Predictive Analytics

Chapter Five

Introduction to Text Mining

"What is a moderate interpretation of the text? Halfway between what it really means and what you'd like it to mean?"

Antonin Scalia

Anasse Bari, Ph.D.

Learning Outcomes

- Learning the Fundamentals of Text Mining and Text Categorization
- Acquiring Understanding of Data Cleaning in Textual Data
- Learning Document Vector Representation, Term Frequency Measures and Document Nearest Neighbors

Outline

- Introduction to Text Mining
- Text Categorization
- Data Cleaning in Textual Data
- Vector Representation
- Term Frequency Measures
- Similarity Measures in Text
- Document Nearest Neighbors

Introduction to Text Mining

- Text Categorization
  - Assign text documents to existing, well-defined categories.
- Clustering
  - Group text documents into clusters of similar documents.
- Text Filtering
  - Retrieve documents which match a user profile.
- Text Summarization: single vs. multiple documents

Text Categorization

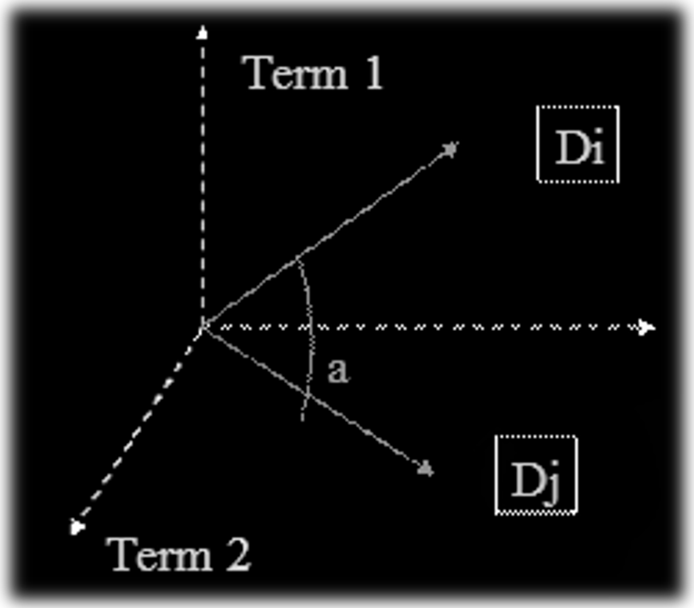
- Classify each test document by assigning category labels.
  - M-ary categorization assumes M labels per document.
  - Binary categorization requires yes/no decision for every document/category pair.
- Most techniques require training.
  - Parametric vs non-parametric.
  - Batch vs. on-line.

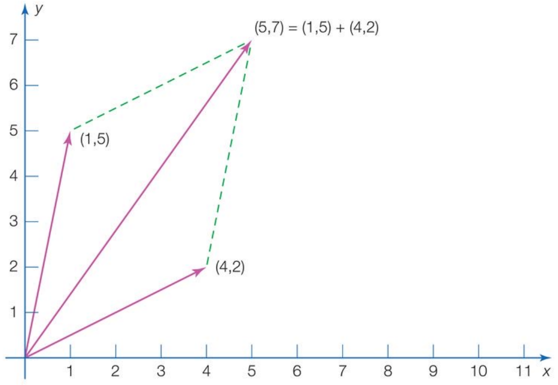
Data Cleaning in Textual Data

- Document parsing
- Stopwords: Set of words that are deemed “irrelevant”, even though they may appear frequently
  - E.g., a, the, of, for, with, etc.
  - Stop lists may vary when document set varies
- Stemming:
  - Several words are small syntactic variants of each other since they share a common word stem
  - E.g., drug, drugs, drugged
  - Porter’s algorithm
  - Dimension reduction
- Proximity Search support (n-gram, sliding window..) : To be able to search for a group of words as a single unit (like a noun phrase)

Vector Representation

- All documents are represented by word vectors
- Each document is represented by a vector
- Each dimension of the vector is associated with a word/term
- For each document, the value of each dimension is the frequency of that word that exists in the vector
- Given a collection of training data, present each term as a n-dimensional vector





a **vector** is a geometric object which has both magnitude or length and direction. A **vector** is commonly represented by a line segment in a specific direction, indicated by an arrow.

