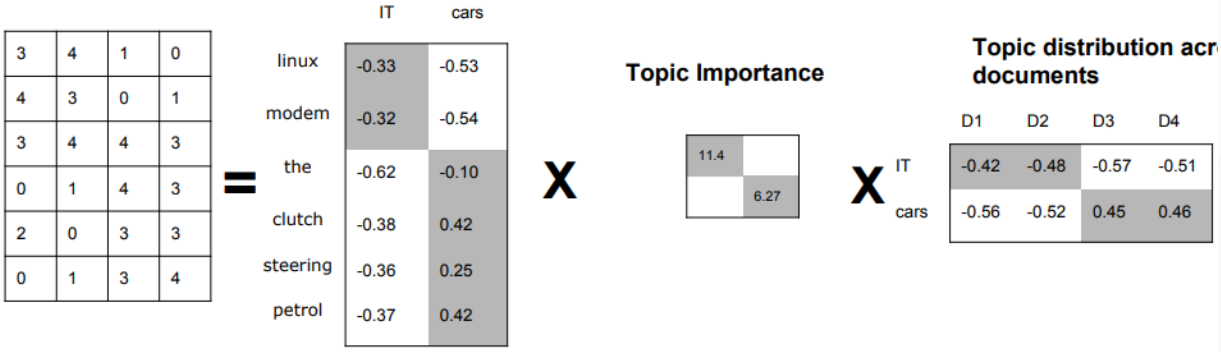
Pick highest assignments for each word to topic, and each topic to document



LSA: Our Example

LSA is essentially low-rank *approximation* of document term-matrix

Word assignment to topics



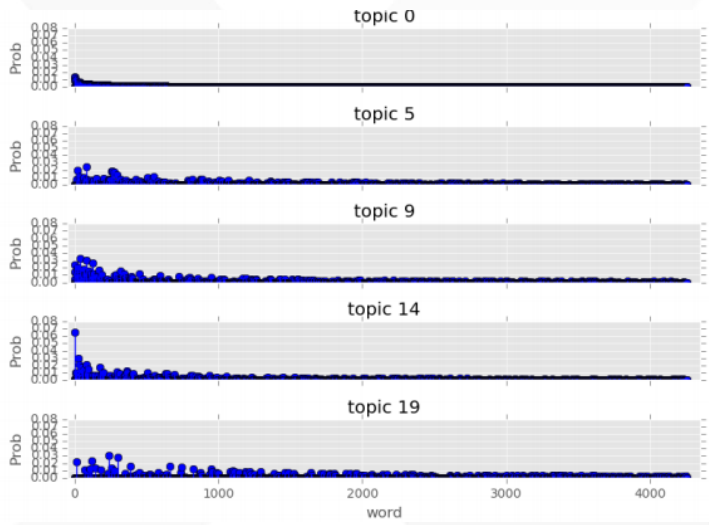
Probabilistic Topic Modeling

What is a topic?

A list of probabilities for each of the possible words in a vocabulary.

Example topic:

- dog: 5%
- cat: 5%
- hause: 3%
- hamster: 2%
- turtle: 1%
- calculus: 0.000001%
- analytics: 0.000001%
- .....
- .....
- .....



ANALYTIXLABS

Probabilistic Topic Modeling

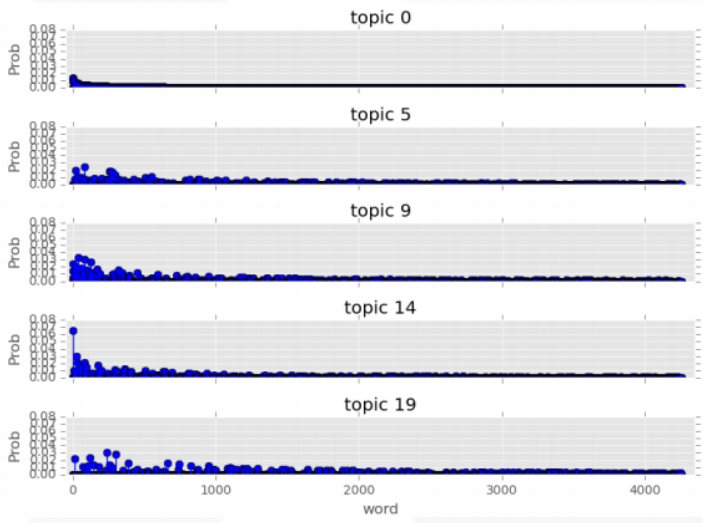
What is a topic?

