CSCI446/946 Big Data Analytics

Advanced Analytical Theory and Methods: Text Analysis

School of Computing and Information Technology
University of Wollongong Australia

Recap: Association Rules

- Association rule discovery:
  - An unsupervised learning method
  - Descriptive, not predictive
  - Discover interesting, hidden relationship
    - Represented as rules or frequent itemsets
  - Commonly used for the analysis of transactions

Text Analysis

- Some Overview
- Collecting and Representing Text
  - Term Frequency (TF)
  - Inverse Document Frequency (IDF)
- Categorizing Documents by Topics
- Determining Sentiments
- Gaining Insights

Figures, tables, codes, examples are from the book "Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data".

Overview of Text Analysis

- Text analysis (text analytics)
  - Refers to the representation, processing, and modelling of textual data to derive useful insights.
  - Suffers from the curse of high dimensionality.
  - Most of the time the text is not structured.
- Corpus
  - A collection of texts (documents) used for various purposes in Natural Language Processing.

Overview of Text Analysis

Corpus	Word Count	Domain	Website
Shakespeare	0.88 million	Written	http://shakespeare.mit.edu/
Brown Corpus	1 million	Written	http://icame.uib.no/brown/bcm.html
Penn Treebank	1 million	Newswire	http://www.cis.upenn.edu/~treebank/
Switchboard Phone Conversations	3 million	Spoken	http://catalog.ldc.upenn.edu/LDC97S62
British National Corpus	100 million	Written and spoken	http://www.natcorp.ox.ac.uk/
NA News Corpus	350 million	Newswire	http://catalog.ldc.upenn.edu/LDC95T21
European Parliament Proceedings Parallel Corpus	600 million	Legal	http://www.statmt.org/europarl/
Google N-Grams Corpus	1 trillion	Written	http://catalog.ldc.upenn.edu/LDC2006T13

Sources of Text

Examples:

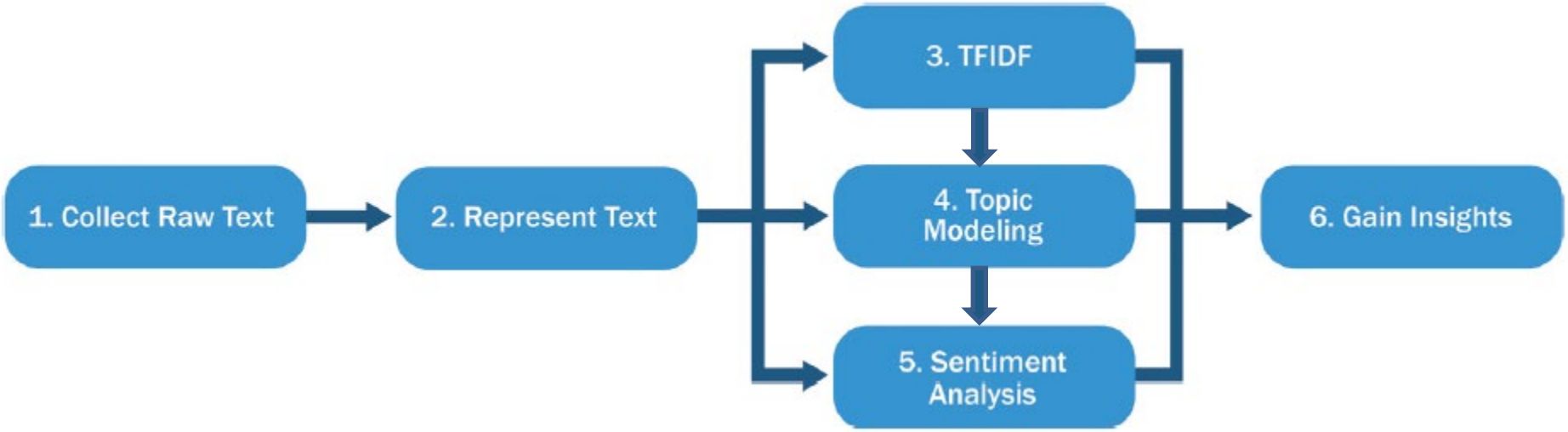
Data Source	Data Format	Data Structure Type
Articles	TXT, HTML, PDF, scanned PDF	Unstructured
Literature	TXT, DOC, HTML, PDF	Unstructured
E-mail	TXT, MSG, EML	Unstructured
Web pages	HTML	Semi-structured
Server logs	LOG, TXT	Semi-structured or Quasi-structured
Social network API	XML, JSON, RSS	Semi-structured
Call center transcripts	TXT	Unstructured
Voice recognition software	TXT	Unstructured

Text Analysis Steps

- Text mining
  - Clustering and classification techniques can be adapted to text mining. For example:
    - Cluster documents into groups.
    - Classify texts for sentiment analysis.
  - Utilises various methods and techniques
    - Statistical analysis.
    - Information retrieval.
    - Natural Language Processing.

A Text Analysis Example

- A company would like to monitor what is being said about its products in social media:
  - Are people mentioning its products?
  - What is being said? Good or bad?



Challenges

- Semantics vs. Syntax vs. Pragmatics
  - Syntax concerns the sentence structure and the rules of grammar. E.g.:
    - "The dog chased a rabbit through the pasture.“ vs "The through pasture the chased a dog rabbit.“
  - Semantics is the study of the meaning of sentences.
  - Pragmatics concerns the meaning of sentences in a certain context.
    - "Break it down.“ can mean knocking over a building, or may be a call to share a business-related concept.

Challenges

- Homonyms vs. acronyms.
  - Homonyms are words that have the same spelling but have different meanings. E.g.:
    - dog *bark* vs. tree *bark*. I *left* my phone on the *left* side of the room. *Amazon* (river, store, rainforest).
  - Acronyms are abbreviated versions of words.
    - CGI (Common Gateway Interface vs Computer Graphics Interface). Meaning of "TSIG"?
- Disambiguation narrows down the meaning of words or acronyms.

Challenges

- To a computer:
  - Text is merely a sequence of characters encoded as numbers.
    - Has no understanding of syntax.
    - Has no understanding of semantic meanings.
    - Has no understanding of pragmatic meaning.
  - In raw form there is no “natural” similarity metric between words or texts.
    - Cannot perform clustering nor classification.
  - There is thus a need to represent and process text in a form suitable for clustering or classification.

First step: Collecting Raw Text

- For text analysis, data must be collected before anything can happen. Example:
  - Start by actively monitoring various websites for user-generated contents.
    - Use public APIs, Web scraper/crawler,...
  - Expect to deal with unstructured or semi-structured data.
- Be careful about the rights of the owner.

