Semantic Annotation of Documents

Participants:

Rashi Shrishrimal	201301203
Naman Singhal	201330072
K S Chandra Reddy	201505544

Project Guide:

Dr. Vasudev Verma

Priya Radhakrishnan

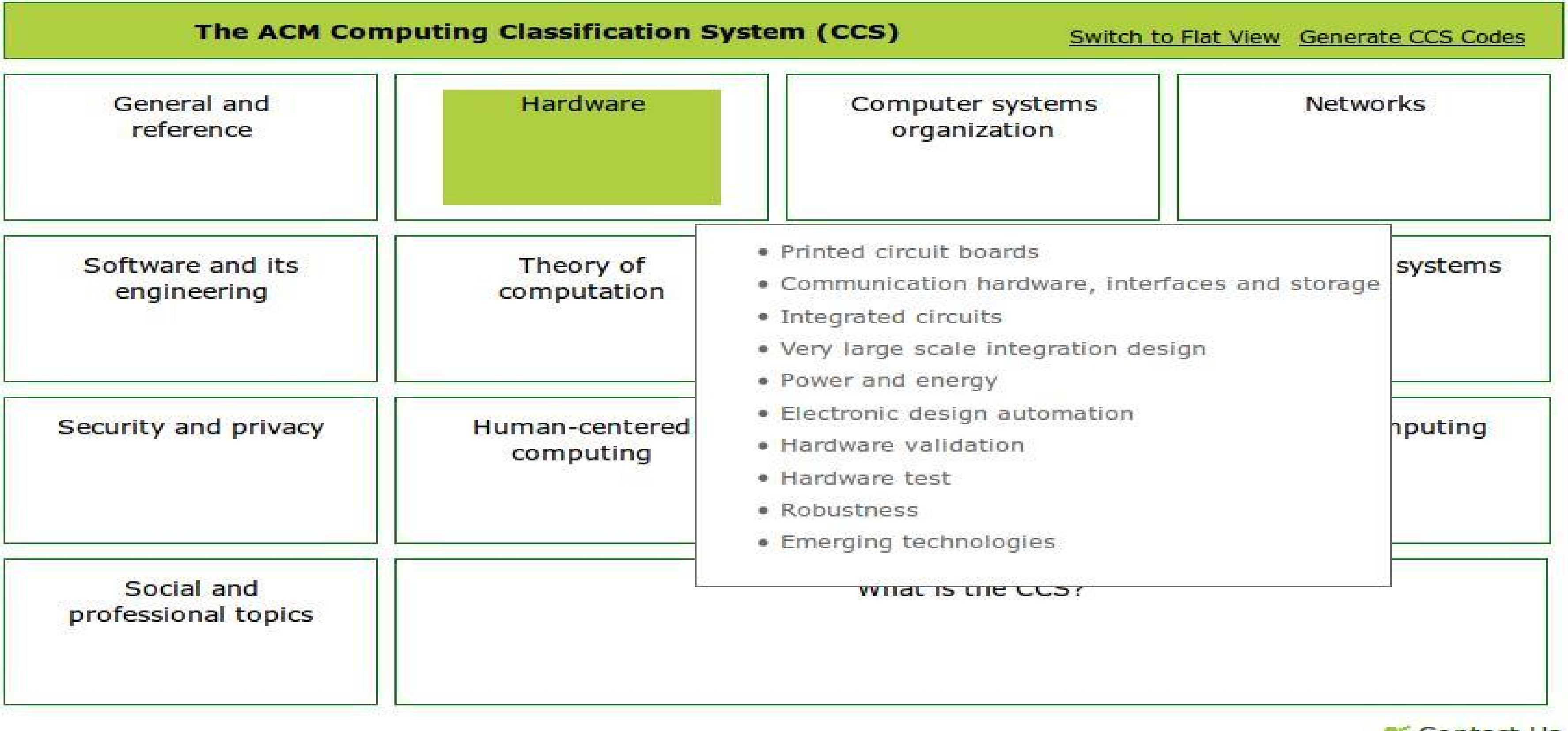
Problem Statement

Annotation of title and abstract of research paper to 2012 ACM classification categories.

The 2012 ACM Computing Classification System has been developed as a poly-hierarchical ontology that can be utilized in semantic web applications. It plays a key role in the development of a people search interface in the ACM Digital Library to supplement its current traditional bibliographic search.

The complete classification tree can be found here.

Snapshot of ACM Classification Tree



The ACM Computing Classification System (CCS)

Switch to Flat View Generate CCS Codes

General and reference

Hardware

Computer systems organization

Networks

Software and its engineering

Theory of computation

systems

Security and privacy

Human-centered computing

nputing

Social and professional topics

What is the CCS?

