International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2017 IJSRCSEIT | Volume 2 | Issue 6 | ISSN : 2456-3307

Document Analysis using Similarity Measures : A Case Study on Text Retrieval System

Suresha M
Department of Computer Science, Kuvempu University, India

ABSTRACT

A document is an information container that contains information either in printed format or in handwritten format and document is a medium for transferring knowledge. Human vision is the most accurate language identification system in the world. Within a few seconds of looking at a document, one can determine the language even without deskewing and segmenting the image, while computer vision is not able to match human capability. Today there is an increasing need for automatic language identification with the support of computers. As the world moves from paper to paperless office, more and more communication and storage of documents is performed digitally which facilitates quicker additions, searches and modifications and increases the life of such records.

Keywords: Document Analysis, Similarity Measures, Text Retrieval.

I. INTRODUCTION

A. Document categories and applications

A document is a written or printed paper that bears the original, official, or legal form of data. There is a wide variety of documents that we encounter in day to day life. This includes the documents that are used to communicate information in the form of letters and newspapers. The broad categories of documents are described below.

B. Hard and soft documents

A hard document is a physical form of document where information is present in textual or Graphical form. Soft documents are the ones which are created with the help of electronic devices. There are a number of electronic devices which allow the conversion of hard documents to soft ones. Examples of digitizing devices are scanners, cameras etc.

Soft documents constitute the documents generated using markup languages, word processors, synthetic document images, scanned images etc. Soft documents can be structured or unstructured. Structured documents present the information in a well organized manner to aid the information extraction.

C. Printed and Hand written documents

Printed documents belong to the class of documents which are generated either mechanically or electronically from the existing data. Most of the documents we encounter now-a-days are of this kind. Handwritten documents are also very popular. Processing and recognition of handwritten documents is relatively difficult. The cursive scripts, variability and separation of characters, document skew, accommodating variability of strokes, learning the human characteristics are very difficult to model. There are also hybrid documents, where printed and handwritten words Are present together. These documents may also contain texts, tables, graphs and graphics.

D. Single language and multilingual documents

A document which contains only one language (either Kannada, English, or Hindi etc.) is known single language document. India is a country with diverse languages and scripts. Therefore, the real-life documents can be also multilingual. Many documents use English, the national language Hindi and a regional language for official and commercial purposes. Processing of such documents is usually approached by identifying the script and employing the appropriate recognizing scheme.

E. Online and Offline Documents

A new class of online documents find tremendous applications with handheld devices and natural interfaces. In this category, the digitizer also provides the time information along with the spatial and intensity content of the image. The time of writing of each point on the curve is stored along with the intensity, color and pressure information. Conventional document analysis has been focusing on the offline documents. They consider a character as a single image where online documents represent characters as a sequence of strokes. Algorithms for processing of online documents utilize time information and map the problem into an ordered sequence analysis.

F. Typical Application Domains

1) *Newspaper Documents*

Newspapers are one of the first mass communication media introduced by human being. Gradually other media like radio, television and now the Internet became popular. Even then newspapers remain as the most appealing information source for many. Newspapers are hard printed offline documents, with very rich information content. These documents exist ever since printing was invented. The current number of available newspapers is so abundant that it is very difficult to maintain them in a perishable paper format.

2) *Form Processing*

Any document requesting or collecting information from a user in specific format is a form. Forms are one of the most common class of documents which organizations encounter. For example, forms are used in railway reservations, handling deposits and withdrawals in banks, educational institutions, applications, data collection etc. In a form, each field has its corresponding value. The problem of form understanding is extracting the information related to each field. Since forms are meant for very specific applications one usually expects very high accuracy and processing speeds. Forms also contain many standard characteristics like lines, boxes etc.

3) *Envelopes and Letters*

Even in this modern era, regular mails form a major communication mode. In the Indian context, addresses are often multilingual and written in cursive scripts without PIN codes. This makes the problem extremely difficult to handle. Most of the time, even the localization of address and other details is itself difficult. People also tend to use their own abbreviations and spelling for many cities and villages. At the same time, the finite number of post offices and city names can provide dictionary information to improve the recognition accuracy.

4) *Archival of Existing Documents*

The most common application of document image analysis is to convert an existing hard document into a soft text form such that manipulation of the document is possible. Many of the commercial scanners also provide software to do this. However, they do not support Indian multilingual documents. Such intelligent digitization of paper documents finds tremendous applications in various domains. An important class of documents falling in this category is Judgments given by courts. The judicial system in India has started 50 years ago and court documents are filed and stored from then. A proper digitization of these documents can help the judiciary to provide speedy and better judgments. Earlier judgments were handwritten, while the later ones were typewritten and some of the recent ones being completely electronic. One has to keep this diversity in mind to design a versatile system. The recognition performance in official documents like judgments can be improved largely by using the domain knowledge because of the limited and special vocabulary used in official documents.