Representing Text

- Raw text needs to be transformed with text normalization techniques.
- Tokenization
  - The task of separating words from the body of text.
  - Tokenizing based on spaces.
    - “day” vs “day.”
  - Tokenizing based on punctuation marks & spaces.
    - we’ll, state-of-the-art.
  - Often more difficult than expected.
    - Back of Bourke, résumé vs. resume.
  - No one-size-fits-all tokenization scheme.

Text Normalization

- Case folding
  - Reduces all letters to lowercase (or uppercase)
  - If implemented incorrectly…
    - General Motors; WHO; US; …
  - May need to create a lookup table of words not to be case folded.

Text Normalization

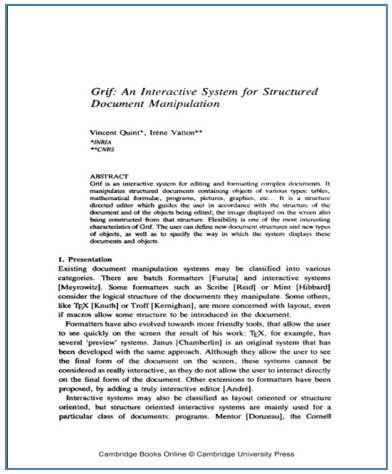
- Stop words
  - Not all the words from a given language may need to be considered.
    - "the, a, of, and, to, …" which are not likely to contribute to semantic understanding.
- Lemmatization and stemming.
  - Walk: walking, walk, walks, walked,… (stemmed)
  - Goose: geese, goose, gander, ganders (lemmatized)
  - Good: good, better (lemmatized)
  - Most popular are "Porter stemmer", "WordNet" lemmatizer.

Text Normalization

- Bag-of-words representation.
  - Simple yet widely used to represent text.
  - Represent a document as a set of terms (words), ignoring other information (such as order, context, inferences, and semantics)
    - “a dog bites a man” same as “a man bites a dog”
  - A naïve and over-simplified approach but is still considered a good approach to start with.

Representing Text

- Bag-of-words representation
  - A document becomes a high-dimensional vector, indicating the presence/absence/frequency of various words in this document.



Grif: An Interactive System for Structured Document Manipulation

Vincent Quint*, Irène Vatton**
*INRIA
**CNRS

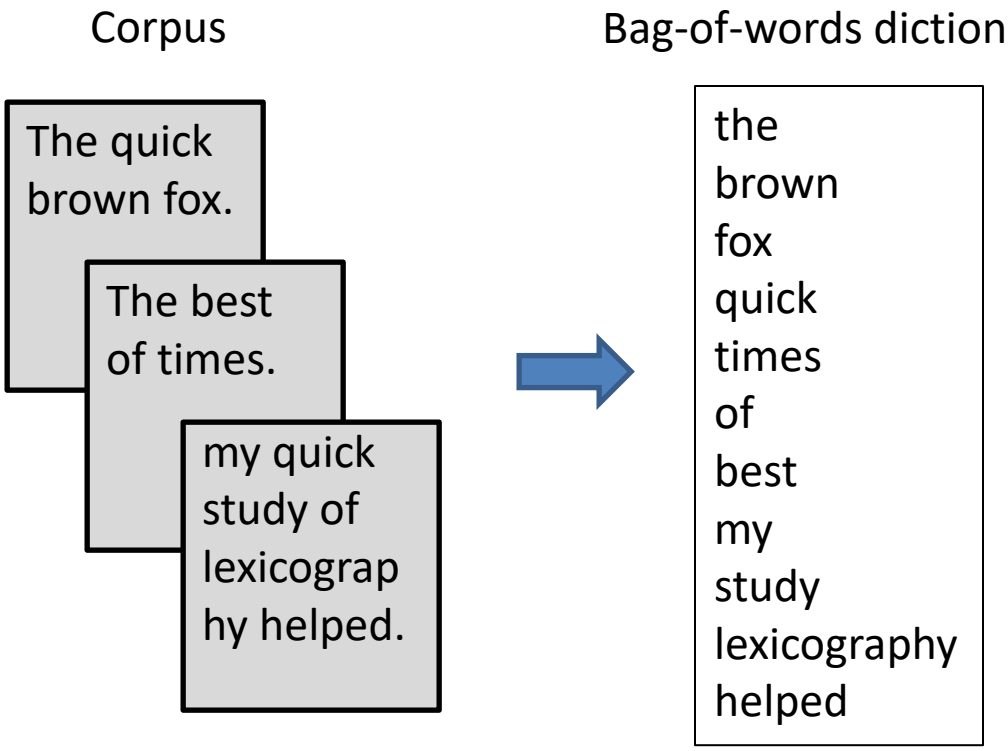
ABSTRACT
Grif is an interactive system for editing and formatting complex documents. It manipulates structured documents containing objects of various types: tables, mathematical formulae, programs, pictures, graphics, etc. It is a structure directed editor which guides the user in accordance with the structure of the document and of the objects being edited; the image displayed on the screen also being constructed from that structure. Flexibility is one of the most interesting characteristics of Grif. The user can define new document structures and new types of objects, as well as to specify the way in which the system displays these documents and objects.

1. Presentation
Existing document manipulation systems may be classified into various categories. There are batch formatters [Furuta] and interactive systems [Meyrowitz]. Some formatters such as Scribe [Reid] or Mint [Hibbard] consider the logical structure of the documents they manipulate. Some others, like TEX [Knuth] or Troff [Kernighan], are more concerned with layout, even if macros allow some structure to be introduced in the document.
Formatters have also evolved towards more friendly tools, that allow the user to see quickly on the screen the result of his work: TEX, for example, has several 'preview' systems. Janus [Chamberlin] is an original system that has been developed with the same approach. Although they allow the user to see the final form of the document on the screen, these systems cannot be considered as really interactive, as they do not allow the user to interact directly on the final form of the document. Other extensions to formatters have been proposed, by adding a truly interactive editor [André].
Interactive systems may also be classified as layout oriented or structure oriented, but structure oriented interactive systems are mainly used for a particular class of documents: programs. Mentor [Donzeau], the Cornell

