PATENT NETWORK ANALYSIS

Project report submitted in partial fulfillment of the requirement for the degree of Bachelor of Technology

in

Computer Science and Engineering/Information Technology

By

Ajay Rana (151328)

Anmol Kamboj (151389)

Under the supervision of

Dr. Ruchi Verma

to

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology Waknaghat, Solan-173234, Himachal Pradesh

CERTIFICATE

I hereby declare that the work presented in this report entitled **" Patent Network Analysis"** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering/Information Technology** submitted in the department of Computer Science & Engineering and Information Technology**,** Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out over a period from August 2018 to May 2019 under the supervision of **Dr. Ruchi Verma(Assistant Professor-Senior Grade)**. The matter embodied in the report has not been submitted for the award of any other degree or diploma.

(Student Signature)

Ajay Rana, 151328

(Student Signature)

Anmol Kamboj, 151389

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

(Supervisor Signature)

Dr. Ruchi Verma

Assistant Professor (Senior Grade)

Computer Science & Engineering and Information Technology

Dated:

ACKNOWLEDGEMENT

It is our privilege to express our sincerest regards to our project supervisor **Dr. Ruchi Verma** (**Assistant Professor -Senior Grade)** for their valuable inputs, able guidance, encouragement, whole-hearted cooperation and direction throughout the duration of our project.

We deeply express our sincere thanks to our Head of Department **Prof. Dr. Satya Prakash Ghrera** for encouraging and allowing us to present the project on the topic **"Patent Network Analysis"** at our department premises for the partial fulfillment of the requirements leading to the award of B-Tech degree.

We are also grateful to **CSE Project lab staff** for their practical help and guidance.

Date: Ajay Rana (151328)
Anmol Kamboj (151389)

TABLE OF CONTENTS

List of Figures

ABSTRACT

Patent network analysis, an advanced method of patent analysis, is a useful tool for technology management. This method visually displays all the relationships among the patents and enables the analysts to intuitively comprehend the overview of a set of patents in the field of the technology being studied. This method provides an automated procedure for searching patent documents, extracting patent keywords, and determining the weight of each patent keyword in order to generate a sophisticated visualization of the patent network. This study proposes a detailed procedure for generating an intelligent patent network that is helpful for improving the efficiency and quality of patent analysis.

Chapter 1 INTRODUCTION

1.1 Introduction

Patent network analysis is a useful tool for technology management. This project visually displays the relationships among the patents and enables the analysts to get an overview of a set of patents in the field of the technology being studied. Based on artificial intelligence techniques, the proposed method provides an automated process for searching patent documents, extracting patent keywords, and calculate the weight of each patent keyword to generate a graph of the network.

Patents which describe main contents of technological inventions contain considerable technical knowledge. These documents are significant sources of technological data and play a critical role in the advancement and diffusion of technology. Furthermore, patent analysis transfers the patent data to systematic and valuable information that is helpful for managing research and development process, exploring technological trends, tracking technological development, and identifying technology plans. It is considered to be a useful vehicle for technology management.

This method uses several patent keywords as input to produce a visual patent network. The network demonstrates the overall relationship among all patents. First, the search for patent documents to be studied relies on the subjective judgments of analysts. Second, the collection of patent documents is a time-consuming task because it requires an exhaustive search of patent databases. The current method lacks a set of systematic and convenient patent searching procedures. As a result, the dataset of patent documents being studied is not complete. Third, the relevant patent keywords used in the current method are selected by technical experts. In reality, the technical experts often use different terminologies to describe the same technology.

Patent Network Analysis, a propelled technique for patent examination, is a valuable device for innovation the board. This strategy outwardly shows every one of the connections among the licenses and empowers the investigators to naturally fathom the outline of a lot of licenses in the field of the innovation being contemplated. Albeit patent system investigation has relative preferences not quite the same as customary techniques for patent examination, it is liable to a few pivotal impediments.

To defeat the disadvantages of the present strategy, this examination proposes a novel patent investigation technique, called the savvy patent system examination strategy, to make a visual system with extraordinary accuracy. In light of man-made reasoning systems, the proposed strategy gives a computerized method to looking patent reports, removing patent catchphrases, and deciding the heaviness of each patent watchword so as to produce an advanced representation of the patent system. This investigation proposes a nitty gritty system for producing a wise patent system that is useful for improving the effectiveness and nature of patent examination

Technological development has broadly been concentrated to advance the maintainability and updating of enterprises. The most significant late issue has been the elements of communitarian development among enterprises . In addition, a great deal of nations are advancing modern bunch arrangements that encourage collective advancement among ventures in explicit areas, and underscoring that the key is making systems among businesses. Patent information is an open and accessible information source. In fact, patent application data gives information concerning the creators and chosen ones of innovation coordinated efforts and imaginative procedures . Various investigations assessed licenses with joint university– industry possession, or communitarian licenses, to quantify the improvement patterns of university– industry communitarian (UIC) advancements around the world.

Assessing community oriented licenses encourages analysts in picking up a superior comprehension of the extension of information and the coordinated effort organizes inside advancement frameworks. A lot of examines utilized a patent system investigation strategy to watch the dispersion of UIC licenses after some time, in this manner deciding innovation advancement patterns. As of late, copious thinks about have utilized system examination strategies to research the patent cooperation and patent coordinated effort arrange in explicit ventures, particularly in nano-related enterprises.

In synopsis, patent joint effort arrange is a significant type of cooperative advancement, and agreeable development is winding up progressively visit. Numerous researchers have been worried on related research from different points of view, including coordinated effort types, cooperation attributes, joint effort inspirations, and so forth. Be that as it may, there are not many investigations concentrating on the patent coordinated effort system of the SG business. In this paper, we endeavor to investigate the patent coordinated effort arrange for industry.

Patent

A patent is a document which has the information regarding technical inventions. A patent gives its owner the advantage to ignore others from creating, utilizing, selling, and bringing in an invention for a restricted timeframe. The patent rights are conceded in return for an empowering open revelation of the development. In many nations patent rights fall under common law and the patent holder needs to sue somebody encroaching the patent so as to uphold his or her rights. In certain enterprises licenses are a basic type of upper hand; in others they are superfluous.

The method for allowing licenses and the degree of the rights given to patentee varies broadly between nations as directed by nationwide laws and global understandings. Usually, an allowed patent application must conclude at least one case that will follow the development. A patent may incorporate various cases, every one of that may characterize a particular right. The cases must meet significant patentability prerequisites, for example, value, non-conspicuousness, and curiosity.

1.2 Problem Statement

- This method is useful for managing technology.
- It graphically shows all the relationships between the patents and allows the user to get an overview of a set of patents in a particular field of technology.
- This provides us with an automated process for determining, searching, and extracting the patent keyword in order to generate a graphical representation of the patent network.

1.3 Objective

The relationship among patents can be visually demonstrated in this analysis, and the analysts are able to comprehend the overall structure of patent network.

1.4 Methodology

The main purpose of this study is to propose an automatically intelligent patent network analysis method. It contains four major stages: searching and selecting patent documents, extracting words, calculating the frequency of each patent keyword, and generating a graphical representation of the patent network. First, this study exploits the ontology of the automatic document classification process which is identified by the patent keywords agents to extract the feature subset documents. This automated technique is used to search, filter and categorize the relevant patent documents in order to collect a complete dataset of patent documents. Next, the enhanced term frequency - inverse document frequency (ETF-IDF) technique is executed to elicit the patent keywords automatically from the selected patent documents. Moreover, the Viterbi algorithm is traditionally used to detect keywords through the HMM configuration. Therefore, through using association rules which are put to combine the Viterbi algorithm with the Apriori algorithm into practice, the intelligent system produces the weighted value of each patent keyword in every patent document and further strengthens those keywords in iteratively appearing different patent documents to derive the really appropriate keywords. Finally, the sets of weighted patent keywords are employed to serve as the input base for generating a sophisticated patent network in order to effectively implement patent analysis.

1.5 Organization

Chapter 1 Highlights the basic explanation of Patent Network Analysis Technique.

Chapter 2 Review of all the literature collected from several journals, conferences and internet sites.

Chapter 3 It includes the tools used, the algorithm which is being studied and its applications.

Chapter 4 Implying the algorithms and its analysis.

Chapter 5 It details the future scope and concludes the project's implementation for future research on this project.

Chapter 2 LITERATURE SURVEY

To completely understand the topic of project and research the best possible way of the problem a bunch of scholarly journals, books and research papers have been read and used where required. Many researchers and scientists have published numerous papers from which a part of the have been mentioned below.

