Lexical And Semantic Analysis Of Sacred Texts Using Machine Learning And Natural Language Processing

Nisha Varghese, M Punithavalli

Abstract: Text mining is the process of exploring and analyzing large amounts of text data and extracting high-quality information based on patterns and trends in data. The patterns and trends include the analysis of similarity measures and opinion mining or sentimental analysis expressed in the texts. Text data mining reveals relationships that lie in single or multiple text data. Applying text mining to religious texts can yield various insights on the cultural and religious basis. Reliable automatic knowledge extraction from sacred texts is a challenging task but it will be beneficial to mankind. This research started with the hypothesis that there is intersection between the Bible, Tanakh, and Quran. All these three books have origins in the Middle East. Bible is the holy book of Christians contains a collection of scriptures that were written by many authors, at different time and locations. Tanakh is the sacred text of Jews with 24 books with three parts- Torah, Nevi'im and Ketuvim and Quran is the central religious text of Islam with 114 chapters as Surah. These three sacred texts contain semi-structured information due to its organized structure of scriptures and numbered chapters, so the comparative studies of the theodicy of three religious texts should reveal interesting insights. The objectives of the research are to implement text analytics on sacred texts and reveal the similarity insights of these sacred texts using Natural Language Processing, ontology modeling, and Machine Learning techniques.

Index Terms: Bags of Words, Corpus, Machine Learning, NLP, Ontology Modeling, polarity, sacred texts, similarity measures, Text Analysis, tf-idf, and word2vec.

1. INTRODUCTION

Text analysis or Text mining is the process of derivation of high-end information through established patterns and trends in a piece of text [1]. Textual data is unstructured and highly noisy, but it usually belongs to a specific language following specific syntax and semantics [2]. The research considers three sacred texts, the Bible, the Tanakh, and the Quran as a corpus for analysis. Text analytics is the methodology and process followed to derive quality and actionable information and insights from textual data. The challenging task is text data is highly unstructured, the three sacred texts contain semi-structured information, so reveal interesting and semantic insights from it also challenging. The studies extract the inner and inter interpretations and meaningful insights using Statistical, Natural Language Processing (NLP), Information Retrieval, Semantic web, ontology Modeling, and Machine Learning and Deep Learning techniques. Bible is the holy book of Christians and a multilingual corpus written in Greek, Hebrew and Aramaic languages. Bible contains a collection of scriptures that were written by many authors, at different times and locations and also includes laws, prophecies, stories, prayers, songs, and wise words. The dreams, parables, and revelations are also included in the bible was made all things to all people in the form of veiled and hidden form. These dreams, parables, and revelations increase the challenge in text analytics because it conceals one meaning other than the normal text resembles or what actually exists. The Bible contains two parts, the Old Testament with 39 books and the New Testament with 27 books. The Old Testament has 929 chapters with 23,214 verses and around 622,700 words. The New Testament consists of 260 chapters and is divided into 7,959 verses and around 184,600 words. The Bible contains some interesting factors like Archaeology, History, Literal criticism and sociological criticism and also includes the values and laws of medieval culture. Tanakh or sometimes called the Mikra is the sacred text of Jews with 24 books with three parts, the first part is Torah with 5 books, then Nevi'im with 8 books and Ketuvim with 11 books. Tanakh is the canonical collection of Hebrew Scriptures, except the books of Daniel and Ezra written in Aramaic. The Tanakh is the textual source for the Old Testament of the Bible.

Quran is the single-authored, central religious text of Islam and written in the eastern Arabian dialect of Classical Arabic. It is slightly shorter than the New Testament. Quran has 30 divisions or Juz with 114 chapters as Surah, according to the length of surahs, but not according to when they were revealed and not by subject matter. Surah is subdivided into verses or Ayat. Quran contains about 6,236 verses with 77,477 words. The Quran was orally revealed by God to Muhammad - the final prophet, through the archangel Gabriel. These three corpora are highly unstructured and do not follow or adhere to structured or regular syntax and patterns. Before applying any statistical techniques or Machine Learning algorithms, the corpus needs to convert into a structured text or a vector format that acceptable by those techniques and algorithms [3].

2.1 Literature Review

Text analytics is a challenging problem due to the unstructured format, the research focused on the knowledge extraction of religious texts. Here in the literature review includes some relevant studies for searching and extracting information from various holy texts. Salha Hassan Muhammed et al [4] implemented various similarity measures to compare the Bible and the Quran. The similarity measures used for the study are Cosine similarity, Hellinger degree of Similarity, Bhattacharyya Distance converted to Similarity, Symmetric Kullback-Libler

- *Nisha Varghese, research scholar in the department of Master of Computer Application in Bharathiar University, Tamilnadu, India, E-mail: nisha.varghes@gmail.com*
- *M Punithavalli is working as a Professor in the Department of Computer Applications, Bharathiar University, Coimbatore, Tamilnadu. India. She has 25 years of research and academic experience. She has published more than 80 research paper articles in international journals and authored three books.*