A list of probabilities for each of the possible words in a given language.

Example topic:

Pets

- dog: 5%
- cat: 5%
- hause: 3%
- hamster: 2%
- turtle: 1%
- calculus: 0.000001%
- analytics: 0.000001%
- .....
- .....
- .....

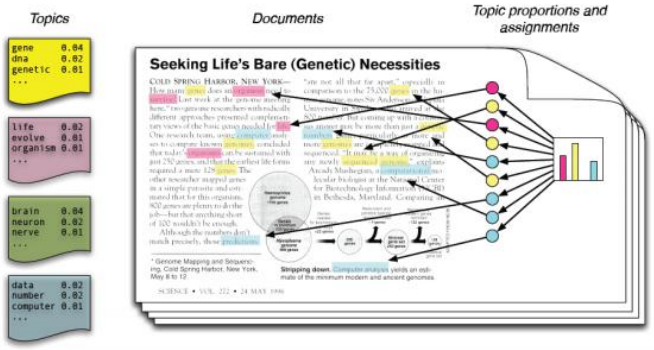


ANALYTIXLABS

Probabilistic spin to LSA

Instead of finding lower-ranked matrix representation, we can try to find a mixture of word->topic & topic->documents distributions that are most likely given the observed documents.

- We define a statistical model of how the documents are being made (generated).
  - This is called generative process in topic modeling terminology.
- Then we try to find parameters of that model that best fit the observed data.

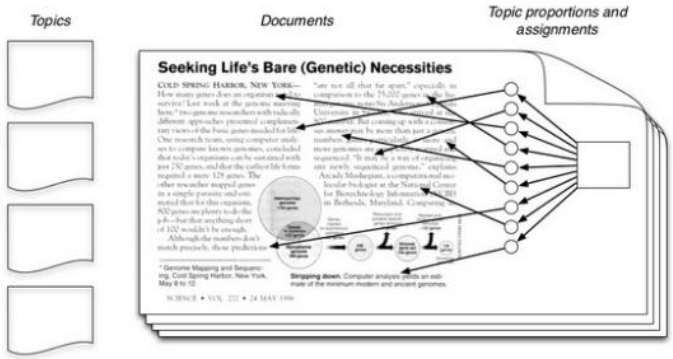


ANALYTIXLABS

Probabilistic spin to LSA

Instead of finding lower-ranked matrix representation, we can try to find a mixture of word->topic & topic->documents distributions that are most likely given the observed documents.

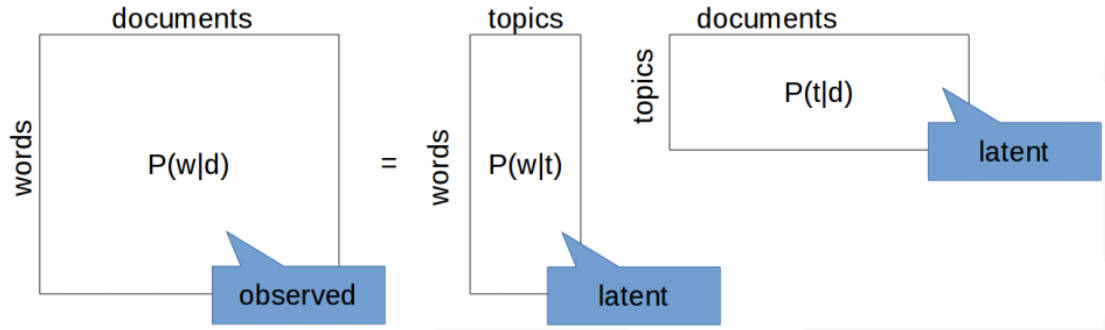
- We define a statistical model of how the documents are being made (generated).
  - This is called generative process in topic modeling terminology.
- Then we try to find parameters of that model that best fit the observed data.