2.1 TITLE: Efficient Viterbi scoring architecture for HMM-based speech recognition systems.

AUTHORS: Cho, Y. S., Kim, J. Y., & Lee, H. S

YEAR OF PUBLICATION: October,2010

The Viterbi algorithm is traditionally used to detect keywords through the HMM Configuration. This approach is used to find keywords is Viterbi decoding using the HMM configuration. Every particular way in the decoder is a order of keyword and unwanted garbage elements. The decoder checks scores for all possible paths, and the one with the highest score is showed as the output. The score is a global score estimated by accumulating all likelihoods for the whole expression

2.2 TITLE: An Improved Algorithm for Mining Association Rules in Large Databases.
AUTHORS: Farah Hanna AL-Zawaidah, Marwan AL-Abed Abu-Zanona & Y.H. Jbara
YEAR OF PUBLICATION:2011

Mining association governs in extensive databases is a center subject of information mining. Finding these affiliations is gainful to the right and suitable choice settled on by chiefs. Finding incessant item sets is the key procedure in affiliation rule mining. One of the difficulties in creating affiliation rules mining calculations is the amazingly expansive number of guidelines produced which makes the calculations wasteful and makes it hard for the end clients to fathom the produced standards. This is on the grounds that most customary affiliation rule mining approaches embrace an iterative procedure to find affiliation rule, which requires very substantial estimations and a confused exchange process. Besides, the current mining calculations can't perform productively because of high and rehashed circle get to overhead. Along these lines, in this paper we present a novel affiliation rule mining approach that can productively find the affiliation manages in vast databases. The proposed methodology is gotten from the regular apriori approach with highlights added to improve information mining execution. We have performed broad analyses and contrasted the exhibition of our calculation and existing calculations found in the writing. Test results demonstrate that our approach outflanks different methodologies and demonstrate that our methodology can rapidly find visit item sets and viably mine potential affiliation rules.

2.3 TITLE: Natural Language Processing and Machine Learning: A Review.

AUTHORS: Fateme Behzadi

YEAR OF PUBLICATION:2015

Natural language processing rises as one of the most sizzling point in field of Speech and language innovation. Likewise Machine learning can grasp how to perform significant NLP assignments. This is regularly feasible and savvy where manual writing computer programs isn't. This paper endeavors to Study NLP and ML and gives bits of knowledge into the basic qualities of both. It abridges normal NLP assignments in this complete field, at that point gives a concise depiction of regular AI approaches that are being utilized for various NLP tasks.Also this paper shows an audit on different ways to deal with NLP and some related subjects to NLP and ML.

2.4 TITLE: Applying Patent Information to tracking a specific technology.
AUTHORS: Liu, C. Y., & Luo, S. Y
YEAR OF PUBLICATION:2007

Patents when all is said in done contain much novel innovative data. This paper exhibits that the use of patent examination can encourage an exceptional plan for following innovation improvement. In this paper, the strolling strategy of the Japanese biped robot is followed for instance. The looking strategy for the FI (ecord file) and F-term characterization framework created by JPO (Japan Patent Office) was utilized in this examination, where all the related patent information were sought from the IPDL (Intellectual Property Digital Library). This examination explored a significant system connected to the humanoid biped robot that impersonates the strolling conduct of the people on two legs. By dissecting the patent data acquired, the relative research capacities, specialized qualities, and patent reference conditions among patent contenders were thought about. Moreover, a defined specialized framework of patent guide is built up in this paper to show that the ZMP (Zero Moment Point) control implies is the principle innovation to accomplish balanced out strolling control of the humanoid biped robot. This examination additionally joins important scholarly diary discoveries and mechanical data. Results displayed in this show licenses can work not just as a guide for following an innovation direction, yet in addition as a manual for the fundamental improvement of another innovation in years to come.

2.5 TITLE: A Re-Examination of Text Categorization Methods.

AUTHORS: Yiming Yang, Xin Liu

YEAR OF PUBLICATION:2007

This paper reports a controlled report with factual centrality tests on ve content classification techniques: the Support Vector Machines (SVM), a k-Nearest Neighbor (kNN) classifer, a neural system (NNet) approach, the Linear Least squares Fit (LLSF) mapping and a Naive Bayes (NB) more tasteful. We center around the heartiness of these techniques in managing a skewed class circulation, and their exhibition as capacity of the preparation set classification recurrence. Our outcomes demonstrate that SVM, kNN and LLSF essentially beat NNet and NB when the quantity of positive preparing occurrences per classification are little (under ten), and that every one of the techniques perform equivalently when the classifications are adequately normal.

Chapter 3 SYSTEM DEVELOPMENT

In this section the various decision were made for the design and implementation of many methods.

3.1 Design

The process of development of software includes various discussion and research regarding design of the project.The aspects of project are discussed here.

3.1.1 Block Diagram Of System

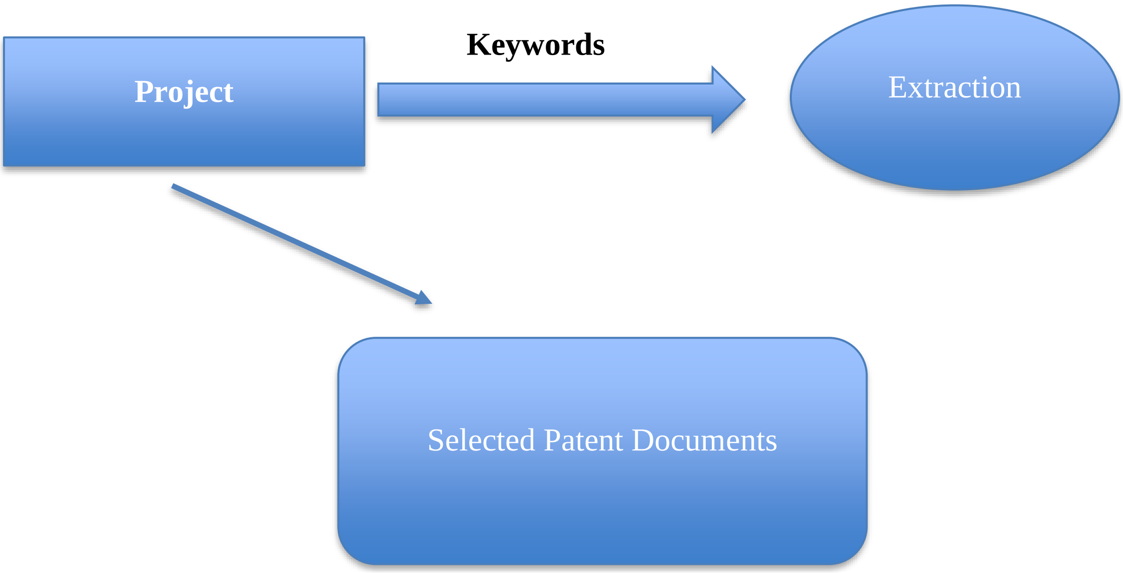


Fig.3.1 Selection of Patent Document.

Explanation:

In this step the first thing is to select a patent document or number of documents that needed to be analysed through the project. Now the selected patents are being studied and keywords are extracted in second step. The concept-based document searching method can be adopted to correctly classify the patent documents that belong to the field of technology being studied.

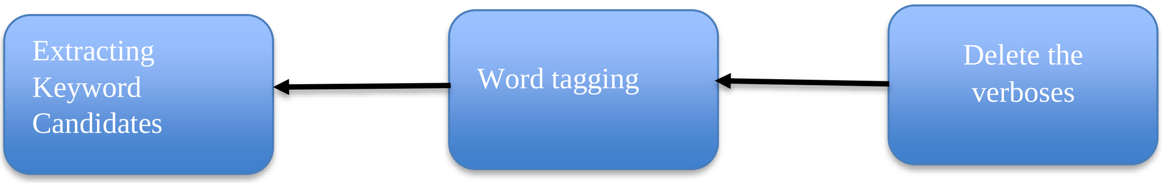


Fig.3.2 Extracting Keyword Candidates

Step 1: Delete the verbose

This step segments the sentence according to different signs, ex: comma mark, full stop mark and period mark. Then, it constructs up a syntax representation tree and deletes all extra words in each sentence.

Step 2: Word tagging

In this step, the program processes the word tagging. We added to its lexicon as references for many domain similar words to enhance the tagging result in order to get a syntax parse

Step 3: Punctuation marks processing

It segments sentences by punctuation marks, it can be achieved to get better results if the main different marks are dealt with and handled. Three types of punctuation marks may change the structure of sentences and should be refined in the processing to upgrade the understandings of context meanings in a sentence.

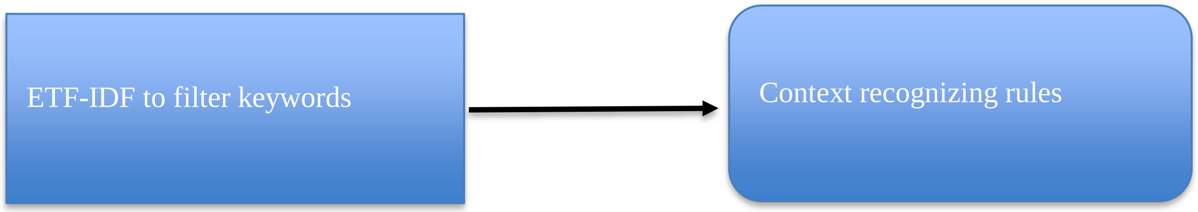


Fig.3.3 Filtering keywords.