- Printed circuit boards
- Communication hardware, interfaces and storage
- Integrated circuits
- Very large scale integration design
- Power and energy
- Electronic design automation
- Hardware validation
- Hardware test
- Robustness
- Emerging technologies

Contact Us

Dataset Overview

1. A private dataset of the SIEL lab of IIIT-Hyderabad. Fields:
   a. Title
   b. Author
   c. Country of authors
   d. Year of publication
   e. Conference
   f. Categories
   g. Abstract (For few research papers)
2. Wiki Dataset : For all the acm categories a dataset was build having the summary of the wikipedia page (The First section of the wikipedia document).
3. Dblp dataset : For initial tasks, this dataset was used to train the lda model. Though this was not used for the final proposition of the model.

Technical Approach and Models

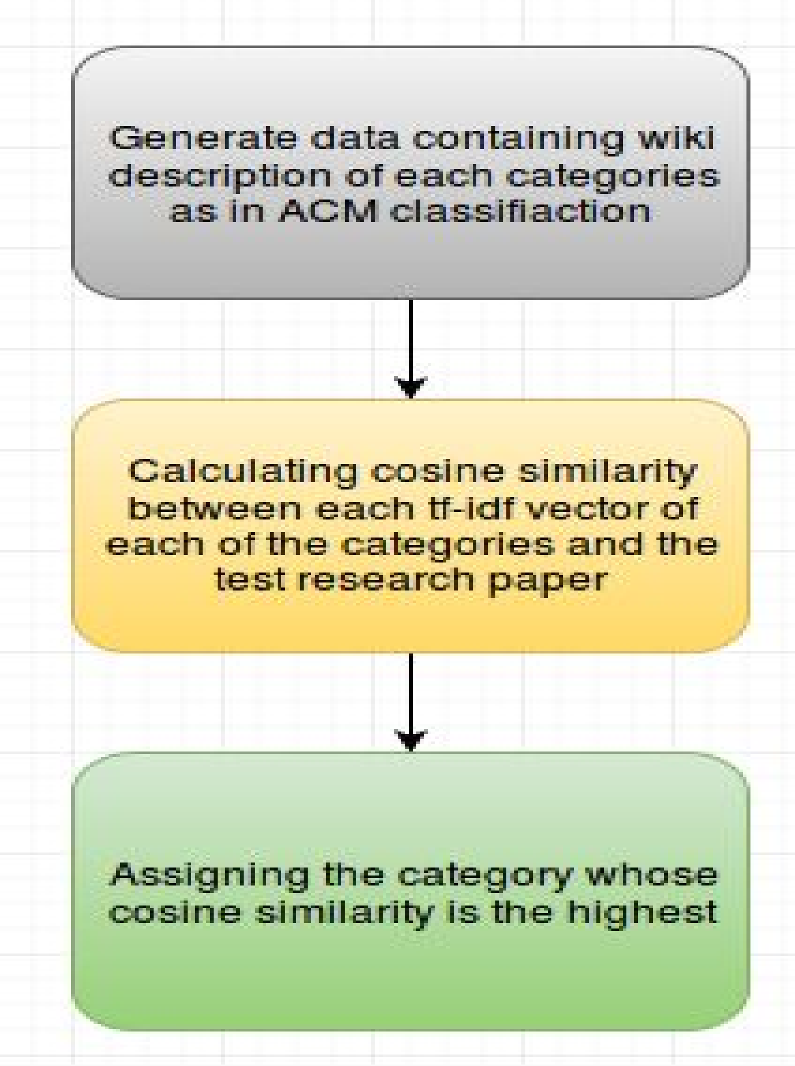
Multiple approaches were tried for the given problem

1) Cosine similarity
2) Latent Dirichlet Allocation
3) Labeled LDA + Doc2vec

Approach 1 : Cosine Similarity

Intuition Behind The Approach:

Title and abstract paper generally contain words contained in the description of the categories so finding the cosine similarity of the tf-idf vector of the title + abstract and all the categories and assigning the closest one became our first approach.