ANALYTIXLABS

Probabilistic spin to LSA

$$P(w|d) = \sum_t P(t|d)\,P(w|t)$$



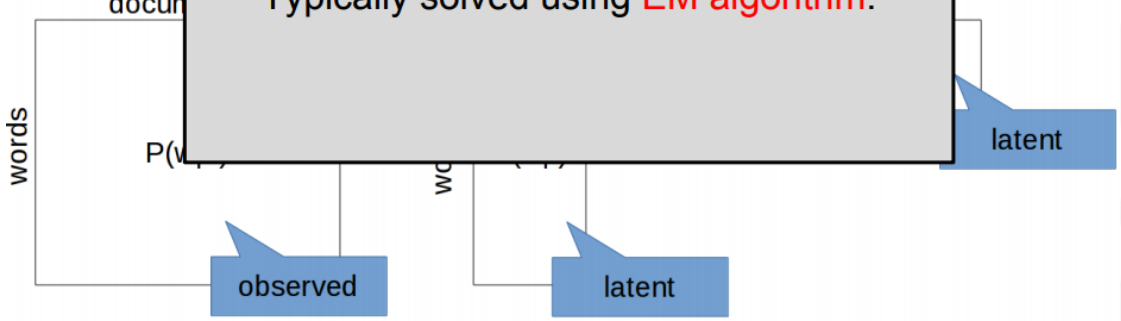
ANALYTIXLABS

Probabilistic spin to LSA

Another more useful formulation: $P(w|d) = \sum_t P(t|d)\,P(w|t)$

Find most likely values for P(t|d) and P(w|t).

Typically solved using EM algorithm.



EM Algorithm: An **expectation–maximization** (**EM**)**algorithm** is an iterative method to find maximum likelihood or maximum a posteriori (MAP) estimates of parameters in statistical models, where the model depends on unobserved latent variables.

ANALYTIXLABS

pLSA Challenges

- Tries to find topic distributions and word distributions that best fit the data
  - Overfitting
  - Some documents have strong associations to only couple of topics, while others have more evenly distributed associations.

LDA: An extension to pLSA

In LDA, we *encode our assumptions about the data.*

Two important assumptions:

1. On average, how many topics are per document?
   a. Few or more?
2. On average, how are words distributed across topics?
   a. Are topics strongly associated with few words or not?

Those assumptions are defined by two vectors α and β:

α: K dimensional vector that defines **how K topics are distributed across documents**.
Smaller αs favor fewer topics strongly associated with each document.

β: V dimensional vector that defines **how V words are associated across topics**.
Smaller βs favor fewer words strongly associated with each topics.

IMPORTANT: Typically, all elements in α and β are the same (in `topicmodels` library 0.1 is default)
Uninformative prior: We don't know what are the prior distributions so we will say all are equally likely.

Those assumptions are then used to generate "dices" (topic-word associations, topic-document associations)

Dirichlet Distribution

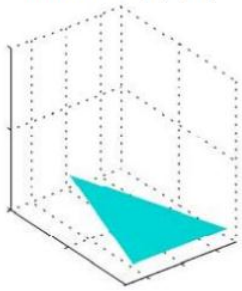
- Multivariate distribution parametrized by a N-dimensional vector.
- *Drawing a sample from a dirichlet distribution parametrized with N-dimensional vector returns an N-dimensional vector that sums to one.*

N=3:

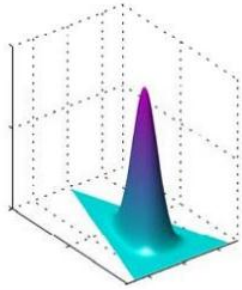
Bigger than 1

Less than 1

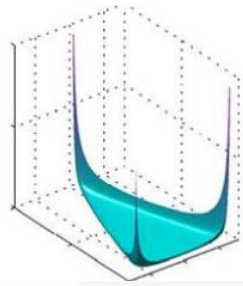
Params = [1, 1, 1]



Params = [10, 10, 10]



Params = [.1, .1, .1]