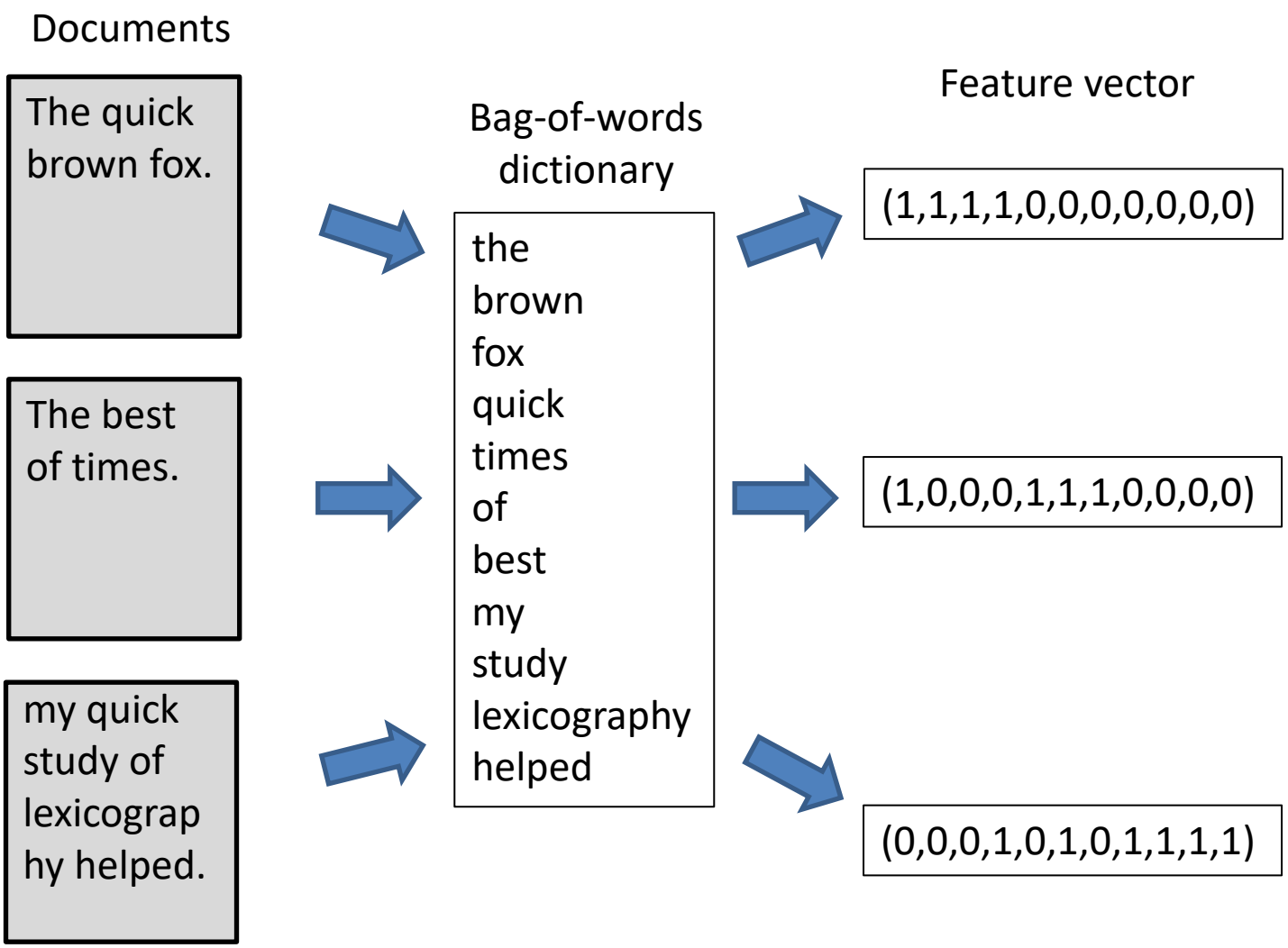
Cambridge Books Online © Cambridge University Press

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \dots \\ x_n \end{bmatrix}$$

Representing Text by BoW



Presenting Text by BoW



Representing Text

- Representation of a corpus
  - A corpus is a collection of documents.
  - Some corpora include the information content of every word in its metadata.
- Information content (IC)
  - A metric denotes the importance of a term in a corpus.
  - Terms with higher IC values are more important.

Representing Text

- However, information content (IC)
  - Cannot satisfy the need to analyse dynamically changing, unstructured data.
- Two problems
  - Both traditional corpora and IC metadata do not change over time.
  - Traditional corpora limits the knowledge used for a text analysis algorithm to what is covered in the corpus.
    - New topics and concepts would not be recognized.

Term Frequency – Inverse Document Frequency (TFIDF)

- We need a metric that adapts to the context and the nature of text (not like IC).
- TFIDF is based entirely on all the fetched documents.
- TFIDF can be easily updated once the fetched documents change.
- TFIDF is a measure widely used in text analysis.

Term Frequency

- Given a term *t* and a document $d = \{t_1, t_2, \ldots, t_n\}$
- Term frequency of *t* in *d* is defined as the number of times *t* appears in *d*.

$$TF_1(t,d) = \sum_{i=1}^{n} f(t,t_i) \qquad t_i \in d; |d| = n$$

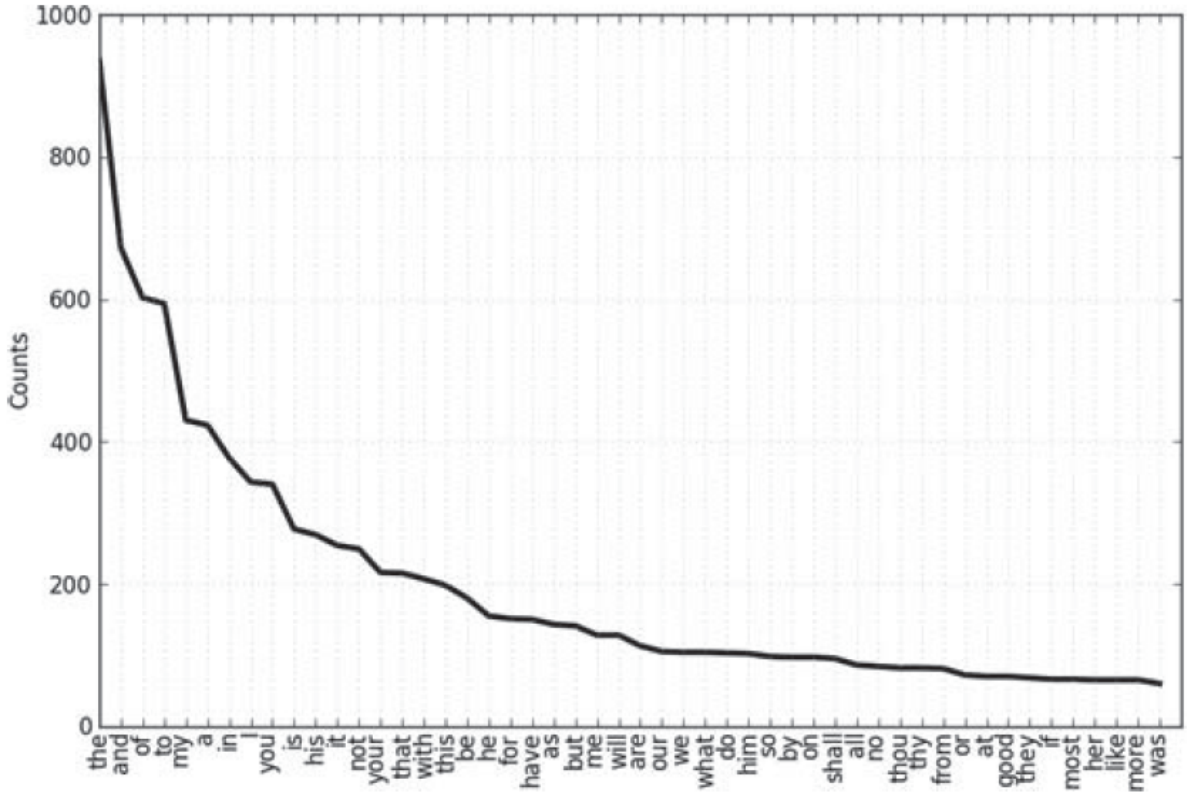
$$f(t,t') = \begin{cases} 1, & \text{if } t = t' \\ 0, & \text{otherwise} \end{cases}$$