Enhanced term frequency – inverse document frequency (ETF-IDF) and context recognizing rules

The ETF-IDF counts the frequency of each word in order to retrieve the meaningful words and compares a query vector with a document vector using a similarity or distance function, such as the cosine similarity function.

Step 1: Problem setting

This study addresses the problem of automatic extraction of semantic similarity relations among lexical items in relational form from which fine grained hierarchical clusters are obtained in the patent tree. In order to restrict the vocabulary and word ambiguity as well as to utilize information in abundant patent texts, this processing is confined to corpora from specific patent domains. This restriction is acceptable in the framework of Natural Language Processing (NLP) systems, which usually operate on sub-languages and are interested only in domain specific word meanings. Therefore, this process aims at developing a method applicable to every domain for which specific corpora are available in order to extract domain independent word meaning relations. Thus, this process can provide the semantic relations of the filtered keywords in relevance to thematic domains as well.

Step 2: Context similarity estimation

Counting the number of occurrences of every semantic token found in the corpus, a frequency threshold under which no semantic clustering is attempted can be defined. Therefore, only Frequent Semantic Entities (FSE) are subjected to clustering (except the FSEs represented in the corpus by known patterns) while all but the rarest semantic tokens are used as clustering parameters. The corresponding frequency thresholds in the present experiments were set to 20 and 10 respectively in order to acquire sufficient contextual data for every FSE constraining computational time. Ideally, any word appearing at least twice in the corpus should be used as a context parameter. Definite determiners and verb auxiliaries are excluded from the processing because they have no semantic connection with their head words while pronouns are handled as semantically empty words.

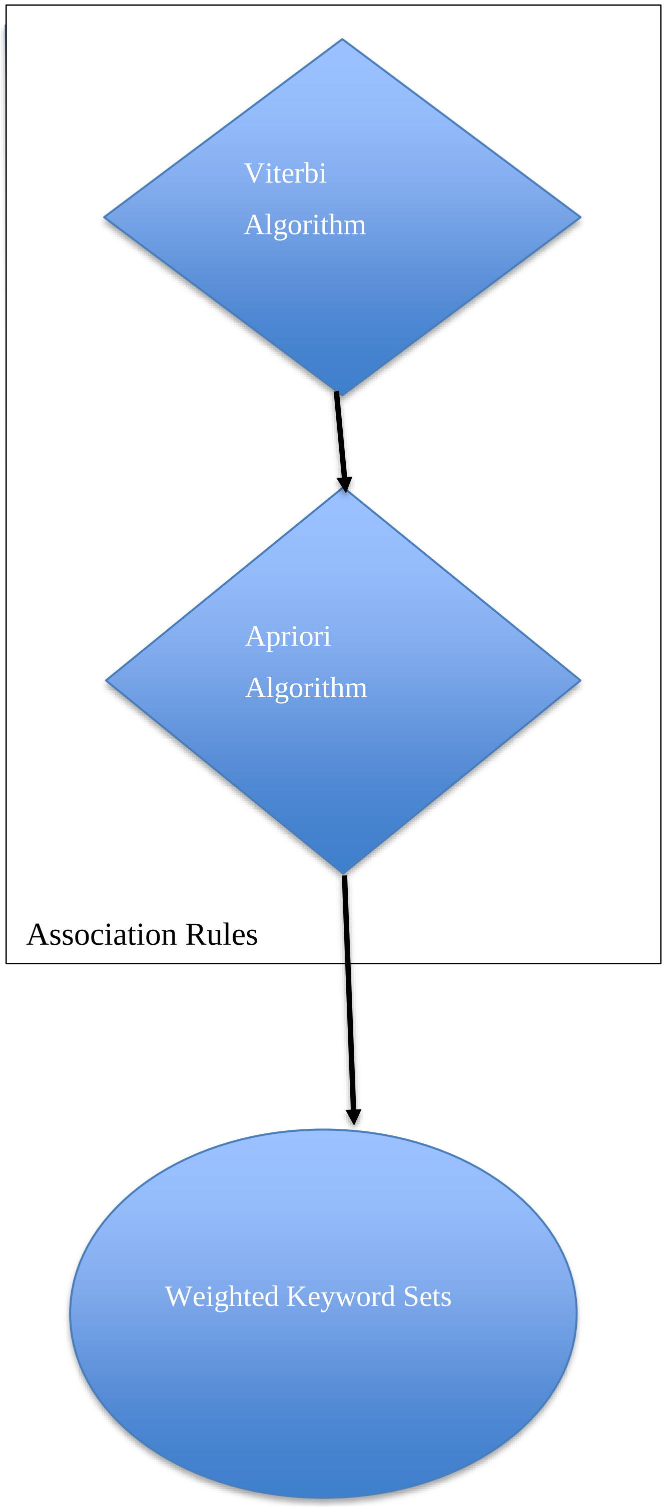


Fig.3.4 Algorithms.

Viterbi: The traditional method for detecting keywords is Viterbi decoding by HMM configuration. Every path in the Viterbi decoder is a order of garbage elements and keywords. The decoder finds values for every path possible, and selects the one with highest value as the output. Combined probability of the path and the other vectors are related to the final value. This approach is concerned about the keyword detection task. The final value is a global score estimated by observing all possibilities for the complete expression.

Apriori: Apriori is used to perform on data stes containing transactions (for example, collection of record on websites visitors). The algorithm is used to find subsets which are common to at least a threshold value of the item sets.

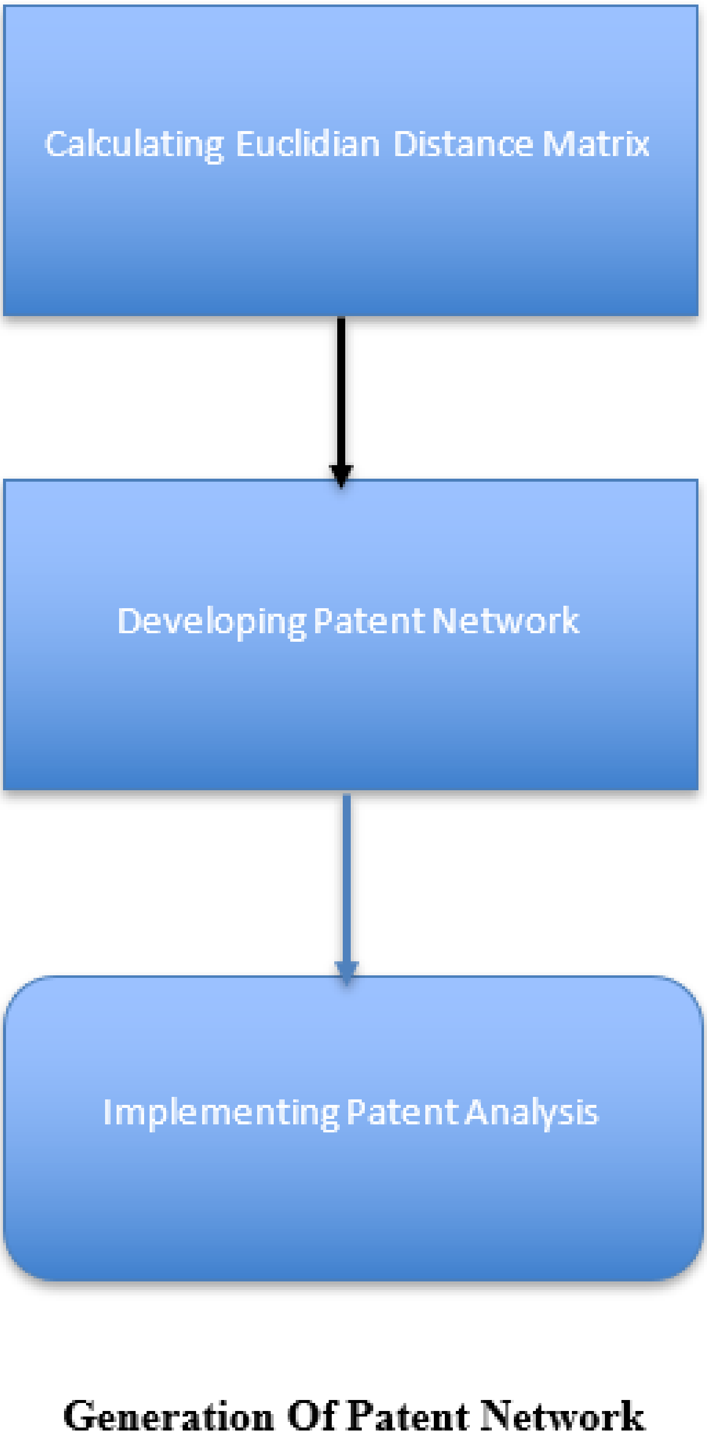

Calculating Euclidian Distance Matrix
Developing Patent Network
Implementing Patent Analysis
Generation Of Patent Network

Fig.3.5 Generation of Patent Network.