	D_1	D_2	…	D_j	…	D_n
T_1	w_{11}	w_{12}	…	w_{1j}	…	w_{1n}
T_2	w_{21}	w_{22}	…	w_{2j}	…	w_{2n}
…	…	…	…	…	…	…
T_i	w_{i1}	w_{i2}	…	w_{ij}	…	w_{in}
…	…	…	…	…	…	…
T_m	w_{m1}	w_{m2}	…	w_{mj}	…	w_{mn}

Weighted Schemes

- The weighted scheme of each term in a vector (sentence or document) is defined as follows:
  - $w(t_{ji}) = L(t_{ji}) \,.\, G(t_j)$ **---- Local Weight and Global Weight**
  - where, $L(t_{ji})$ is the local weight for term j in sentence i (or in the document)

    $G(t_j)$ is the global weight for term j in the whole document.
- The local weights are:
  - No weight (TF): $L(t_{ji}) = tf(t_{ji})$
  - Binary weight: $L(t_{ji}) = 1$, if $tf(t_{ji}) \geq 1$, $L(t_{ji}) = 0$, otherwise
  - Augmented weight: $L(t_{ji}) = 0.5 + 0.5 * (tf(t_{ji})/tf(max))$ where, $tf(max) = max\{ tf(t_{1i}), tf(t_{2i}), ... , tf(t_{mi})\}$ and m is the max number of terms in the document.
  - Logarithm weight: $L(t_{ji}) = \log (1 + tf(t_{ji}))$
- The global weights are:
  - No weighting: $G(t_j) = 1$
  - Inverse document frequency (IDF): $G(t_j) = \log(N/n(t_j))$ where, N is the total number of sentences in the document, and $n(t_j)$ is the number of sentences that contain term j.
- Normalization
  - Normalizes the sentence S_i (or document D) by its length $| S_i |$ (or $| D |$)

Term Frequency Measure

- Let's define some statistics for text documents:
  - TF: term frequency
    - In the case of the term frequency tf(t,d), the simplest choice is to use the raw frequency of a term in a document, i.e. the number of times that term t occurs in document d.
  - IDF: Inverse document frequency
    - The inverse document frequency is a measure of how much information the word provides, that is, whether the term is common or rare across all documents.

$$\mathrm{idf}(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$

    - N is the total number of documents in the corpus.
    - The denominator of above equation is the number of documents where the term t appears. If the term is not in the corpus, this will lead to a division-by-zero. It is therefore common to adjust the denominator to

$$1 + |\{d \in D : t \in d\}|$$

Term Frequency Measure

- Let's define some statistics for text documents:
  - TFIDF: Term frequency–Inverse document frequency
  - Then tf–idf is calculated as

$$\mathrm{tfidf}(t, d, D) = \mathrm{tf}(t, d) \times \mathrm{idf}(t, D)$$

  - A high weight in tf–idf is reached by a high term frequency (in the given document) and a low document frequency of the term in the whole collection of documents;
  - Since the ratio inside the idf's log function is always greater than or equal to 1, the value of idf (and tf-idf) is greater than or equal to 0.
  - As a term appears in more documents, the ratio inside the logarithm approaches 1, bringing the idf and tf-idf closer to 0.

Similarity Measures in Text

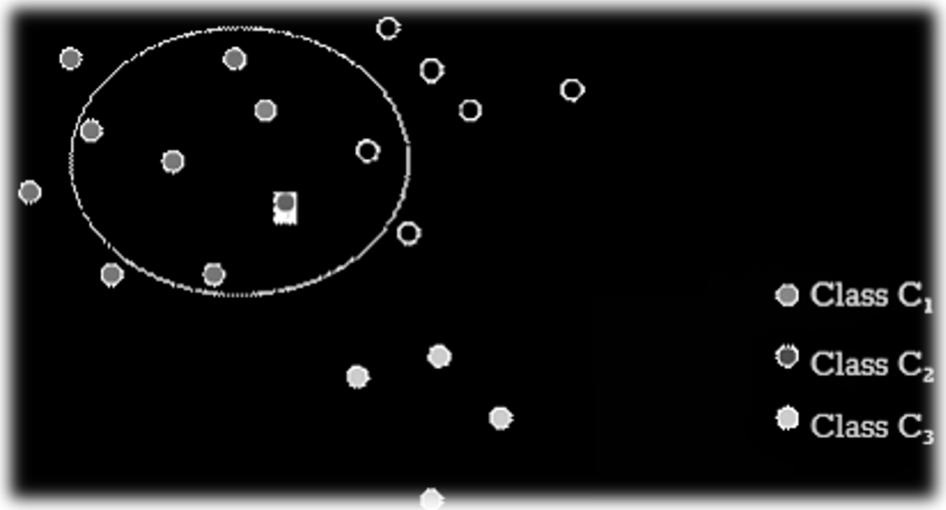
- For various tasks, need measurement of similarity between documents
  - Cosine similarity
  - Manhattan Distance
  - Mahalanobis Distance
- Cos similarity correspond to the angle between the two vectors

$$\left|\bar{v}\right| = \sqrt{\sum_{i=1}^{n} v_i^2}\,; \left|\bar{u}\right| = \sqrt{\sum_{i=1}^{n} u_i^2}\,;$$

$$\cos(\bar{u}, \bar{v}) = \frac{\bar{u} \bullet \bar{v}}{\left|\bar{u}\right| \times \left|\bar{v}\right|}$$