$$TF_2(t,d) = \log[TF_1(t,d) + 1]$$

$$TF_3(t,d) = \frac{TF_1(t,d)}{n} \qquad |d| = n$$

Term Frequency

- Zipf's Law: the *i*-th most common word occurs approximately 1/*i* as the most frequent term.



Term Frequency

- An issue with Term Frequency
  - The importance of a term is solely based on its presence within a particular document.
  - What if this term frequently appears in every document? Is it still important?
- We need to have a broader view of the world
  - Consider the importance of a term not only in a single document but also in a corpus.

Document Frequency

- Document Frequency of a term:
  - The number of documents in a corpus that contain a term.
- Let a corpus $D = \{d_1, d_2, \ldots, d_N\}$

$$DF(t) = \sum_{i=1}^{N} f'(t, d_i) \qquad d_i \in D; |D| = N$$

$$f'(t, d') = \begin{cases} 1, & \text{if } t \in d' \\ 0, & \text{otherwise} \end{cases}$$

Inverse Document Frequency

- Inverse Document Frequency of a term

$$IDF_1(t) = \log \frac{N}{DF(t)} \qquad IDF_2(t) = \log \frac{N}{DF(t)+1}$$

- The IDF of a rare term would be high.
- The IDF of a frequent term would be low.
- IDF solely depends on the DF.

Term Frequency – Inverse Document Frequency (TFIDF)

- A measure that considers:
  - The prevalence of a term within a document (TF).
  - The scarcity of the term over the corpus (IDF).
- The TFIDF of a term *t* in a document *d* is

$$TFIDF(t,d) = TF(t,d) \times IDF(t)$$

- TFIDF scores a term higher if it appears more often in a document but less in a corpus.

Categorizing Documents by Topics

- TFIDF approach:
  - Represents a document *d* as a high-dimensional vector of TFIDF(*t*,*d*) values.
  - Provides relatively small amount of reduction in description length.
  - Reveals little inter-document or intra-document statistical structure.
- Topic models can overcome this problem.
  - A topic: a cluster of words with related meanings that frequently occur together.
    - Each word has a weight inside this topic.

Categorizing Documents by Topics

- Topic models are statistical models that:
  - examine words from a set of documents,
  - determine the themes over the text, and
  - discover how the themes are associated or change over time.

Categorizing Documents by Topics

- A document typically consists of multiple themes running through the text in different proportions

"This paper presents NeuroChess, a program which learns to play chess from the final outcome of games. NeuroChess learns chess board evaluation functions, represented by artificial neural networks. It integrates inductive neural network learning, temporal differencing, and a variant of explanation-based learning. Performance results illustrate some of the strengths and weaknesses of this approach."

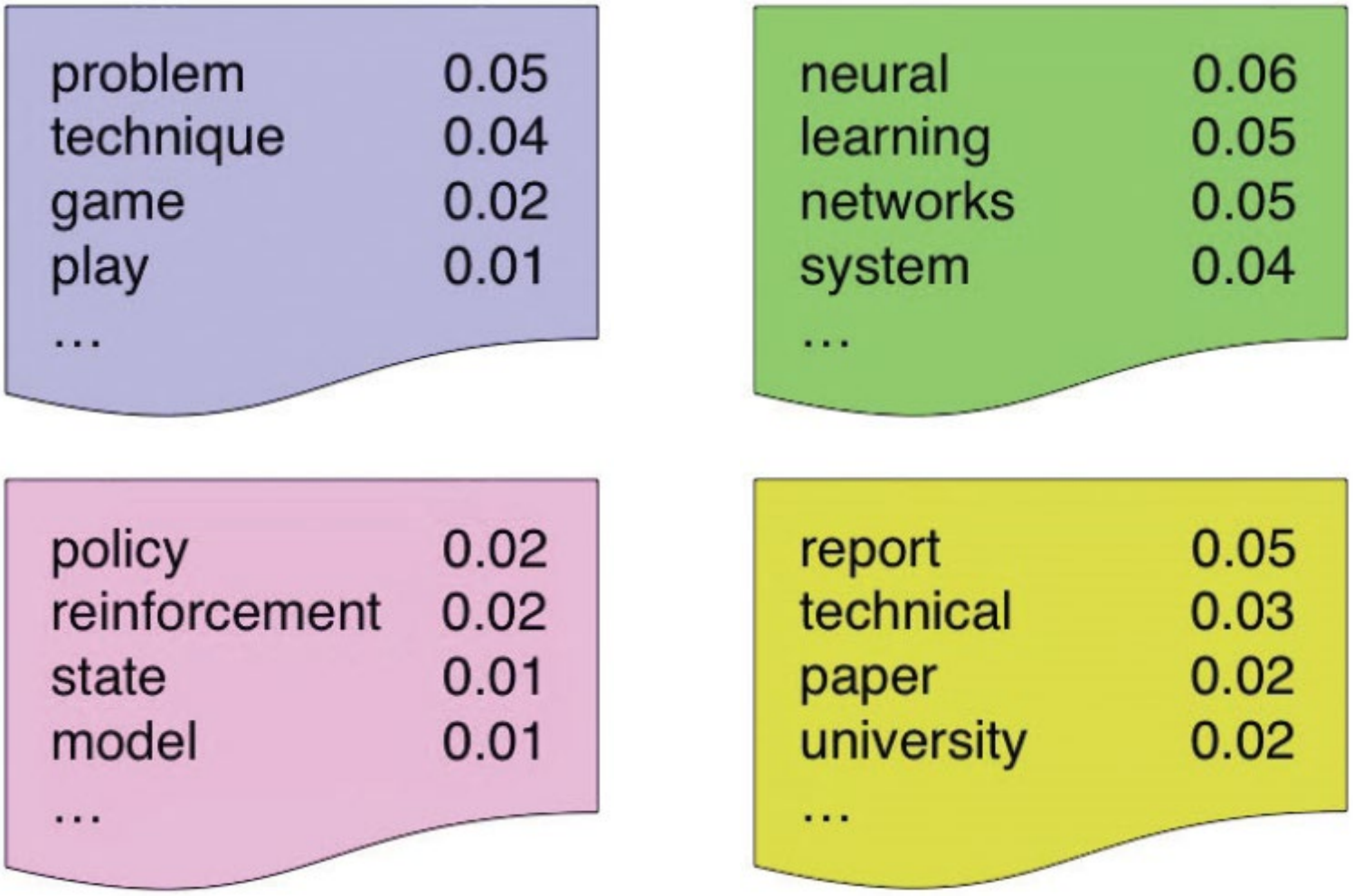
Categorizing Documents by Topics

- The process of topic modeling can be used to:
  1. Uncover the hidden topical patterns within a corpus.
  2. Provide short descriptions for documents.
  3. Annotate documents according to these topics.
     - Use annotations to organize, search, understand, and summarize texts.