5) *Speech Applications*

With advancements in speech technologies applications involving speech generation and OCR are developed. Applications like document reading devices involve OCR followed by a Text to Speech (TTS) converter. The text obtained from the OCR output of the document image is converted into sound signal by TTS converter. These applications are of immense help to blind and illiterate people. Advanced applications in this field may also introduce a language translation module between the OCR and TTS. This helps any person to understand a document in any unknown language.

The exponential growth of the Internet has led to a huge increase in the rich textual information and Text data is ubiquitous everywhere As the volume of text data increases, management and analysis of text data becomes very important task

II. Text Preprocessing Techniques

The main objective of text preprocessing is to transform unstructured or semi structured text data into structured data model

- Collection reader: Transform raw document collection into a common format, e.g., XML
- Detagger: Find the special tags in document e.g., "", "".
- Tokenizer: Nonempty sequence of characters, excluding spaces and punctuations e.g., " "
- Stop word removal: Function words and connectives Appear in a large number of documents and have little use in describing the characteristics of documents e.g., "of", "a", "by", "and", "the", "instead"
- Stemming: Remove inflections that convey parts of speech, tense and number e.g., *University* and *Universal* both stem to *Universe*
- Prunning: Discard words appearing rarely or more frequently
- Term weighting: Weight the frequency of a term in a document
- Not all terms are equally useful, Terms that appear too rarely or too frequently are ranked lower than terms that balance between the two extremes

III. Text Data and Representation Models

Text data occur in different formats

- Plain text
- DOC
- PDF
- PS
- HTML
- XML
- Email

Representation Model

In information retrieval and text mining, text data of different formats is represented in a common representation model for example Vector Space Model (VSM) is the most popular representation model used in information retrieval and text mining. In VSM, a text document is represented as a vector of terms <t_1, t_2, …, t_i, …, t_n>. Each term t_i represents a word or a phrase. The set of all *n* unique terms in a set of text documents forms the vocabulary for the set of documents. A set of documents are represented as a set of vectors, that can be written as a matrix. Text data is converted to the model representation is shown in figure 1 and graphical representation is shown in figure 2.

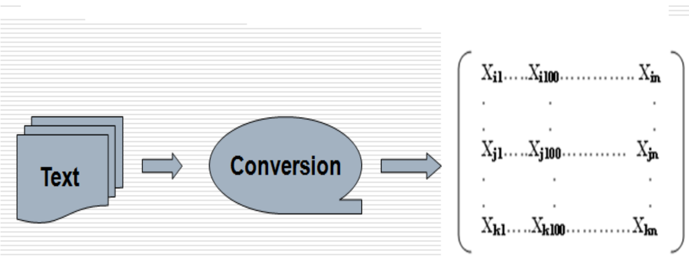

Text
Conversion
X_{i1} … X_{i100} … X_{in}
X_{j1} … X_{j100} … X_{jn}
X_{k1} … X_{k100} … X_{kn}

Figure 1. Text data is converted to the model representation

Example*:*

$$D_1 = 2T_1 + 3T_2 + 5T_3$$
$$D_2 = 3T_1 + 7T_2 + \quad T_3 \qquad (1)$$
$$Q = 0T_1 + 0T_2 + 2T3$$

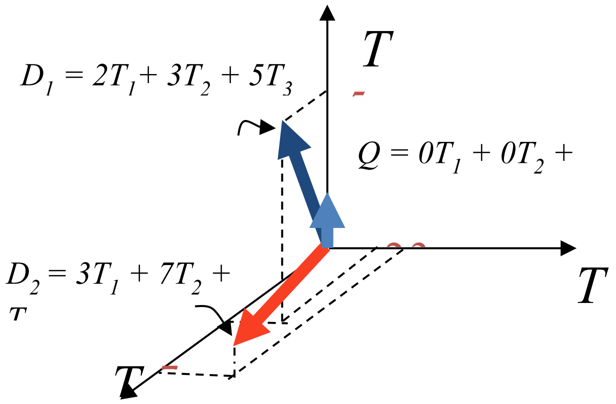
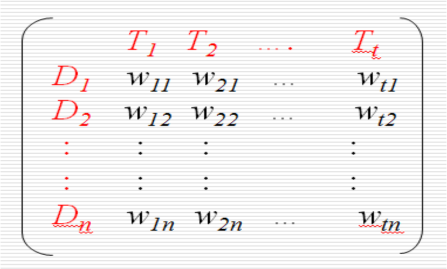


Figure 2 : Text data is converted to Graphical Representation

- Is D_1 or D_2 more similar to Q?
- How to measure the degree of similarity? Distance? Angle? Projection?