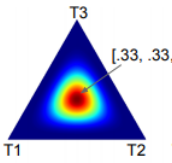
ANALYTIXLABS

Dirichlet Distribution

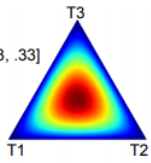
- Multivariate distribution parametrized by a N-dimensional vector.
- *Drawing a sample from a dirichlet distribution parametrized with N-dimensional vector returns an N-dimensional vector that sums to one.*

N=3:

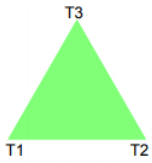
Params = [2, 2, 2]



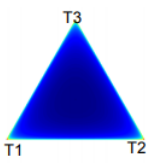
Params = [5, 5, 5]



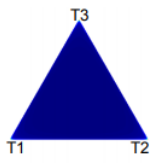
Params = [1,1,1]



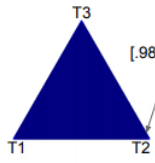
Params = [.9, .9,0.9]



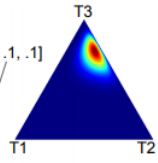
Params = [.5, .5, .5]



Params = [.1, .1, .1]



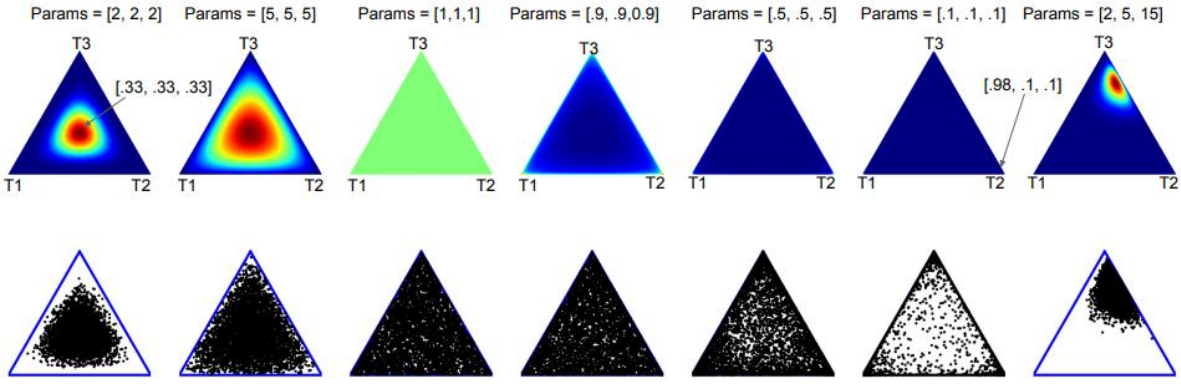
Params = [2, 5, 15]



ANALYTIXLABS

Dirichlet Distribution

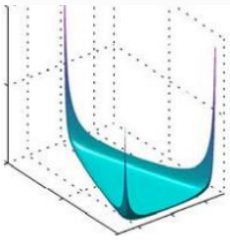
How topics are distributed across documents and how words are distributed across topics are drawn from Dirichlet distribution!



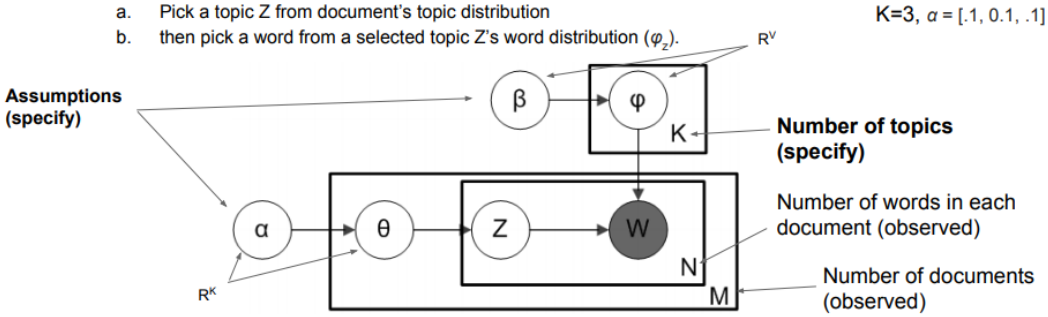
LDA: Generative process outline

How we imagine that documents were generated:

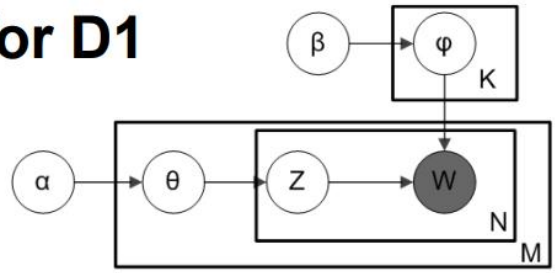
1. We specify number of topics K and our assumptions $\alpha \in R^K$ and $\beta \in R^V$ that capture general associations between document-topic and topic-word relationships.
2. For each document we pick one sample from a Dirichlet distribution (defined by α) that defines document's distribution over topics ($\theta \in R^K$).
3. For each topic we pick one sample from a Dirichlet distribution (defined by β) that defines topic's distribution over available words ($\varphi \in R^V$).
4. For each position in each document:
   a. Pick a topic Z from document's topic distribution
   b. then pick a word from a selected topic Z's word distribution (φ_z).



K=3, α = [.1, 0.1, .1]



Generative process for D1



D1: modem the steering linux modem linux the modem steering the modem linux

Our vocabulary is 6 words in total (V=6):
linux, modem, the, clutch, steering, petrol

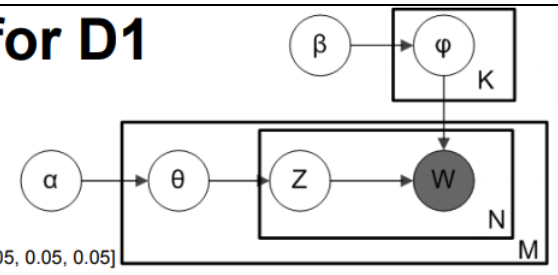
We decide to use three topics (K=3). We call them IT, cars, and pets.

As we have three topics α has three elements. We set α to [.1, .1, .1].

As we have six words, β has six elements. We set β to [.1, .1, .1, .1, .1, .1].

ANALYTIXLABS

Generative process for D1

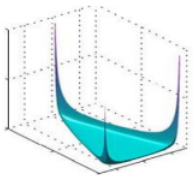


D1: modem the steering linux modem linux the
modem steering the modem linux

K=3, V=6, $\alpha \in R^3$ = [0.1, 0.1, 0.1],
$\beta \in R^6$ = [0.1, 0.1, 0.1, 0.1, 0.1, 0.1],

From *Dirichlet*(β) we sample 3 times, once for each topic:

1. For topic IT we pick φ_{IT}. It happened to be = [0.3, 0.5, 0.05, 0.05, 0.05, 0.05]
2. For topic cars we pick φ_{cars}. It happened to be = [0.1, 0.05, 0.3, 0.15, 0.2, 0.2]
3. For topic pets we pick φ_{pets}. It happened to be = [0.05, 0.1, 0.1, 0.20, 0.3, 0.25]



From *Dirichlet*(α) we sample M times, once for every document:

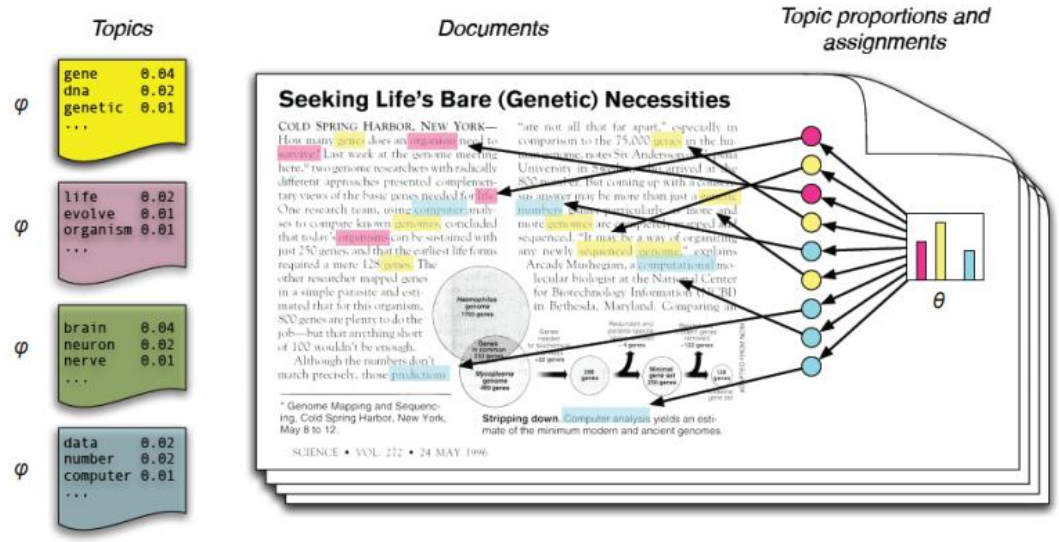
4. For document D1, we pick θ_{D1}. It happened to be [0.9, 0.05, 0.05]
5. *For document D2, we pick θ_{D2}. It happened to be .....*