Generate data containing wiki description of each categories as in ACM classifiaction

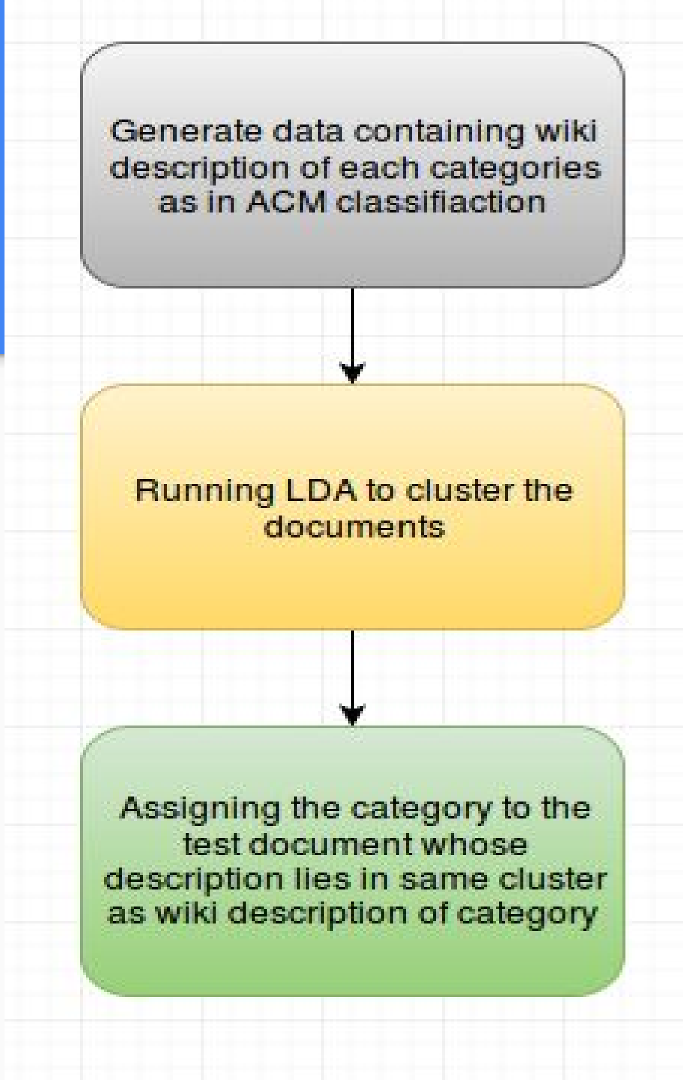
Calculating cosine similarity between each tf-idf vector of each of the categories and the test research paper

Assigning the category whose cosine similarity is the highest

Approach 2 : Latent Dirichlet Allocation

Intuition Behind The Approach:

Topic modelling - a topic model is a type of statistical model for discovering the abstract "topics" that occur in a collection of documents.
We picked the most common topic modelling algorithm - Latent Dirichlet Allocation



Generate data containing wiki description of each categories as in ACM classifiaction

Running LDA to cluster the documents

Assigning the category to the test document whose description lies in same cluster as wiki description of category

Proposed Model : Labeled LDA + Doc2Vec

Intuition :

Accuracy can only be improved by supervised topic modelling i.e. by labelling the documents with topics and taking care of the semantic distance between the research paper and the categories.

References :

Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora - Stanford University (2009)

Distributed Representations of Sentences and Document - Google Inc. (2014)