Categorizing Documents by Topics

- A topic is formally defined as a distribution over a fixed vocabulary of words.
  - Different topics have different distributions over the same vocabulary.
- A topic can be viewed as a cluster of words with related meanings.
  - A word from the vocabulary can reside in multiple topics with different weights.

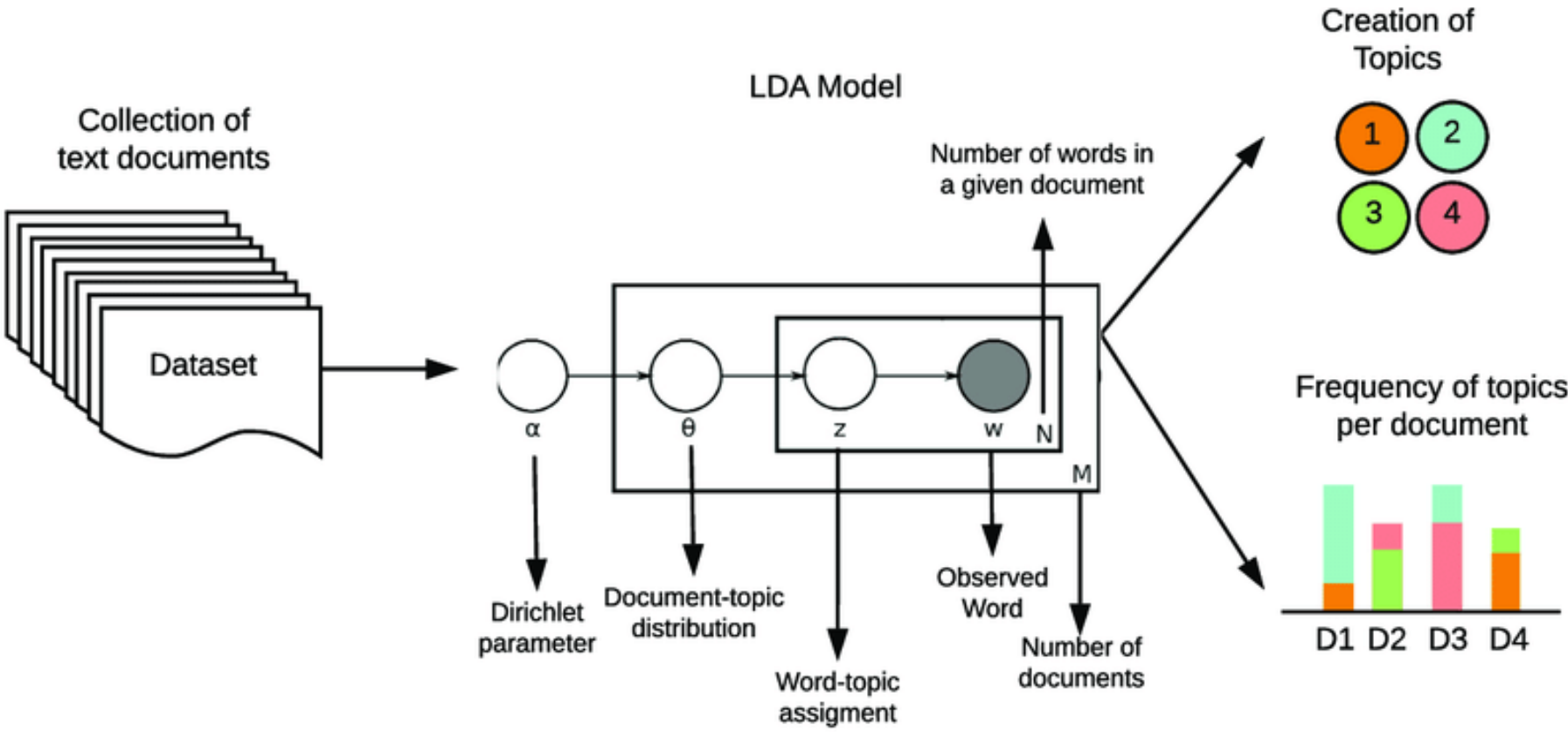
Categorizing Documents by Topics



Latent Dirichlet Allocation

- The simplest topic model is Latent Dirichlet Allocation (LDA)
  - A generative probabilistic model of a corpus.
- Generative probabilistic model
  - Model observations drawn from a probability density function.
  - LDA uses a hierarchical Bayes method.
- In LDA, documents are treated as the result of a generative process (with hidden variables)…

Latent Dirichlet Allocation (LDA)



Latent Dirichlet Allocation

- LDA assumes that each documents has been generated by the following process:
  - Select the number of words *N* for the document.
  - Choose a distribution over the topics.
  - For each of the *N* words of this document
    - Choose a topic based on the above distribution.
    - Choose a word from the corresponding topic.

- In reality, only the documents are available.
- LDA aims to infer the underlying topics, topic proportions, and topic assignment for each document.

Latent Dirichlet Allocation

- LDA assumes
  - There is a fixed vocabulary of words.
    - the vocabulary of words is fixed
  - The number of the latent topics is predefined.
    - the number of topics is fixed.
  - Each latent topic is characterised by a distribution over words in a vocabulary .
  - Each document is represented as a random mixture over latent topics.

Blei, David M. and Ng, Andrew Y. and Jordan, Michael I., "Latent Dirichlet Allocation", J. Mach. Learn. Res., 2003.