In this stage, several techniques are employed to generate the patent network. The detailed content is described as follows:

Step 1: Counting the occurrence frequency of keywords in each patent document and then the weighted value of each keyword multiplied by the occurrence frequency to generate the weighted occurrence frequency of keywords in each patent document. The final values of each patent are integrated into keyword vectors as below:

Patent 1: $(p11, p12, p13, L, p1n)$

Patent 2: $(p21, p22, p23, L, p2n)$

Patent m: $(pm1, pm2, pm3, L, pmn)$

For example, $p11$ is the weighted occurrence frequency of the first keyword in the Patent 1.

Step 2: Utilizing Euclidian distance to calculate the distance among the patents and to establish the relationship among patents.

Step 3: Transforming the real values of *Ed* matrix into the standardized values of *Es* matrix in order to graph the patent network for next procedure.

Step 4: The cell of the *Es* matrix must be a binary transformation, comprising 0s and 1s if it is to exceed the cut-off value *q*.

3.2 Libraries

NumPy

NumPy is a library for the Python programming language, including support for extensive, multi-dimensional exhibits and lattices, alongside an expansive gathering of abnormal state scientific capacities to work on these clusters.

NumPy is the crucial bundle for logical registering with Python. It contains in addition to other things:

- an amazing N-dimensional cluster object
- advanced (telecom) capacities
- apparatuses for coordinating C/C++ and Fortran code
- helpful direct variable based math, Fourier change, and irregular number abilities

PyPDF2 for Python

PyPDF2 is an unadulterated python PDF library equipped for part, combining, trimming, and changing the pages of PDF documents. It can likewise include custom information, seeing alternatives, and passwords to PDF documents. It can recover content and metadata from PDFs just as consolidation whole documents together.

The PyPDF2 bundle is an unadulterated Python PDF library that you can use for part, blending, trimming, and changing pages in your PDFs. As indicated by the PyPDF2 site, you can likewise utilize PyPDF2 to include information, seeing alternatives, and passwords to the PDFs, as well. At long last, you can utilize PyPDF2 to remove content and metadata from your PDFs.PyPDF2 is really a fork of the first PyPdf, which was composed by Mathiew Fenniak and discharged in 2005. Nonetheless, the first PyPdf's last discharge was in 2014. An organization called Phaseit, Inc. talked with Mathieu and wound up supporting PyPDF2 as a fork of PyPdf.

Matplotlib

Matplotlib is a Python 2D plotting library which produces creation quality figures in an arrangement of printed duplicate positions and shrewd conditions transversely over stages. Matplotlib tries to make basic things straightforward and hard things possible.

You can create plots, histograms, control spectra, reference diagrams, error charts, scatterplots, etc., with just two or three lines of code.

For direct plotting the pyplot module gives a MATLAB-like interface, particularly when united with IPython. For the power customer, you have full control of line styles, content style properties, tomahawks properties, etc, by methods for a thing organized interface or through a ton of limits normal to MATLAB customers.

3.3 Tools

Python 3

Python is an abnormal state, translated, intuitive and object-arranged scripting language. Python is intended to be exceptionally coherent. It utilizes English watchwords every now and again though different dialects use accentuations. Python 3.0 was discharged in 2008. Despite the fact that this variant should be in reverse incompatibles, later on a large number of its significant highlights have been backported to be good with adaptation 2.7 It has less grammatical developments than different dialects.

- Python is Interpreted
- Python is Interactive
- Python is Object-Oriented
- Python is a Beginner's Language

IDE Anaconda

Anaconda constrictor is a logical Python dissemination. It has no IDE of its own. The default IDE packaged with Anaconda is Spyder which is simply one more Python bundle that can be introduced even without Anaconda.

Anaconda constrictor packages an entire group of Python bundles that are usually utilized by individuals utilizing Python for logical processing as well as information science. It gives a solitary download and an introduce program/content that introduce every one of the bundles in one go. Substitute is to introduce Python and exclusively introduce all the required bundles utilizing pip. Furthermore, it gives its very own bundle chief (conda) and bundle store. In any case, it permits establishment of bundles from PyPI utilizing pip if the bundle isn't in Anaconda storehouses.

It is particularly great in the event that you are introducing on Microsoft Windows as it can without much of a stretch introduce bundles that would somehow or another expect you to introduce C/C++ compilers and libraries in the event that you were utilizing pip. It is unquestionably an additional favorable position that conda, notwithstanding being a bundle chief, is likewise a virtual domain director enabling you to introduce free improvement conditions and change from one to the next (like virtualenv).

There is an insignificant Anaconda Python without every one of the bundles, called Miniconda. In the wake of introducing miniconda, you can utilize conda and introduce just those logical bundles that you wish and keep away from an enlarged establishment.

NLTK

The Natural Language Toolkit is a suite of libraries and projects for emblematic and measurable regular language handling (NLP) for English written in the Python programming language. NLTK is proposed to help research and educating in NLP or firmly related regions, including observational etymology, psychological science, computerized reasoning, data recovery, and machine learning.

NLTK has been utilized as showing device, as an individual investigation instrument, and as a stage for prototyping and building research frameworks. Discovering collocations requires first ascertaining the frequencies of words and their appearance with regards to different words. Frequently the accumulation of words will at that point requiring sifting to just hold helpful substance terms.

3.4 ALGORITHMS

Natural Language Processing

Synopsis structure just given. Characteristic language preparing (NLP) is a noteworthy zone of man-made reasoning exploration, which in its turn fills in as a field of use and association of various other customary AI zones. As of not long ago, the concentration in AI applications in NLP was on learning portrayal, sensible thinking, and limitation fulfillment - first connected to semantics and later to the syntax.

Normally, it is open to learn and improve techniques that comprise the center of present day AI, most remarkably hereditary calculations and neural systems. In this paper we give a diagram of the present patterns in NLP and talk about the potential uses of customary AI procedures and their mix in this interesting region.

NLP Steps for a PDF file:

- Step 1: Sentence Segmentation
- Step 2: Word Tokenization
- Step 3: Predicting Parts of Speech for Each Token
- Step 4: Text Lemmatization
- Step 5: Identifying Stop Words
- Step 6: Dependency Parsing
- Step 7: Finding Noun Phrases
- Step 8: Named Entity Recognition (NER)
- Step 9: Coreference Resolution

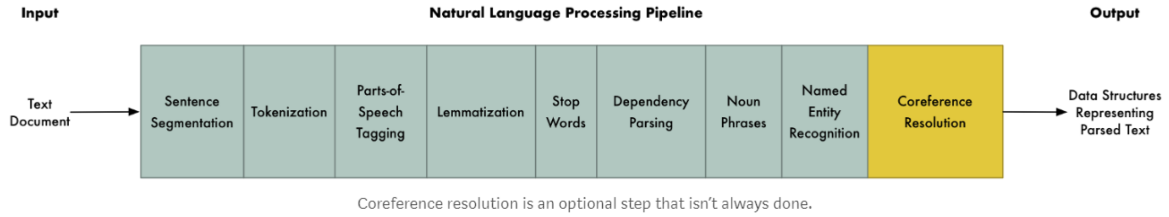


Fig.3.6 NLP Pipeline.

Step 1: Sentence Segmentation

The first step is to divide the text into different sentences. That gives us this:

1. “London is the capital and most populous city of England and the United Kingdom.”
2. “Standing on the River Thames in the south east of the island of Great Britain, London has been a major settlement for two millennia.”
3. “It was founded by the Romans, who named it Londinium.”

We know that every sentence has a different meaning or thought. It will be easier for us to develop a program to work on a single sentence rather than a complete section.

The point at which you see an accentuation mark, writing a program for Sentence Segmentation is also as simpler as partly separated sentences. Yet, advance systems are regularly used by current NLP pipelines that work in spite of improperly arranged archive.

Step 2: Word Tokenization

Since we've part our work divided into sentences, they can be processed turn by turn. We must start with our record’s primary sentence:

“London is the capital and most populous city of England and the United Kingdom.”

Now divide the sentence into independent words in a process known as tokenization. This is the outcome:

“London”, “is”, “the”, “capital”, “and”, “most”, “populous”, “city”, “of”, “England”, “and”, “the”, “United”, “Kingdom”, “.”