Document Nearest Neighbors

- Training set includes classes.
- Examine K documents near document to be classified.
- K is determined empirically.
- New document placed in class with the most number of close documents.
- O(n) for each document to be classified
- For each pattern in the test set, search for the k nearest patterns to the input pattern using a Euclidean distance measure
- Compute the confidence Ci /k for a class i, that is the number of patterns among the K nearest patterns belonging to class i . The output is the class with the highest confidence.



Document Nearest Neighbors - Example

- We have three kinds of document:
  - Politics
    - D1: President Obama went to Europe to negotiate on foreign policy
    - D2: North Korea changed it's foreign policy to make the world more peaceful for people.
    - D3: President Obama will talk about peaceful world in the future.
  - Health
    - D1: Recent research on human body tell us successes to treat the cancer.
    - D2: To have more healthy body you should have minimum 10 minutes workout
    - D3: Doing sport, workout prevent your body to get some cancers and even make you more happier.
  - Social
    - D1: There are lots of homeless people in the world.
    - D2: To have more happy life, do the thing you like.
    - D3: We hope a day all people in the world have a happier life.

Document Nearest Neighbors - Example

- We have three kinds of document:
  - Politics
    - D1: President Obama went to Europe to negotiate on foreign policy
    - D2: North Korea changed it's foreign policy to make the world more peaceful for people.
    - D3: President Obama will talk about peaceful world in the future.
  - Health
    - D1: Recent research on human body tell us successes to treat the cancer.
    - D2: To have more healthy body you should have minimum 10 minutes workout
    - D3: Doing sport, workout prevent your body to get some cancers and even make you more happier.
  - Social
    - D1: There are lots of homeless people in the world.
    - D2: To have more happy life, do the thing you like.
    - D3: We hope a day all people in the world have a happier life.

Dictionary
negotiate
foreign
policy
world
peaceful
Research
Human
Body
Treat
Cancer
Healthy
Workout
Sport
Homeless
People
Happy
Life

TF – Term Frequency

Class	Politics			Health			Social		
Dictionary	D1	D2	D3	D1	D2	D3	D1	D2	D3
negotiate	1	0	0	0	0	0	0	0	0
foreign	1	1	0	0	0	0	0	0	0
policy	1	1	0	0	0	0	0	0	0
world	0	1	1	0	0	0	1	0	1
peaceful	0	1	1	0	0	0	0	0	0
Research	0	0	0	1	0	0	0	0	0
Human	0	0	0	1	0	0	0	0	0
Body	0	0	0	1	1	1	0	0	0
Treat	0	0	0	1	0	0	0	0	0
Cancer	0	0	0	1	0	1	0	0	0
Healthy	0	0	0	0	1	0	0	0	0
Workout	0	0	0	0	1	1	0	0	0
Sport	0	0	0	0	0	1	0	0	0
Homeless	0	0	0	0	0	0	1	0	0
People	0	1	0	0	0	0	1	0	1
Happy	0	0	0	0	0	1	0	1	1
Life	0	0	0	0	0	0	0	1	1

Document Nearest Neighbors - Example

- Now, suppose we have a new document and we are looing for the most related class:
  - When you help some homeless people, you will feel more happy.
  - **We create the term frequency vector of input document ➔**
- We need to compute the cosine distance based on mentioned formulas

Dictionary	TF
negotiate	0
foreign	0
policy	0
world	0
peaceful	0
Research	0
Human	0
Body	0
Treat	0
Cancer	0
Healthy	0
Workout	0
Sport	0
Homeless	1
People	1
Happy	1
Life	0

Document Nearest Neighbors - Example

- We have three kinds of document (higher value = more similar).
- We used the mentioned cos formula, the results are:
  - Politics
    - D1: cos(u,v) is 0
    - D2: cos(u,v) is 0.29
    - D3: cos(u,v) is 0
  - Health
    - D1: cos(u,v) is 0
    - D2: cos(u,v) is 0
    - D3: cos(u,v) is 0.26
  - Social
    - D1: cos(u,v) is 0.66
    - D2: cos(u,v) is 0.33
    - D3: cos(u,v) is 0.66
- **Therefore, the new document is classified as social class (K=2 or 3).**

Predictive Analytics

Chapter Five

Introduction to Text Mining

"What is a moderate interpretation of the text? Halfway between what it really means and what you'd like it to mean?"

Antonin Scalia

Anasse Bari, Ph.D.