   .....

6. For first position in document D1, we pick a topic from *Multinomial*([0.9, 0.05, 0.05]). Say it is IT.
   As the topic for the first position is IT, we pick a word from *Multinomial*([0.3, 0.5, 0.05, 0.05, 0.05, 0.05]).
   Oh look, it is 'modem'.
7. For second position in document D1, we pick a topic based on *Multinomial*(θ_{D1}). Say it is cars.
   As the topic for the second position is cars, we pick a word from *Multinomial*(φ_{cars}). Oh look, it is 'the'.
8. .....

ANALYTIXLABS

Generative process of LDA



ANALYTIXLABS

topicmodels R Library

- "Interface" to `lda` library. Provides less functionality, but is easier to use.

- Analysis steps:
  a. reading data
  b. data cleanup (stopwords, numbers, punctuation, short words, stemming, lemmatization)
  c. building document term matrix
  d. filtering document term matrix based on TF and TF-IDF thresholds
     - TF: we don't want words that appear only in few documents, they are useless for topic description
     - TF-IDF: we don't want words that are appear a lot, but are too general (e.g., 'have')
  e. Either select K upfront, or evaluate several values of K for best fit.
     - Plot evaluation of different values of K
  f. Show top words for each topic
  g. Additional analyses (covariates etc)

ANALYTIXLABS

topicmodels R Library

```
library(topicmodels)
data <- load files into memory
corpus <- Corpus(VectorSource(data));
preprocessing.params <- list(stemming = T,
                             stopwords = T,
                             minWordLength = 3,
                             removeNumbers = T,
                             removePunctuation = T)
dtm <- DocumentTermMatrix(corpus, control = preprocessing.params)
dtm <- filter noise from DTM (TF, TF-IDF)
lda.params <- list(seed=1, iter= 2000)
LDA(dtm, k = 20, method = "Gibbs", control = lda.params)
plot graphics
do further analysis....
```

ANALYTIXLABS

topicmodels R Library

- Problem of picking number of topics K
  - Perplexity on the left-out test set
  - Harmonic mean evaluation
- A bit more coding for:
  - Filtering of words
  - Plotting perplexity scores requires coding
  - Calculating frequencies of each topic requires a bit of coding
  - Finding most likely topics for each document requires

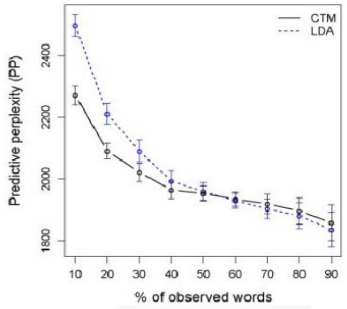
ANALYTIXLABS

Advanced Topic Modeling

ANALYTIXLABS

Correlated Topics Model

- CTM allows the topics to be correlated
- CTM allows for better prediction
  - Logistic normal instead of Dirichlet distribution
- More robust to overfitting



- Package "topicmodels"
  - function CTM

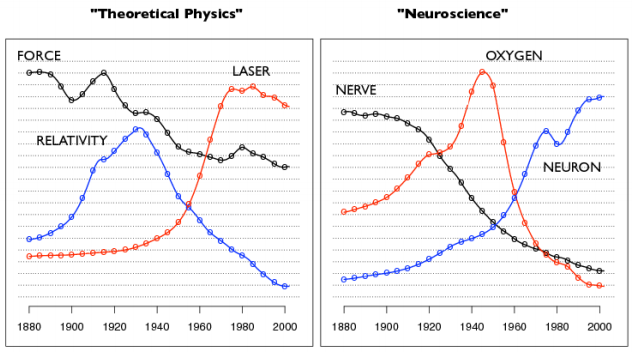
CTM(x, k, method = "VEM", control = NULL, model = NULL, ...)