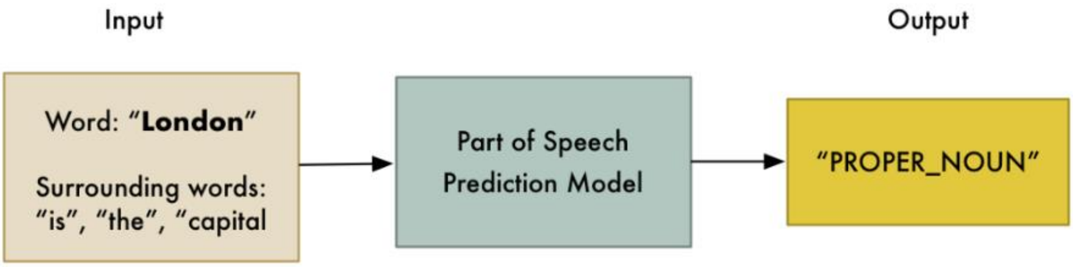
Tokenization is hard to be done in English language. We will now part words separated at a point where there is a blank space among them. Moving further, accentuation checks will be treated as solitary tokens since accentuation likely has meaning.

Step 3: Predicting Parts of Speech for Each Token

Moving further, we will take a look at each and every token and part of speech is figured out—for which it is an action word, an item, an identifier or anything else. Now we are familiar that every word in the sentence will allow us to start making sense of what is discussed in sentence.

Initially, the grammatical form model was nourished great. Different English sentences with functionally labelling every word's grammatical feature and having it conclude how to repeat it.

The model is completely based on statistical data—it does not really process the meaning of word similarly like humans. It only knows how to figure a grammatical form based on related sentences and words that it has seen.


Input
Output
Word: "London"
Surrounding words: "is", "the", "capital
Part of Speech Prediction Model
"PROPER_NOUN"

Step 4: Text Lemmatization

Words can appear in various forms in English. Like the below two sentences:

I had a **pony**.

I had two **ponies**.

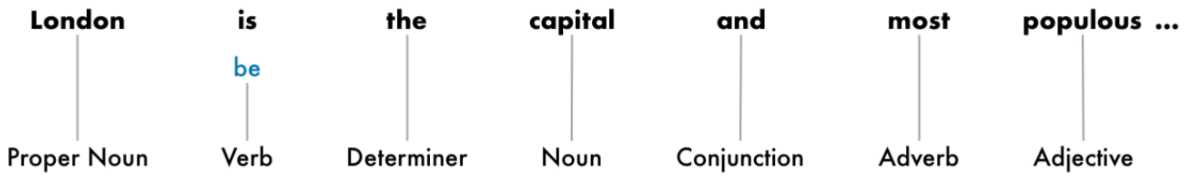
Considering **pony** in both sentences**,** different inflections are used. While programming with text in computer, each word's base form should be known so we know that same concept is talked in both of the sentences. Else, the strings "ponies" and "pony" seem like two disjoint words to a computer.

In NLP, the process is called *lemmatization*—figuring out the basic of the form or *lemma* of sentence's every word.

Similar things can be applied to verbs. Verbs can also be lemmatized by finding their incoherent form or by its root. Now "**I had two ponies**" will be "**I [have] two [pony].**"

Lemmatization can be done, by making a look-up table of the lemma forms of words depending upon their part of speech and customized set of rules to take care of words that you might have never seen before.

Here is what our sentence will be looking like after the root form of our verb is lemmatized:

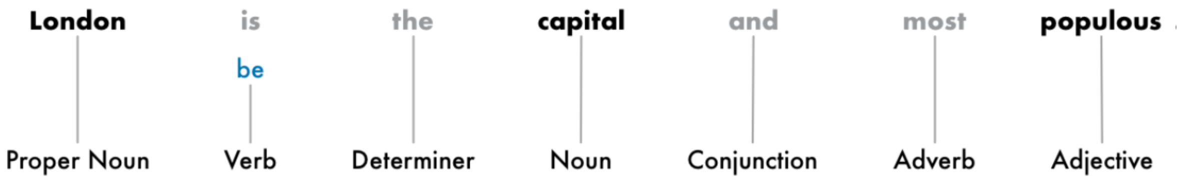


Turning "is" into "be" was the only change made by us.

Step 5: Identifying Stop Words

We need to think about significance of a every word in the sentence. In English, filler words that seem in all respects as often as possible like "and", "the", and "a". A great commotion is shown by the words such that they more frequent as possible than different words. A few NLP pipelines will be signaled as stop words — the words that are needed to shift through before any factual examination is done.

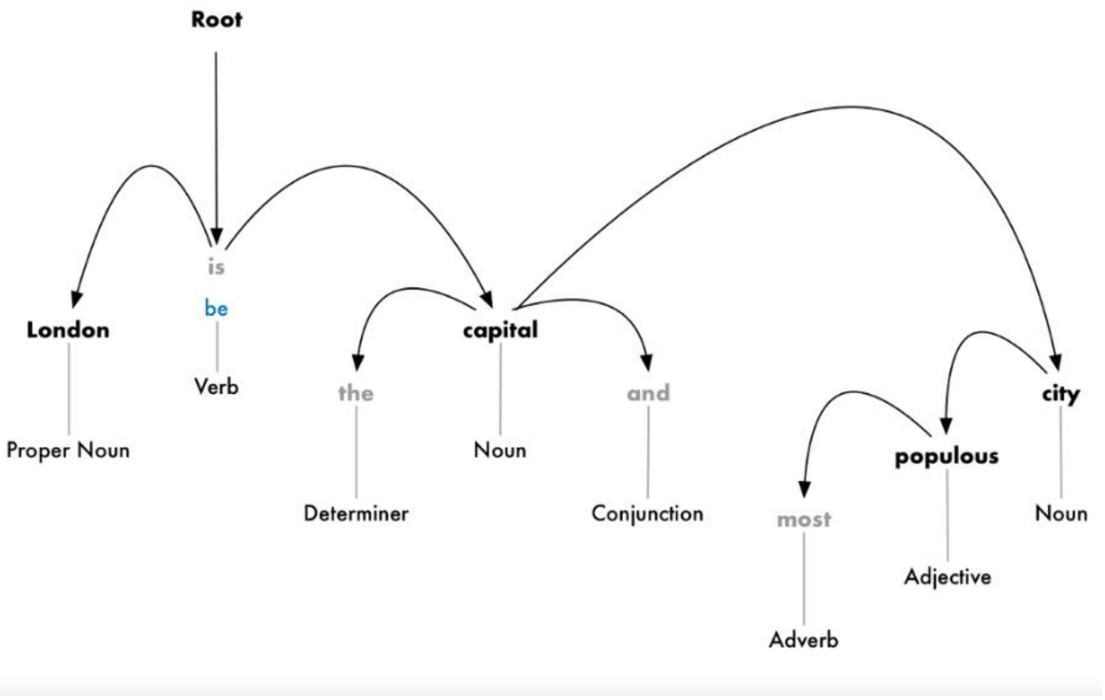
Stop words are recognized generally by checking of realized stop words. There is no default rundown of stop words fitting for each application. The rundown of words to overview can shift upon unpredicted circumstances on the application.



Step 6: Dependency Parsing

To find out how words in the sentence are related. This is called *dependency parsing*.

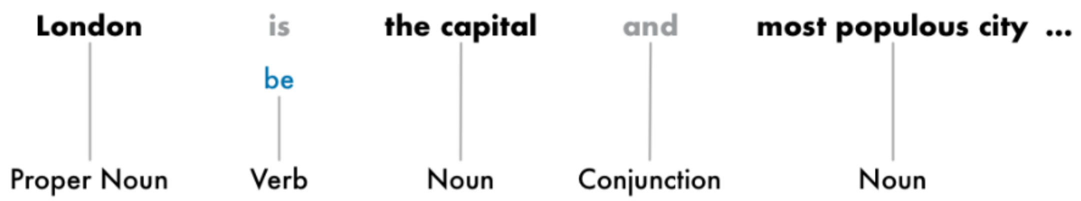
The objective is to appoint a single **parent** word to every word in the sentence by constructing a tree. Most important verb will be the root of the tree. Here is how it will be like:



Sentences are argueable and very hard to parse. In that case, a guess will be made by model based upon already parsed version of the sentence, looks likely but it is not perfect and on few occasions the model can be wrong. By with passage of time, NLP models are getting better at parsing the text in a more sensible way.

Step 7: Finding Noun Phrases

Till now, we've checked each word in the sentence as a some other substance. In any case, now and again it bodes good to pick the words that speak to a solitary thought or thing. We can use the data from the reliance parse tree to naturally combine words that are for the most part discussing something very similar.



Step 8: Named Entity Recognition (NER)

Now that we’ve done all that hard work, we can finally move beyond grade-school grammar and start actually extracting ideas.

N*ER* is there to detect and mark these nouns with some real-world concepts that they represent.

Step 9: Coreference Resolution

Now we have representation to our sentence and know the parts of speech of each keyword and how are they related to each other.