Latent Dirichlet Allocation

Latent Dirichlet allocation (LDA) is a generative probabilistic model of a corpus. The basic idea is that documents are represented as random mixtures over latent topics, where each topic is characterized by a distribution over words.[1]

LDA assumes the following generative process for each document $\mathbf{w}$ in a corpus D:

1. Choose $N \sim \text{Poisson}(\xi)$.
2. Choose $\theta \sim \text{Dir}(\alpha)$.
3. For each of the N words w_n:
   (a) Choose a topic $z_n \sim \text{Multinomial}(\theta)$.
   (b) Choose a word w_n from $p(w_n \mid z_n, \beta)$, a multinomial probability conditioned on the topic z_n.

author = {Blei, David M. and Ng, Andrew Y. and Jordan, Michael I.}, title = {Latent Dirichlet Allocation}, journal = {J. Mach. Learn. Res.}, year = {2003},

Latent Dirichlet Allocation

The principle algorithm:

1. Choose a value k (the number of topics)
2. For each document randomly assign each word in the document to one of the k topics.
3. For each document d, go through each word w and compute the proportion of words in d that are assigned to topic t (smoothing is normally applied):

$$P(t\,|d) = \frac{\text{count}(\text{words in d} \in \text{t}) + \epsilon}{N + k * \epsilon}$$

4. Compute the proportion of documents assigned to topic t for a given word w: $P(\,w \mid t\,)$
5. Compute the probability (weight) for the word w belonging to topic t:

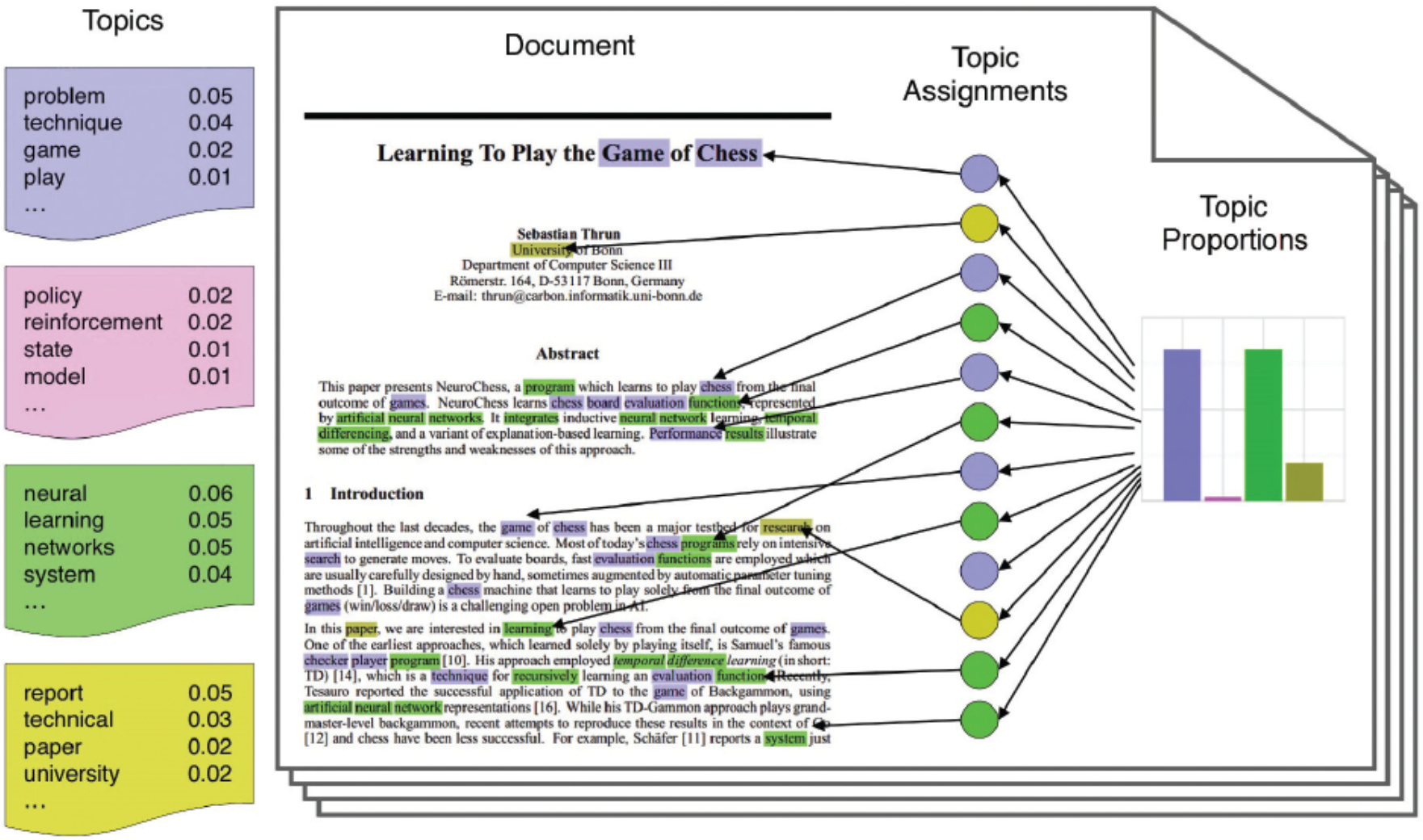
$$\text{P}(\text{w} \in t) = P(t\,|d)\ * P\,(\,w\,|t\,)$$

6. Reassign each word in each document based on $\text{P}(\text{w} \in t)$
7. Repeat steps 3 to 6 for several iterations

Latent Dirichlet Allocation

- LDA considers documents as a mixture of topics.
- LDA considers a topic is a mixture of words.
- If a word w has high probability of being in a topic, all the documents having w will be more strongly associated with t.
- If w is not very probable to be in t, the documents which contain the w will have a low probability of being in t, because the rest of the words in d will belong to some other topic and hence d will have a higher probability for those topic. So even if w gets added to t, it won't be bringing many such documents to t.

Latent Dirichlet Allocation



Latent Dirichlet Allocation

- For details: D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent Dirichlet Allocation,” Journal of Machine Learning Research, vol. 3, pp. 993–1022, 2003.
- R comes with an `lda package` that has built-in functions and example datasets
  - `cora` datasets (2,410 scientific documents)