IV. Document Collection

A collection of *n* documents can be represented in the vector space model by a term-document matrix. An entry in the matrix corresponds to the weight of a term in the document. The value zero means the term has no significance in the document or it simply doesn't exist in the document as shown in the figure 3.



$$\begin{array}{c|cccc} & T_1 & T_2 & \dots & T_t \\ \hline D_1 & w_{11} & w_{21} & \dots & w_{t1} \\ D_2 & w_{12} & w_{22} & \dots & w_{t2} \\ : & : & : & & : \\ : & : & : & & : \\ D_n & w_{1n} & w_{2n} & \dots & w_{tn} \end{array}$$

Figure 3: Term Weights

More frequent terms in a document are more important

f_{ij} = frequency of term i in document j

May want to normalize *term frequency* (*tf*) across the entire corpus:

$tf_{ij} = f_{ij} \;/ \max\{f_{ij}\}$

df_i = document frequency of term

i = number of documents containing term i

idf_i = inverse document frequency of term

$i = \log_2 (N/ df_i)$

where N is total number of documents

V. Similarity Measure

A similarity measure is a function that computes the *degree of similarity* between two vectors. There are mainly two types of similarity measures, namely

A. Inner product similarity measure and
B. Cosine Similarity Measure

A. Inner product similarity measure

Gives the similarity between vectors for the document d_i and query q is computed as the vector inner product.

$$\text{sim}\ (d_j,q) = d_j \bullet q = w_{ij} \cdot w_{iq} \qquad (2)$$

Where w_{ij} is the weight of term i in document j and w_{iq} is the weight of term i in the query for binary vectors, the inner product is the number of matched query terms in the document. For weighted term vectors it is the sum of the products of the weights of the matched terms.

Binary vectors

D = 1, 1, 1, 0, 1, 1, 0
Q = 1, 0 , 1, 0, 0, 1, 1
Sim (D, Q) = 3
Size of vector = size of vocabulary = 7
0 means corresponding term not found in document or query

Weighted:
$D_1 = 2T_1 + 3T_2 + 5T_3$
$D_2 = 3T_1 + 7T_2 + 1T_3$
$Q = 0T_1 + 0T_2 + 2T_3$
sim(D_1 , Q) = 2*0 + 3*0 + 5*2 = 10
sim(D_2 , Q) = 3*0 + 7*0 + 1*2 = 2

B. Cosine Similarity Measure

Cosine similarity measures the cosine of the angle between two vectors.

Cos sim(dj, q) = d j / |dj|. |q|
$D_1 = 2T_1 + 3T_2 + 5T_3$
CosSim(D_1 , Q) = 10 / √(4+9+25)(0+0+4) = 0.81
$D_2 = 3T_1 + 7T_2 + 1T_3$
CosSim(D_2 , Q) = 2 / √(9+49+1)(0+0+4) = 0.13
$Q = 0T_1 + 0T_2 + 2T_3$

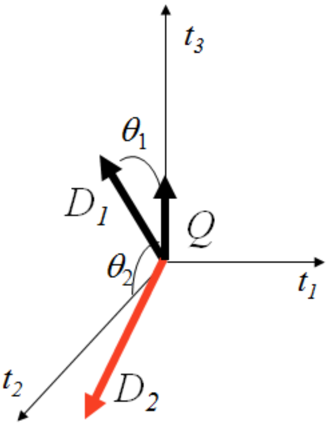


Figure 4: Graphical representation of text data

D_1 is 6 times better than D_2 using cosine similarity but only 5 times better using inner product.