Viterbi Algorithm:

The ordinary way to deal with recognize catchphrases is Viterbi translating through the HMM design. Every way in the decoder is an arrangement of catchphrase and trash components. The decoder discovers scores for every conceivable way, and the one with the most astounding score is chosen as the yield. This score is identified with the joint likelihood of the way and the component vectors. This scoring approach concerns the watchword spotting task. The score is a worldwide score evaluated by gathering all probabilities for the entire articulation. The score isn't standardized concerning the likelihood of the acoustic perception and in this way is with respect to the specific acoustic perception space. For instance, it very well may be identified with the length of the expression, the length and number of catchphrases and waste components, the numerical range for estimations of confirmations, and so on. The estimations of these scores are punished by changing watchword and waste passageway, punishments, which are powerful spotting edges in this methodology. There is no important elucidation for the passageway punishment esteems, and they ought to be balanced experimentally to improve the exhibition criteria. This suggests for every watchword there ought to be an adequately expansive improvement or preparing set. Hopefully we will locate a sensible limit dependent on catchphrase qualities, for example, length, which can be known apriori or effectively assessed or estimated as opposed to modifying in an improvement set.

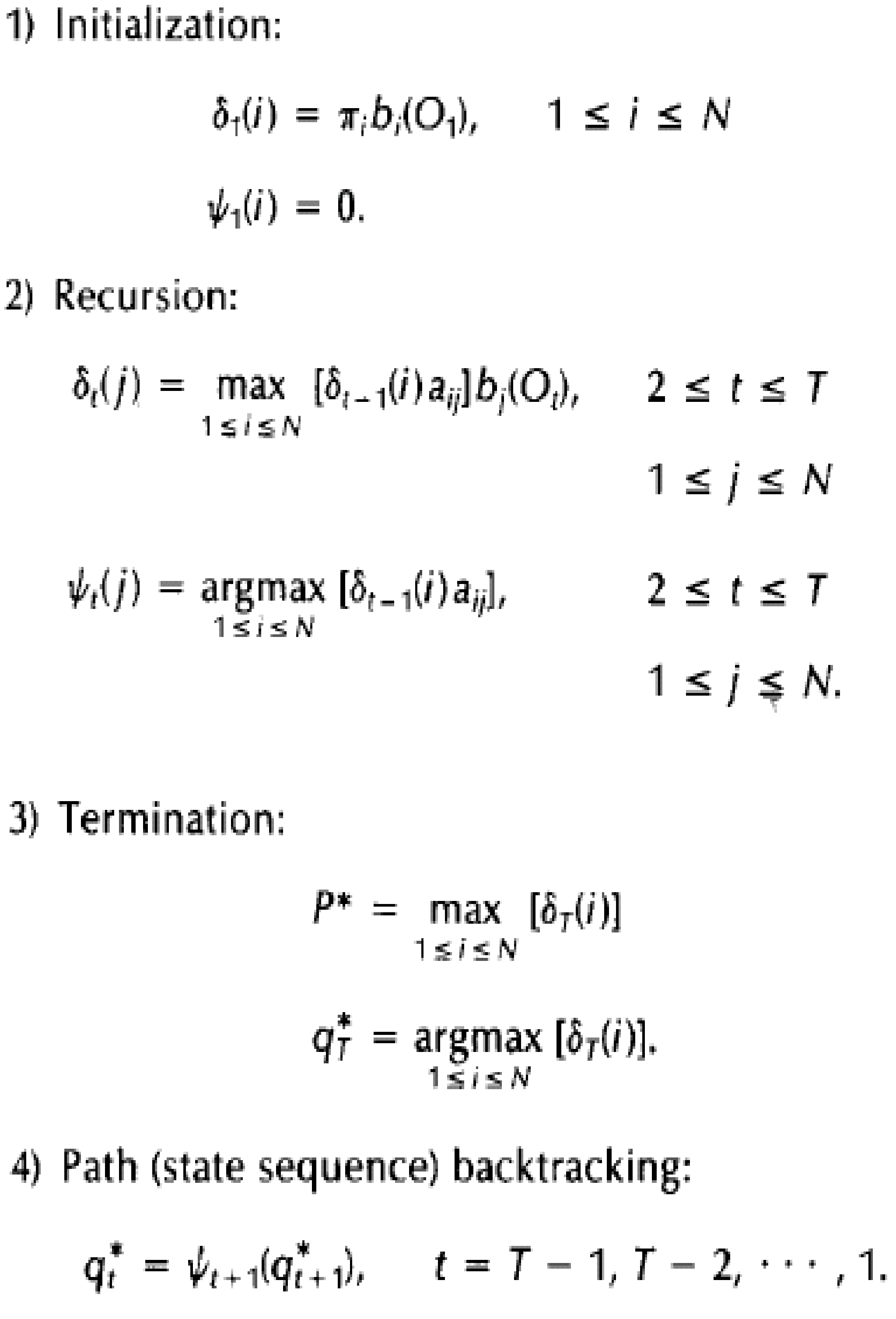
Define $\delta_t(i)$ such that,

$$\delta_t(i) = \max_{q_1, q_2, \cdots, q_{t-1}} P[q_1 \, q_2 \cdots q_t = i, \, O_1 \, O_2 \cdots O_t | \lambda]$$

i.e. the sequence which has the best joint probability so far.

- By induction, we have,

$$\delta_{t+1}(j) = [\max_i \delta_t(i) a_{ij}] \cdot b_j(O_{t+1}).$$



1) Initialization:

$$\delta_1(i) = \pi_i b_i(O_1), \qquad 1 \le i \le N$$

$$\psi_1(i) = 0.$$

2) Recursion:

$$\delta_t(j) = \max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}] b_j(O_t), \qquad 2 \le t \le T$$
$$1 \le j \le N$$

$$\psi_t(j) = \operatorname*{argmax}_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}], \qquad 2 \le t \le T$$
$$1 \le j \le N.$$

3) Termination:

$$P^* = \max_{1 \le i \le N} [\delta_T(i)]$$

$$q_T^* = \operatorname*{argmax}_{1 \le i \le N} [\delta_T(i)].$$

4) Path (state sequence) backtracking:

$$q_t^* = \psi_{t+1}(q_{t+1}^*), \qquad t = T-1, T-2, \cdots, 1.$$

Basic Concept:

Let us consider the binary case:

- 2 branches arrive at each node
- 2 branches leave each node
- All the paths going through 1 node use one of the 4 possible paths.

If the best path goes through one node, it will arrive by the better of the 2 arriving branches.

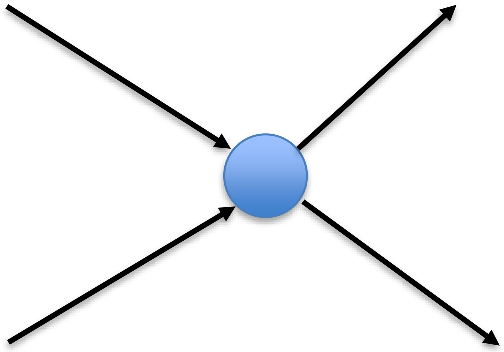


Fig.3.7 Viterbi graph.

The receiver keeps only one path, among all the possible paths at the left of one node.

- This best path is called the survivor.

For each node the receiver stores at time k:

- the cumulated distance from the origin to this node
- the number of the surviving branch.

There are 2 steps in the Viterbi algorithm

- A left to right step from k=1 to k=K in which the distance calculations are done
- Then a right to left step called traceback that simply reads back the results from the trellis.

The left to right step from k=1 to k=K:

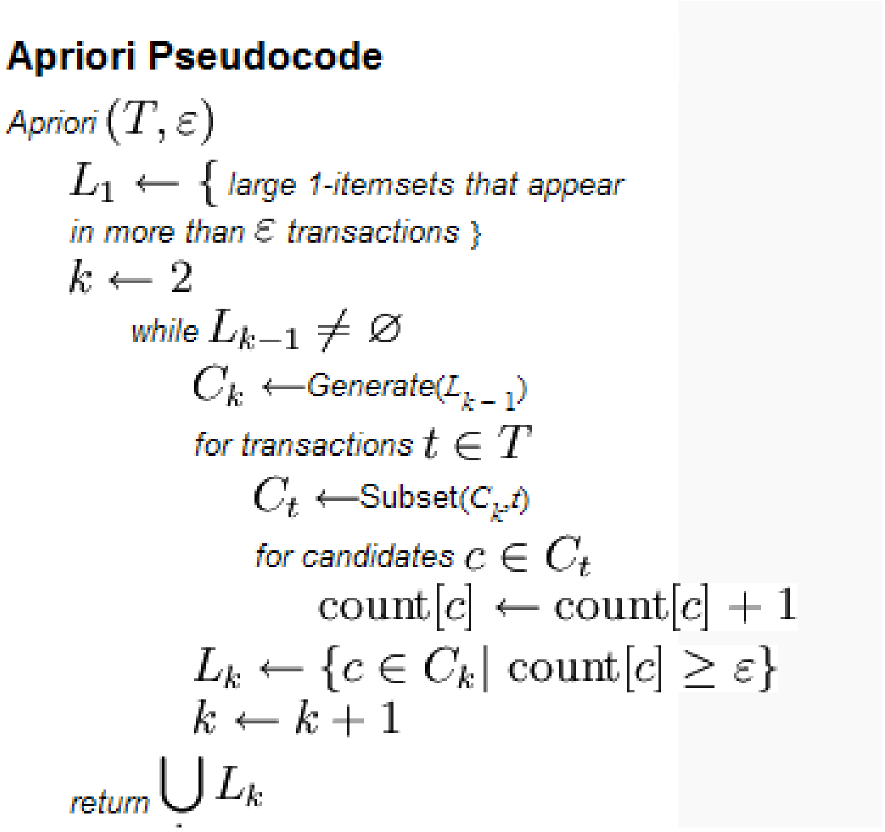
- For each stage k and each node, calculate the cumulated distance D for all the branches arriving at this node.
- Distance calculations are done recursively:
    - The cumulated distance at time k for a node i: D(k,i) reached by 2 branches coming from nodes m and n is the minimum of:
    - D(k-1,n) + d(n,i)
    - D(k-1,m) + d(m,i)
    - Where d(n,i) is the local distance on the branch from node n at time k-1 to node i at time k.
    - $d(n,i)=(Yk-Sk(n,i))^2$ where Sk(n,i) is the output when going from node n to node i.