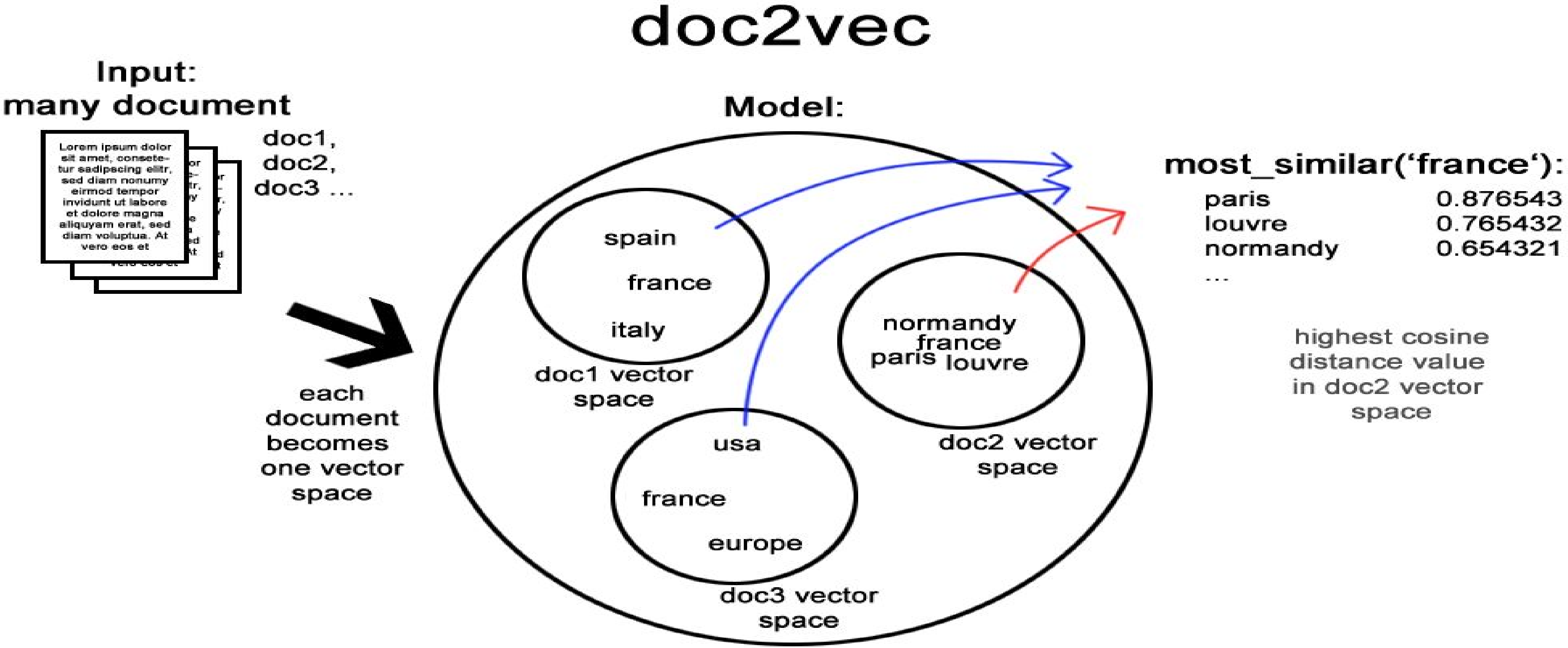
Labeled Latent Dirichlet Allocation

Labeled LDA is a probabilistic graphical model that describes a process for generating a labeled document collection. Like Latent Dirichlet Allocation, Labeled LDA models each document as a mixture of underlying topics and generates each word from one topic. Unlike LDA, L-LDA incorporates supervision by simply constraining the topic model to use only those topics that correspond to a document's (observed) label set.

Doc2Vec

Doc2Vec is a tool provided by gensim that maps each sentences to a paragraph vector. Paragraph vector is an unsupervised framework that learns continuous distributed vector representations for pieces of texts. The texts can be of variable-length, ranging from sentences to documents. The name Paragraph Vector is to emphasize the fact that the method can be applied to variable-length pieces of texts, anything from a phrase or sentence to a large document.

Doc2Vec (continued...)