- Arguments
  - x - DocumentTermMatrix object
  - k - Integer, number of topics
  - method - currently only "VEM" supported

ANALYTIXLABS

Dynamic Topic Model

- DTM models how each individual topic changes over time



ANALYTIXLABS

Supervised LDA (sLDA)

- Each document is associated with an external variable
  - for example, a movie review could be associated with a numerical ratings.
- Defines a one-to-one correspondence between latent topics and user tags
- Allows sLDA to directly learn word-tag correspondences.

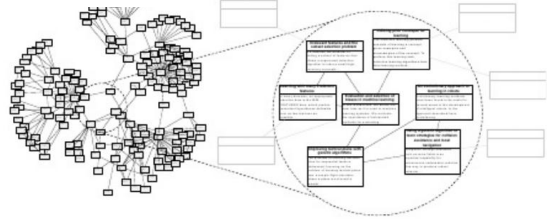
- Package "`lda`"

```
slda.em(documents,
        K,
        vocab,
        num.e.iterations,
        num.m.iterations,
        alpha,
        eta,
        annotations,
        params,
        variance,
        logistic = FALSE,
        lambda = 10,
        regularise = FALSE,
        method = "sLDA",
        trace = 0L)
```

ANALYTIXLABS

Relational Topic Models

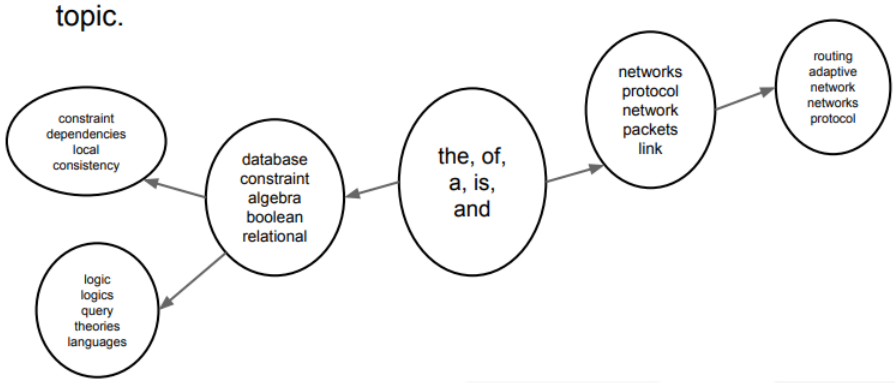
- Given a new document RTM predicts which documents it is likely to be linked to
- For example - tracking activities on Facebook in order to predict a reaction on an advertisement
- RTM is also good for certain types of data that have spatial/geographic dependencies



ANALYTIXLABS

Hierarchical Topic Modeling

- LDA fails to draw the relationship between one topic and another.
- CTM considers the relationship but fail to indicate the level of abstract of a topic.



ANALYTIXLABS

Structural Topic Modeling (STM)

- STM allows for the inclusion of covariates of interest.
- STM vs. LDA
  - topics can be correlated
  - each document has its own prior distribution over topics, defined by covariate X rather than sharing a global mean
  - word use within a topic can vary by covariate *U*

- The STM provides fast, transparent, replicable analyses that require few a priori assumptions about the texts under study

ANALYTIXLABS

Contact Us

Visit us on: http://www.analytixlabs.in/

For more information, please contact us: http://www.analytixlabs.co.in/contact-us/
Or email: info@analytixlabs.co.in

Call us we would love to speak with you: (+91) 99105-09849

Join us on:
Twitter - http://twitter.com/#!/AnalytixLabs
Facebook - http://www.facebook.com/analytixlabs
LinkedIn - http://www.linkedin.com/in/analytixlabs
Blog - http://www.analytixlabs.co.in/category/blog/

ANALYTIXLABS