Apriori algorithm: Developed by Agrawal and Srikant 1994

- Innovative way to find association rules on large scale, allowing implication outcomes that consist of more than one item
- Based on minimum support threshold (already used in AIS algorithm)
- Three versions:
  - Apriori (basic version) faster in first iterations
  - AprioriTid faster in later iteratons
  - AprioriHybrid can change from Apriori to AprioriTid after first iterations

Use of Apriori algorithm:

- Initial information: transactional database D and user-defined numeric minimun support threshold min_sup
- Algortihm uses knowledge from previous iteration phase to produce frequent itemsets
- This is reflected in the Latin origin of the name that means ”from what comes before”



Apriori Pseudocode

```
Apriori $(T, \varepsilon)$
    $L_1 \leftarrow \{$ large 1-itemsets that appear in more than $\varepsilon$ transactions $\}$
    $k \leftarrow 2$
        while $L_{k-1} \neq \varnothing$
            $C_k \leftarrow$Generate($L_{k-1}$)
            for transactions $t \in T$
                $C_t \leftarrow$Subset($C_k$,$t$)
                for candidates $c \in C_t$
                    $\text{count}[c] \leftarrow \text{count}[c] + 1$
            $L_k \leftarrow \{c \in C_k \mid \text{count}[c] \geq \varepsilon\}$
            $k \leftarrow k + 1$
    return $\bigcup_k L_k$
```

Key Concepts :

- Frequent Itemsets: The sets of item which has minimum support (denoted by L_i for i^{th}-Itemset).
- Apriori Property: Any subset of frequent itemset must be frequent.
- Join Operation: To find L_k , a set of candidate k-itemsets is generated by joining L_{k-1} with itself.

- Find the frequent itemsets: the sets of items that have minimum support
  - A subset of a frequent itemset must also be a frequent itemset
    - i.e., if {AB} is a frequent itemset, both {A} and {B} should be a frequent itemset
  - Iteratively find frequent itemsets with cardinality from 1 to k (k-itemset)
- Use the frequent itemsets to generate association rules.

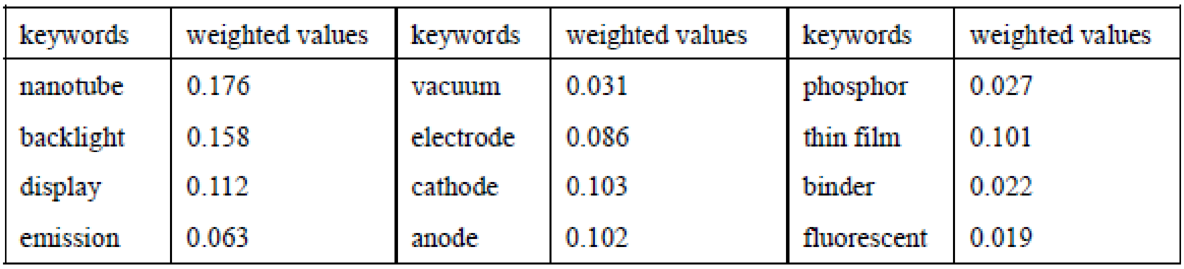
.

Enhanced term frequency – inverse document frequency (ETF-IDF) and context recognizing rules

In this investigation, we center around to revise the term recurrence - converse report recurrence (TF-IDF) to reinforce those increasingly significant catchphrases which ought to have the higher weighting esteems. Thus, the ETF-IDF calculation is overhauled from TF-IDF by thinking about the general significance of every keyword in each patent record. TF-IDF is the broadest weighting innovation which has connected to order the content orders in data recover. The TF-IDF work figures the heaviness of every vector segment (every one of them identifying with an expression of the vocabulary) of each report on the accompanying premise. To start with, it joins the word recurrence in the record. Along these lines, the more a word shows up in a report (e.g., its term recurrence (TF) is high), the more it is evaluated to be huge in this patent record. What's more, along these lines, IDF estimates how rare a word is in all patent record set and its esteem can be sensibly evaluated. Consequently, if a word is visit in an archive set, the IDF isn't accepted to be especially illustrative of this archive since it happens in most patent reports, for example, stop words, etc. Despite what might be expected, on the off chance that a word is rare in the archive set, it is viewed as pertinent for the record in the field.

Consequently, by utilizing recurrence tallying, the TF-IDF can distinguish the patent watchwords and to diminish a few mix-ups in the separating watchwords process. Despite the fact that the TF-IDF technique can distinguish the catchphrases from the patent record, it can't protect that the chose catchphrases are the best agent proficient words. In other words, the patent watchword through our ETF-IDF sifting procedure can be increasingly reasonable and truly catchphrases, so the upgraded TF-IDF calculation is utilized to improve these disadvantages of the first TF-IDF.

For Example: This shows the weights of keywords from a patent.



keywords	weighted values	keywords	weighted values	keywords	weighted values
nanotube	0.176	vacuum	0.031	phosphor	0.027
backlight	0.158	electrode	0.086	thin film	0.101
display	0.112	cathode	0.103	binder	0.022
emission	0.063	anode	0.102	fluorescent	0.019

Chapter-4 PERFORMANCE ANALYSIS

The data has been taken as an example document made by ourselves due to unavailibility of open source patent document. The pdfs contaiins various words in paragraphs or list of words which is being extracted by the program. The following result shows the graph of the most commonly used keyword in the document and has drawn the comparisn. Also it shows the most occuring word among the comparison.