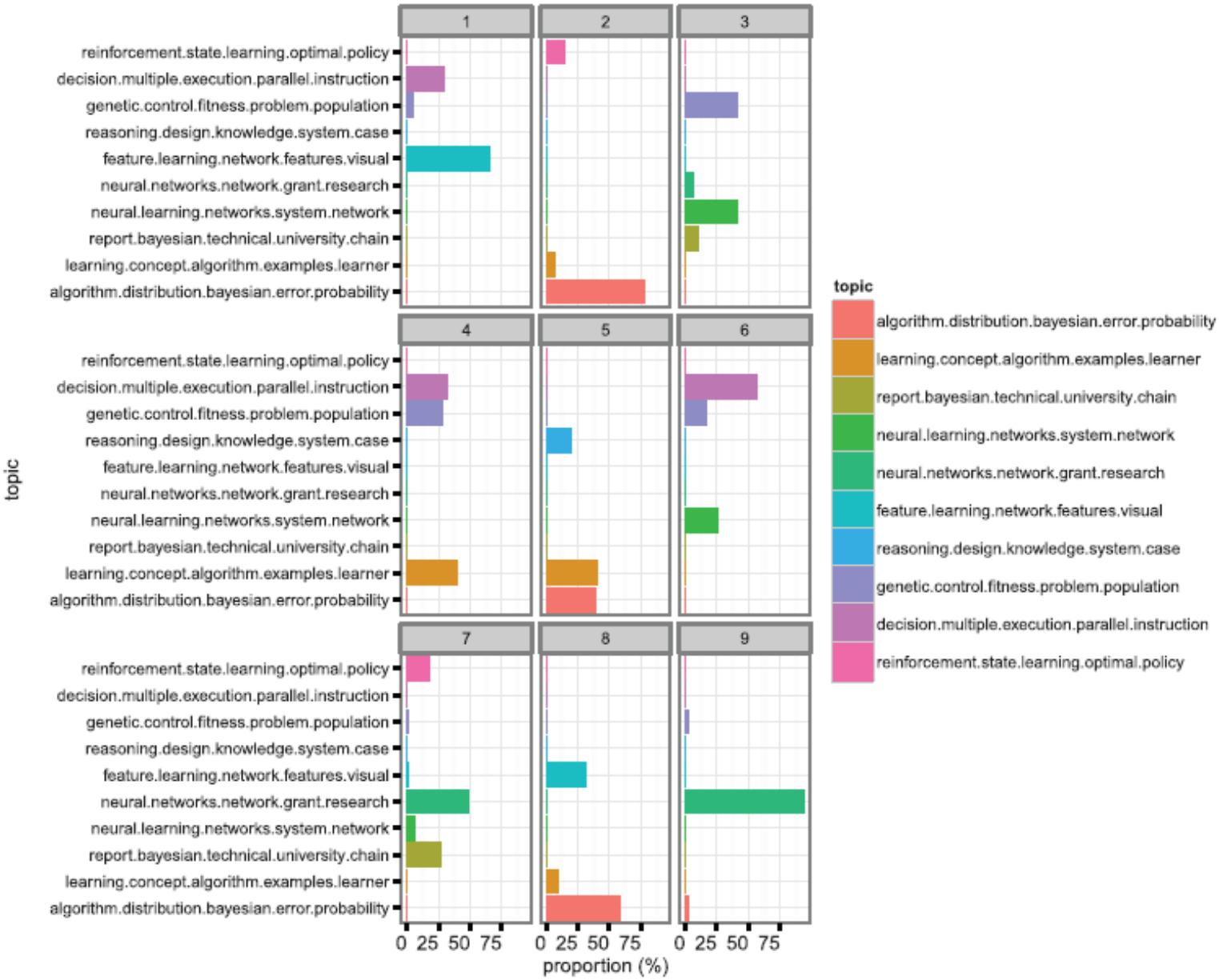
Latent Dirichlet Allocation in R

```
library("lda")
data(cora.documents)    #inbuild collection of scientific docs
data(cora.vocab)        #stemmed words
K=10                    #Number of topics
result <- lda.collapsed.gibbs.sampler(cora.documents,
                        K, cora.vocab,
                        25, ## Num iterations
                        0.1,0.1,compute.log.likelihood=TRUE)

# Get the top words in the cluster
top.words <- top.topic.words(result$topics, 5, by.score=TRUE)
```

Details can be found at https://cran.r-project.org/web/packages/lda/lda.pdf

Latent Dirichlet Allocation



Topic models vs Sentiment Analysis

- Topic models can be used in document modeling, document classification, and collaborative filtering.

- Sentiment analysis: mine opinions to identify and extract subjective information from texts.

Determining Sentiments

- Sentiment analysis
  - Uses statistics and NLP to mine opinions to identify and exact subjective information from texts.
- Applications
  - Detect the polarity of product or movie reviews.
- Analysis level
  - Document, sentence, phrase, and short-text.