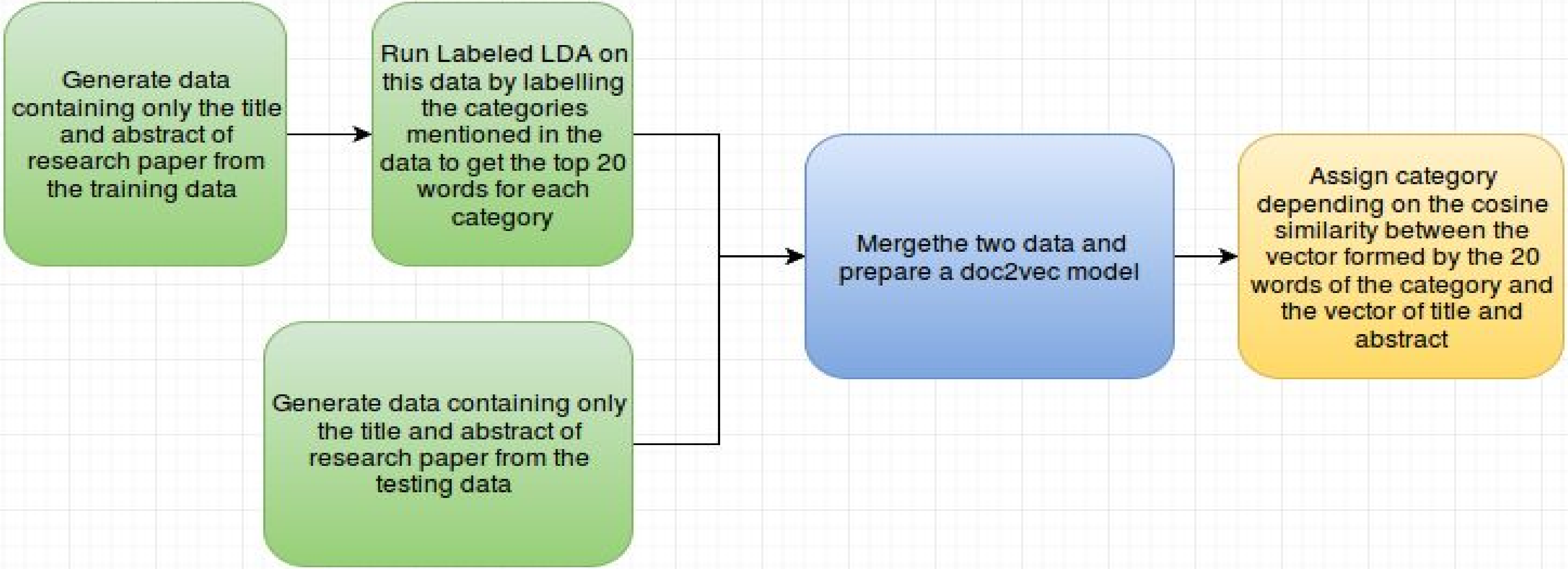

doc2vec
Input:
many document
Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et
doc1,
doc2,
doc3 ...
each document becomes one vector space
Model:
spain
france
italy
doc1 vector space
normandy
france
paris
louvre
doc2 vector space
usa
france
europe
doc3 vector space

most_similar('france'):

paris	0.876543
louvre	0.765432
normandy	0.654321
...	

highest cosine distance value in doc2 vector space

Proposed Model



Generate data containing only the title and abstract of research paper from the training data

Run Labeled LDA on this data by labelling the categories mentioned in the data to get the top 20 words for each category

Generate data containing only the title and abstract of research paper from the testing data

Mergethe two data and prepare a doc2vec model

Assign category depending on the cosine similarity between the vector formed by the 20 words of the category and the vector of title and abstract

MAP

Mean average precision for a set of queries is the mean of the average precision scores for each query.

where Q is the number of queries. $\text{MAP} = \frac{\sum_{q=1}^{Q} \text{AveP(q)}}{Q}$

In our work, the number of queries is the number of test documents and average precision is calculated by the definition: Average of the precision values at the points at which each relevant document is retrieved.

For the final model (approach 3), we are getting a MAP of 0.68 (68%).

We believe for our work and the data provided, this MAP is quite good. The results of such approaches depend a lot on the data.

Results Overview

Categories used for classification :

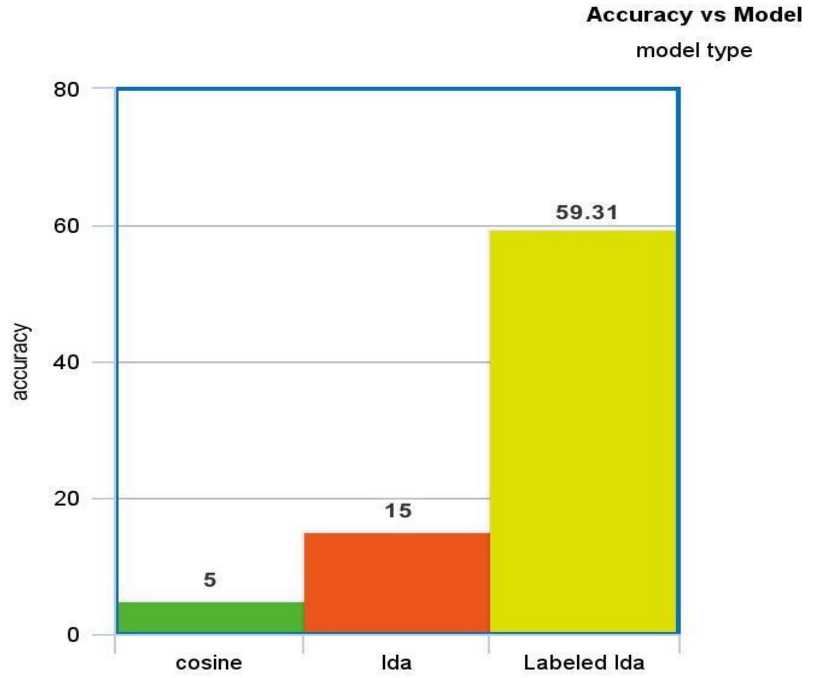
Artificial - Intelligence, Databases, Computer-Vision, Information Retrieval and Other.

Training Dataset :

No. of research papers : 549

Test Dataset:

No. of research papers : 51745
No. of research papers correctly classified : 36334
MAP score : 59.31 %



Achievement of the project

With so less training data, we got a MAP of 59.31%.

Doc2Vec and Labeled LDA are responsible for such surprisingly good results.

The model will given even better results for good training set, provided labeled as the requirement of our model.