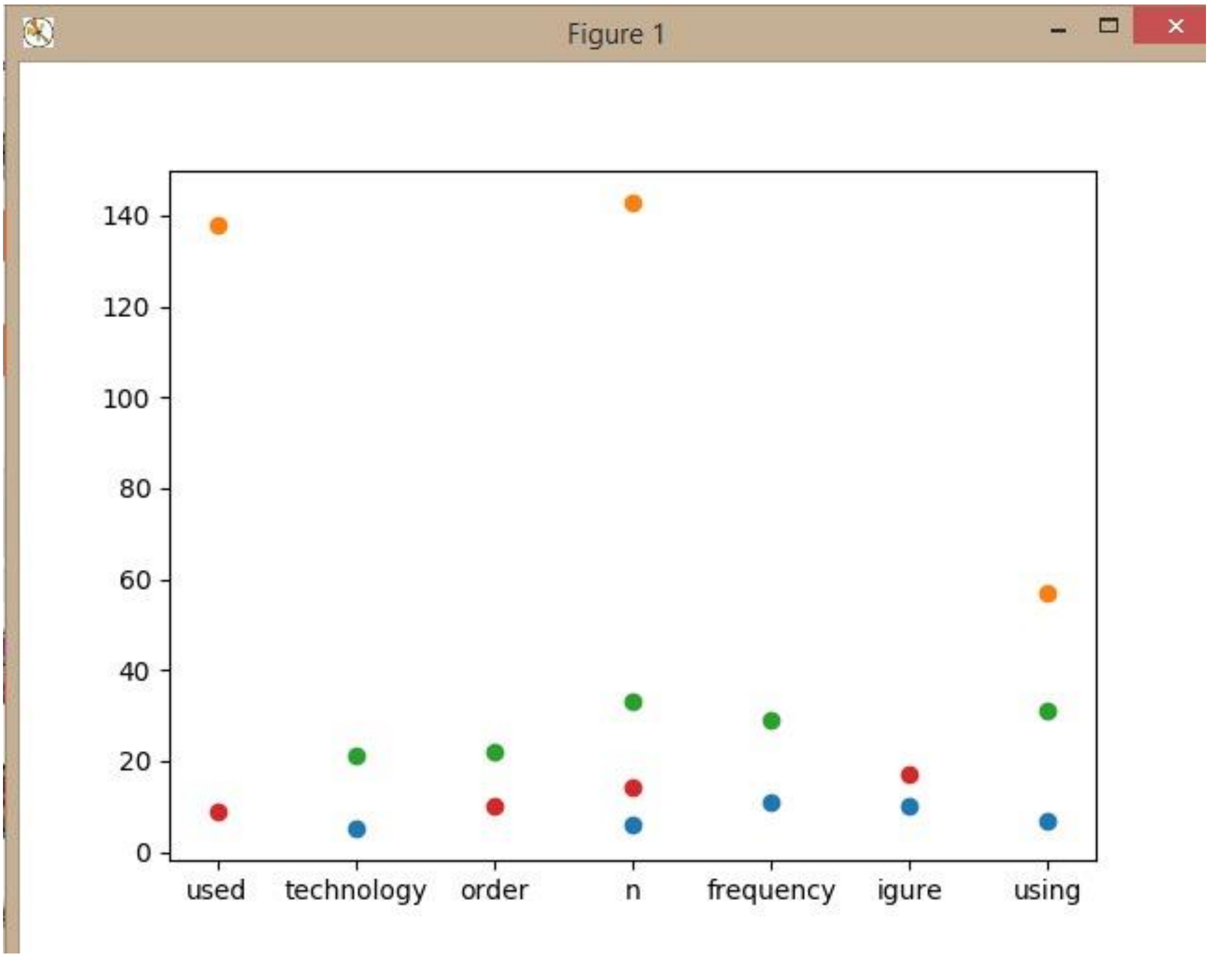
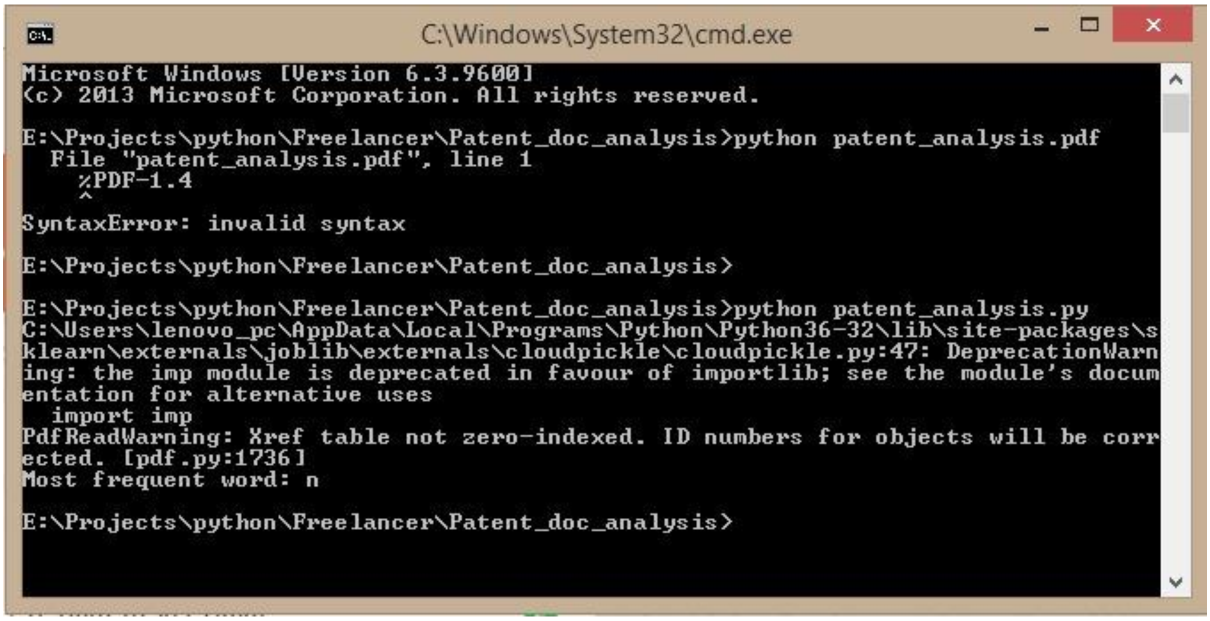


Fig.4.1 Comparison graph of common keywords.



```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

E:\Projects\python\Freelancer\Patent_doc_analysis>python patent_analysis.pdf
  File "patent_analysis.pdf", line 1
    %PDF-1.4
    ^
SyntaxError: invalid syntax

E:\Projects\python\Freelancer\Patent_doc_analysis>

E:\Projects\python\Freelancer\Patent_doc_analysis>python patent_analysis.py
C:\Users\lenovo_pc\AppData\Local\Programs\Python\Python36-32\lib\site-packages\s
klearn\externals\joblib\externals\cloudpickle\cloudpickle.py:47: DeprecationWarn
ing: the imp module is deprecated in favour of importlib; see the module's docum
entation for alternative uses
  import imp
PdfReadWarning: Xref table not zero-indexed. ID numbers for objects will be corr
ected. [pdf.py:1736]
Most frequent word: n

E:\Projects\python\Freelancer\Patent_doc_analysis>
```

Fig.4.2 Showing most frequent word in console.

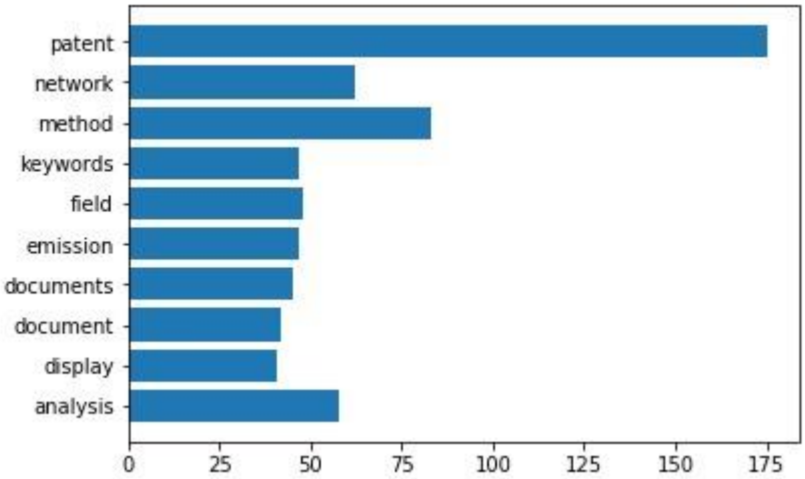


Fig.4.3 Graph showing frequency of keywords of single pdf.

Chapter- 5 CONCLUSION

5.1 Conclusion

This examination builds a novel patent investigation technique, called the patent network analysis, to make an exact visual system. In light of man-made consciousness procedures, this examination proposes a point by point method for creating a wise patent system. To begin with, this examination used the idea of philosophy to seek furthermore, sort important patent records for gathering a total dataset of patent reports. Second, through utilization of the upgraded term recurrence – converse record recurrence (ETF-IDF) method, dependable patent watchwords reasonable for further process examination were extricated. Third, affiliation rules were utilized to decide the weighted estimation of every watchword. At long last, arrangements of patent watchwords were utilized to fill in as the information base for creating a refined patent system.

5.2 Limitation

Unavoidable mistakes in the aftereffects of patent content order presumably exist that would prompt the extraction of mistaken catchphrases. To determine this issue, the programmed classification consequences of the patent documents ought to be reconfirmed, that is, a blended arrangement ought to be embraced that mixes man-made brainpower and human insight to advance rightness and adequacy when preparing the bottomless patent archives.

5.3 Future Scope

- This examination applies man-made consciousness systems to adjust ebb and flow practice and proposes a thorough strategy to make the visual system more sophisticated. The proposed technique has extraordinary enhancements regarding patent inquiry, data extraction, perception, and investigation.
- The keen patent network analysis strategy is important to the handy undertakings of specialists or researchers. It empowers architects and researchers to naturally comprehend the outline of an arrangement of licenses and to distinguish the formative patterns of basic advances.
- It empowers architects and researchers to instinctively comprehend the diagram of an arrangement of licenses and to distinguish the formative patterns of basic advancements. In particular, specialists and researchers can reveal noteworthy mechanical data and handle important innovative experiences in the patent network.

REFERENCES

- CONSTRUCTING AN INTELLIGENT PATENT NETWORK ANALYSIS METHOD by Chao-Chan Wu1 and Ching-Bang Yao

- https://www.ibmbigdatahub.com/blog/introduction-text-mining Introduction to text Mining

- Natural Language Processing is Fun! How computers understand Human Language By Adam Geitgey

- Yoon, B., & Park, Y. (2004) A text-mining-based patent network: analytical tool for high-technology trend. Journal of High Technology Management Research, 15, pp 37-50.

- Mining Frequent Itemsets – Apriori Algorithm

- https://en.wikipedia.org/wiki/Viterbi_algorithm

- http://intechopen.com

- http://link.springer.com

- https://www.geeksforgeeks.org