VI. Text retrieval system

Figure 5 is the block diagram of Text retrieval system

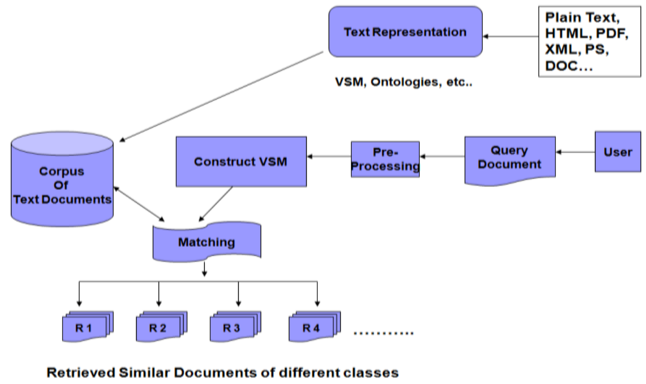


Figure 5: Retrieved similarity Documents of different Classes

Case Study

Doc 1
This is a document which contains information on text clustering

Doc 2
Text clustering is the information which is presented in this document

Doc 3
Cricket is an interesting game which is played by eleven players

Doc 4
The maximum numbers of players in the game cricket are eleven

Query
Eleven players will be playing cricket game

Elimination of Stop Words

Doc 1
Document contains information text clustering

Doc 2
Text clustering information presented document

Doc 3
Cricket interesting game played eleven players

Doc 4
maximum numbers players game cricket eleven

Query

Eleven players playing cricket game

BOW for Document 1

Document
contains
information
text
Clustering
Text
Text
Document
information

BOW for Document 2

Text
clustering
information
presented
document
Text
Clustering
document

BOW for Document 3

Cricket
interesting
game
played
eleven
players
Cricket
Game
cricket

BOW for Document 4

maximum
numbers
players
game
cricket
eleven
Eleven
Cricket
numbers

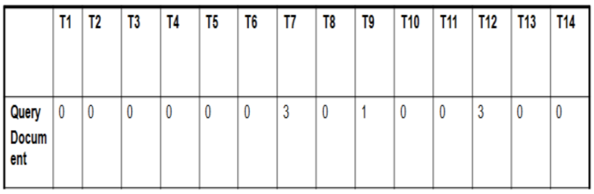
Assumption: Add more term to documents to get more term frequency

Query

Eleven players will be playing cricket game

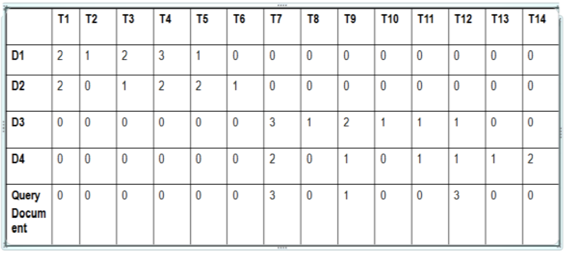
BOW for Document Query
Eleven
players
playing
cricket
Game
players
players
cricket
cricket

Table 1: Document Vs Term Matrix for Query Document



	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14
Query Document	0	0	0	0	0	0	3	0	1	0	0	3	0	0

Table 2: Matching of terms



	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14
D1	2	1	2	3	1	0	0	0	0	0	0	0	0	0
D2	2	0	1	2	2	1	0	0	0	0	0	0	0	0
D3	0	0	0	0	0	0	3	1	2	1	1	1	0	0
D4	0	0	0	0	0	0	2	0	1	0	1	1	1	2
Query Document	0	0	0	0	0	0	3	0	1	0	0	3	0	0

CosSim($D1$, Q) = 0 / 196 = 0
CosSim($D3$, Q) = 14 / 196 = 0.071

0.7790
CosSim(*D2* , *Q*) = 0 / 196 = 0
CosSim(*D4* , *Q*) = 10 / 196 = 0.051
0.6623

Ranking of Documents

Sort the values in descending order and rank the documents
CosSim(*D1* , *Q*) = 0 / 196 = 0
CosSim(*D2* , *Q*) = 0 / 196 = 0
CosSim(*D3* , *Q*) = 14 / 196 = 0.071
CosSim(*D4* , *Q*) = 10 / 196 = 0.051
Sorted Sequence: D3, D4, D2, D1
Documents Related to query document: D3, D4, D2, D1

VII. CONCLUSION

Document image analysis is very essential now a days. Document image analysis systems are designed to extract information from images, read text on a page, find fields in the form, identification of lines and symbols, recognition of logos etc. In this work, different documents that are available for analysis, representation of document data and case study for text retrieval system have been discussed.

VIII. REFERENCES

[1]. B B Chaudhuri and U Pal, Skew Angle Detection of Digitized Indian Script Documents, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.19, No.2, 1997.

[2]. Cattoni R., Coianiz T., Messelodi S., and Modena M C., 1998. Geometric Layout Analysis Techniques for Document Image Understanding: A Review, ITC - IRST, 1998.

[3]. Rangachar Kasturi, Lawrence o Gorman and Venu Govindaraju, Document image analysis: a primer, Sadhana, Vol 22, Part I, pp 3-22, 2002.

[4]. Song Mao, Azriel Rosen Feld, and Tapas Kanungo, Document structure analysis algorithms: A literature survey, Electronic Imaging, 2003.

[5]. Yu B., and Jain A K., A generic system for form dropout. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 18, No.11, 1996.

[6]. Yuan Y Tang, Seong Whan Lee, and Ching Y Suen, Automatic Document Processing: A Survey, Vol 29, No.12, pp 1931-1952, 1996.