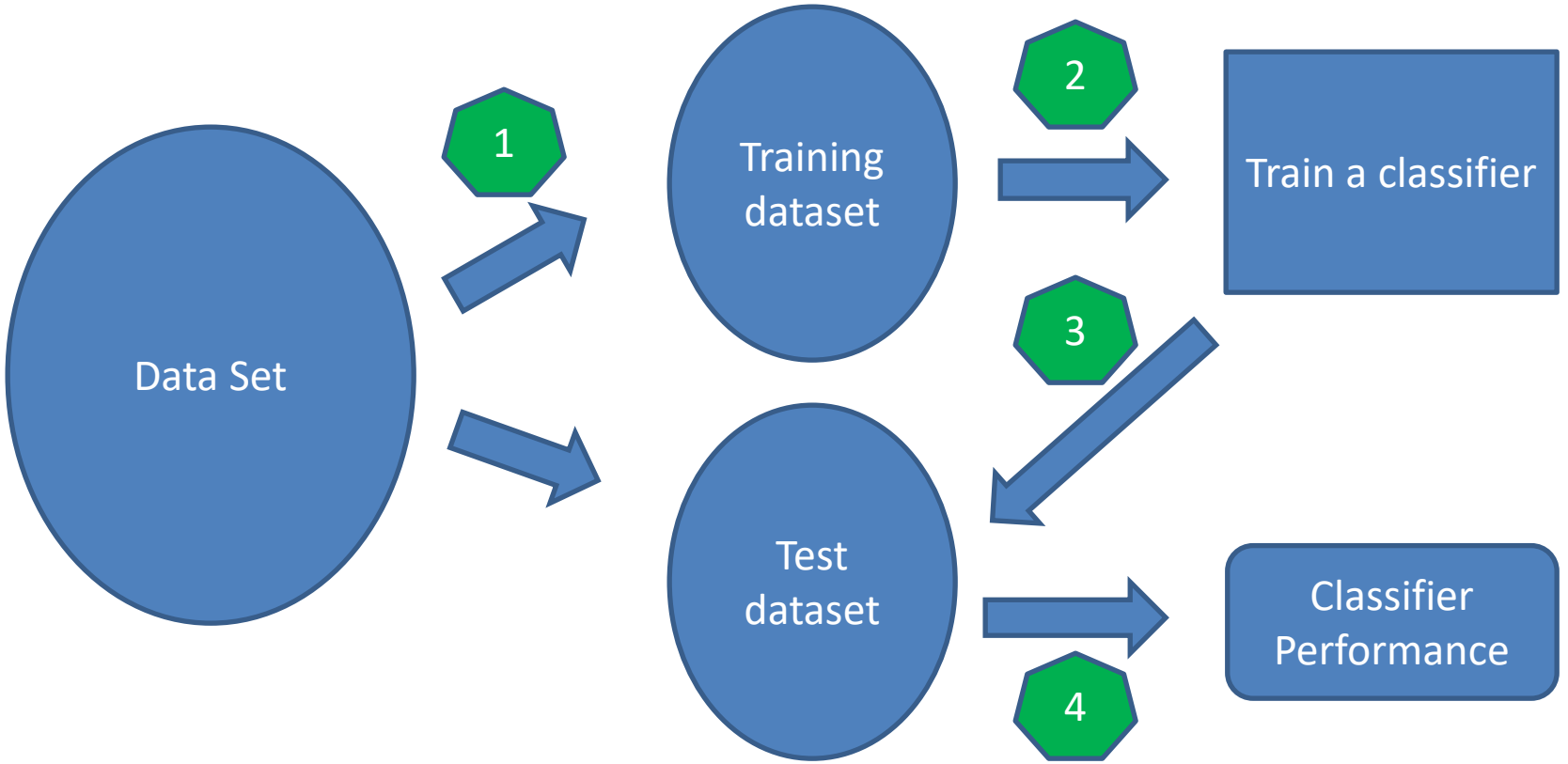
Determining Sentiments

- Classification methods are often used to extract corpus statistics for sentiment analysis
  - Naïve Bayes classifier, Maximum Entropy, Support Vector Machines, ….
- Movie review corpus, e.g. MovieLens
  - Consists of 2000 movie reviews.
  - Manually tagged into 1000 positive and 1000 negative reviews .

A. Go, R. Bhayani, and L. Huang, "Twitter Sentiment Classification Using Distant Supervision," *CS224N Project Report, Stanford,* pp. 1–12, 2009.

Determining Sentiments

- How to perform classification on a data set for sentiment analysis?



Determining Sentiments

- An example
  1. Using the Natural Language Processing Toolkit (NLTK) library in python.
  2. Split the 2000 reviews into 1600 reviews as training set and 400 reviews as testing set.
  3. Using BoW features.
  4. Naïve Bayes classifier learns from the training set.
     - The classifier achieves an accuracy of 73.5%.
     - Show most informative features for pos/neg.

Determining Sentiments

- Classifiers determine sentiments solely based on the datasets on which they are trained
  - Word meaning varies with the domain.
  - Cannot be directly apply to another domain.
- Absolute sentiment level is not informative
  - Compare with baseline result.
- How to label a larger number of reviews?
  - Use emoticons 
  - Use Amazon Mechanical Turk (MTurk).

Gaining Insights

- How data scientists use text analysis techniques to gain insights into their tasks?
- Word cloud (tag cloud):

Gaining Insights

- TFIDF can be used to highlight the informative words in text

★★★★★ **minor bugs** September 17, 2013
this was for my sister who loves it. she says it has minor bugs but nothing she cant deal with. she is overall satisfied with it

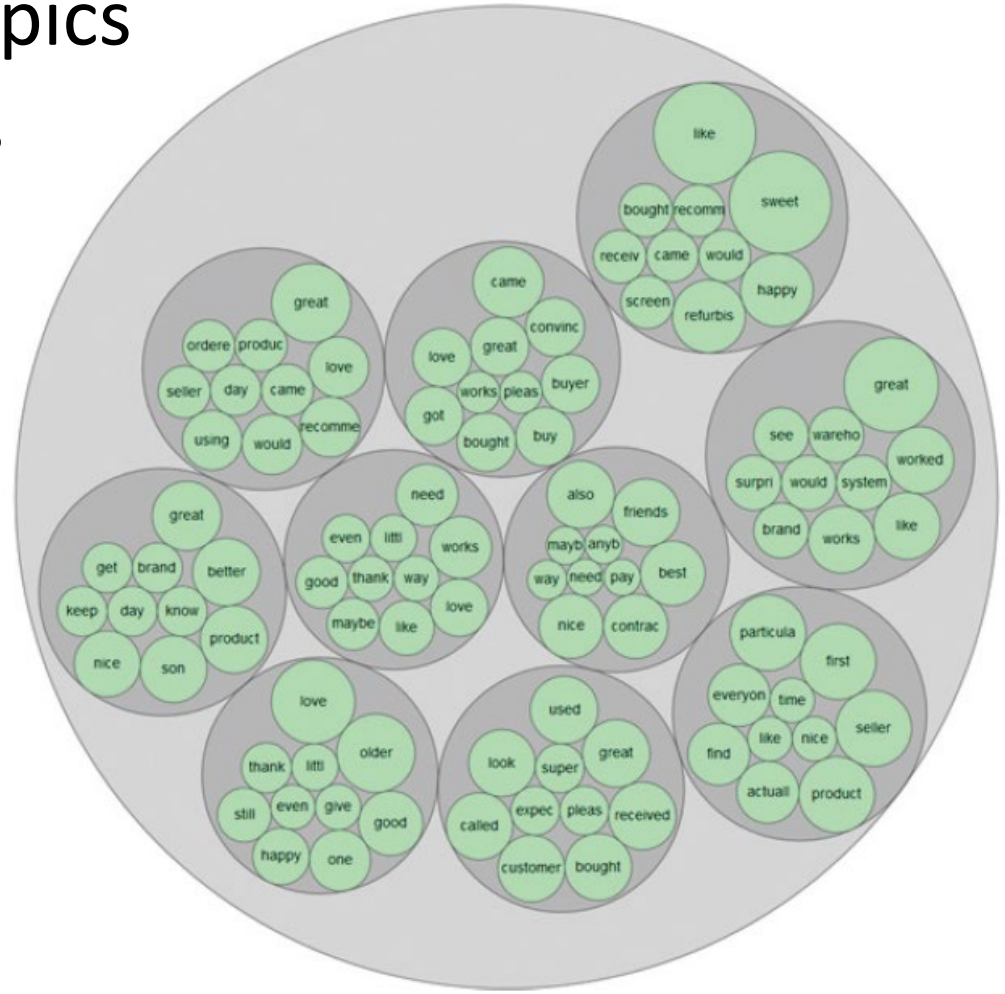
★★★★★ **mint condition ! ! 3** September 13, 2013
great price , not a scratch or bump on the bphone ! it came a lot speedier than expected so thats always a plus ! its just wonderful , only had it for a couple of days and could n't ask for anything more ! ! ! !

★ **buttons did not work** September 08, 2013
when i went to have my contacts transferred it was found that the two buttons need to switch did not work consistantly

★★★ **it 's a bphone.** August 12, 2013
i hate acme and acme products. base both on principle and on functionality (or lack thereof) . that being said i guess this phone is great for old people that are n't tech savvy. i bought this for my aunt .

Gaining Insights

- Circular graph of topics obtained from LDA.
  - The disc size represents the weight of a word.



Gaining Insights

- Another way to visualize topics:

Summary

- Discussed several subtasks of text analysis.
- Talks about a typical text analysis process
  - Collecting raw text.
  - Representing text.
  - Using TFIDF to describe each word in each doc.
  - Topic modelling.
  - Sentiment analysis.
  - Gaining greater insights.

Q&A

Images Courtesy of Google Image