Text Classification using knn

What is Text mining?

- To discover the useful patterns/contents from the large amount of data that can be structured or unstructured.

Prachi M Joshi
Machine Learning Demystified

Text mining

- What can be used for text mining??
  - Classification/categorization
  - Clustering
  - Summarization
  - Retrieval…….

Pre-processing of text

- Tokenisation: Separation of tokens with removal of special symbols that are not required in the text.
- Stemming: Convering the words like ‘playing’, ‘played’ into ‘play’.
- Lemmatisation: Returning the base form of the word. Eg: heard – ‘hear’
- Case folding: Conversion of case- caps to small
- Stop word removal: ‘the’, ‘an’, ‘on’… are called stop words. They are of limited use when it comes to determine the weight of a document for retreival.
- Normalisation: Equivalence classing. Use of synonyms. Spell check too can be performed here.

Prachi M Joshi
Machine Learning Demystified

Different models for representation

- Term frequency and weighing
  - Bag of words: number of occurrance of word where the exact ordering is ignored.
- Vector space model and so on….

Term frequency

- Term frequency is the number of times, the term occurs in the document.
- Eg: 'Cricket is a game. Sam likes the game of cricket.'

Terms	Cricket	is	a	game	Sam	likes	the	of
Freq	2	1	1	2	1	1	1	1
normalised	2/10	1/10	1/10	2/10	1/10	1/10	1/10	1/10

- Each documents varies in size.
- Thus the frequency of terms differs with the size. And it impacts the smaller ones
- Thus it is normalised

Prachi M Joshi
Machine Learning Demystified

Inverse document frequency

- The whole intension for the terms generation is finding out relevant documents to one specific or to a query that is fired.
- Occurrence of a term more times cannot indicate the power or potential to determine relevance.
- Thus their weight needs to be scaled down.
- We use idf :
- $idf_t = \log\left(\frac{N}{df_t}\right)$
  - Where t is the terms, N = total no. of documents and
  - df_t = no. of documents with t term.

- So,
- For 3 documents:
  - D1= Cricket is a game. Sam likes the game of cricket.
  - D2= Do you play cricket?
  - D3 = Playing any game is good for health. I play basketball.
  - For D1, the idf values:

	Cricket	is	a	game	Sam	likes	the	of
Tf	2	1	1	2	1	1	1	1
Normalised tf	2/10	1/10	1/10	2/10	1/10	1/10	1/10	1/10
idf	Log(3/2)	Log(3/2)	Log(3/1)	Log(3/2)	Log(3/1)	Log(3/1)	Log(3/1)	Log(3/1)

Prachi M Joshi
Machine Learning Demystified

Tf-idf

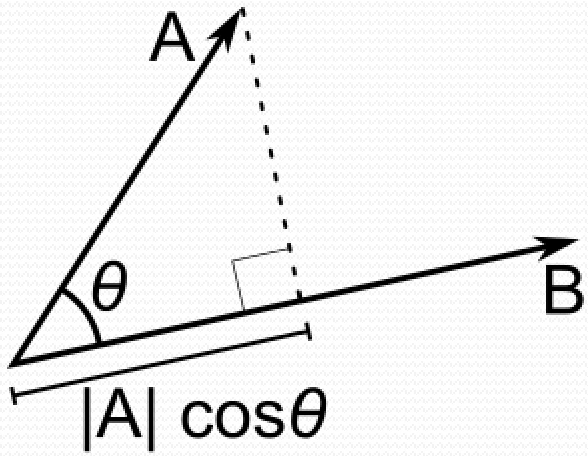
- To find relevant documents, generally a combined weighted approach is used called as tf-idf.
- So:
  - $w_{t,d} = tf_{t,d} * idf_t$

- Representation of set of documents as vectors in common vector space is known as vector space model.

Calculating similarities between the documents

- Often cosine similarity is used.
- It is a measure of orientation and not magnitude.
- We are interested in determining the orthogonality.

- More about Dot product:
  - When we consider the dot product of two vectors say $\vec{a}.\vec{b}$, we are trying to project a into b. The angle between these vectors determines the orthagonality. If it is 90 degrees, the vectors are orthogonal.



Cosine similarity

Cosine Similarity (d1, d2) = Dot product(d1, d2) / ||d1|| * ||d2||

Dot product (d1,d2) = d1[0] * d2[0] + d1[1] * d2[1] + … + d1[n] * d2[n]

||d1|| = square root($d1[0]^2 + d1[1]^2 + ... + d1[n]^2$)

||d2|| = square root($d2[0]^2 + d2[1]^2 + ... + d2[n]^2$)

Prachi M Joshi
Machine Learning Demystified

Classifying the text documents

- For the given training data:
  - Calculate tf of each document
  - Normalize it
  - For a new unknown test data, calculate tf, normalise
  - Use kNN to classify this document using cosine similarity.
  - Use training data:
    - D1: “This is big classroom” - classroom
    - D2: “Classroom has many benches” - classroom
    - D3: “This is house” - house
    - D4: “The house has garden” - house
    - D5: “The house is big” - house
    - Classify: ‘Big house’ - house