Divergence, Jensen-Shannon Similarity, Euclidean Similarity, Manhattan Similarity, Symmetric Chi-Square, and Clark Similarity. The drawbacks of the statistical similarity measures, they consider the words, frequency of words and length of sentences. They cannot reveal any similarity if the sentence with different words but the same meaning. Out of these similarity measures cosine similarity can provide better accurate result and it measures irrespective of the size of the corpus. For example the Euclidean and Manhattan similarity measures can compute the similarity only if the term vectors and root vectors are in the same length, Euclidean Distance is not sufficient for smaller distances and the Jaccard similarity index is very sensitive to small corpus. Daniel Merrill McDonald [5] used Self Organizing Maps (SOM) for the automatic extraction and categorization of noun and verb phrases from nine Holy Books: the Book of Mormon, the Greater Holy Assembly, the New Testament, the Old Testament, the PopolVuh, the Qur'an, the Rig Veda, the Tao TeChing, and the Torah. SOM is a double-layered neural network algorithm used for dimensionality reduction and clustering. This study extracted the noun categories like God, animals, plants, body parts and so on and verb categories. The similarity represented in the self-organizing maps based on these categories not based on the semantics of the texts. Mayuri Verma [6] extracted lexical information of ten holy books: the Holy Bible, the Bhagwad Gita, the Guru Granth Sahib, the Agama, the Quran, the Dhammapada, the Tao TeChing, the Rig Veda, the Sarbachan and the Torah using text mining and machine learning tools. The paper extracted the lexical tokens in the form of total words, nouns, verbs using NLP techniques analysis and observations added based on these categories, the extracted information are incomplete. A. Ta'a et al [7] presented an ontology modeling that can extract the Quran in machine readable structure and extracted information from the Quran and the development of ontology Al-Quran. The application used for extracting knowledge using the semantic search approach and RDF/OWL, but advanced techniques such as machine learning and deep learning can reveal or extract more accurate results. Mohamed Osman Hegazi [8] presented a model that read Arabic texts and convert into datasets in the form of relational database records and electronic sheet using algorithms but striving to include multidimensional data or relations. Ramona Popa et al [9] show the implementation of ontology modeling to extract the relations and concepts in the New Testament of the Bible and Ramona Popa et al [10] also proposed knowledge extraction ontology modeling system from Old Testament and New Testament of Bible. Both studies extracted information using the Text2onto tool, by the proper noun elimination most of the central concepts such as God, Jesus Christ are removed from the data.

3 TEXT ANALYTICS

Text Analytics is the process of exploring and analyzing large amounts of unstructured text data and extracting the concepts, patterns, topics, keywords and other attributes in the data. Data mining, natural language processing (NLP), and natural language understanding (NLU) are the overlapping areas include techniques to process large corpora and extract useful insights from the text [11]. The techniques in text analytics to clean data and extracting meaningful information from the corpus are Text Pre-processing and Text Normalization, Text classification or Text Categorization, Text Summarization, Text Similarity and Clustering and Semantic and Sentiment Analysis. The first phase of analytics is Text Pre-processing and Text Normalization which converts raw corpus to standard structure data. A corpus is a collection of sentences with clauses, phrases, and words. Text classification is the process of assigning text documents into one or more classes or categories. Text Summarization and information extraction is the extraction of meaningful inferences and insights from a huge corpus. Text Similarity and Clustering, the similarity means the degree of closeness of two corpora and Text clustering is the process of grouping similar documents into clusters. The semantic analysis specifically deals with understanding the meaning of text or language and Sentiment Analysis to extract subjective and opinion related information.

3.1 Text Pre-processing and Text Normalization

Text Pre-processing is the conversion of raw text data to a structured sequence of linguistic components in the form of a standard vector. The most popular text pre-processing techniques are Tokenization, Tagging, Chunking, Stemming, and Lemmatization. Text processing improves the analysis and accuracy of classifiers and also provides the metadata and additional information, which are useful for extracting knowledge from the corpus. Tokenization is the process of tokenizing or splitting a word, sentence or text into a list of tokens. Tokens are the basic text components with specific syntax and semantics. The two tokenization techniques are sentence tokenization and word tokenization. Sentence tokenization or sentence segmentation segments or splits the text corpus to meaningful sentences. Word tokenization is the process of splitting or segmenting sentences into constituent words. word_tokenize is used to extract the tokens from the corpus. Part-of-speech tagging, POS-tagging, or simply tagging is the process of classifying words into parts of speech and labeling the type of words. It is also termed as word classes or lexical categories. Chunks are the words defined by the part-of-speech tags. The process of extraction of chunks from POS-tagging is known as Chunk extraction or partial parsing. The words remaining after the extraction is known as chinks. Fig. 1 shows the normalized text tokens.

Text Tokens	Frequent items	Normalized Text
['In', 'the', 'beginning', 'God', 'created', 'the', 'heaven', 'and', 'the', 'earth', '.', '1', ':', '2', 'And',	[('the', 60), ('and', 46), (':', 26), (',', 23), ('.', 18), ('god', 17), ('1', 16), ('was', 11), ('let', 9), ('light', 9)]	['In', 'the', 'beginning', 'God', 'created', 'the', 'heaven', 'and', 'the', 'earth', 'And', 'the', 'earth',

Fig. 1 *Pre-processing and Normalized Text*

Stemming is the process of eliminating affixes from a word in

order to obtain a word stem. The removal of suffixes from a word is known as suffix stripping. Two popular stemmers in Python NLTK are PorterStemmer and LancasterStemmer. Lemmatization captures canonical forms based on a word's lemma. TABLE 1 shows examples of stemming and Lemmatization.

Text normalization is also known as text cleansing or wrangling which creates a standardized textual data from raw text using Natural Language Processing and Analytics Systems. The text normalization includes Sentence extraction, HTML escape sequences, Expand contractions, Lemmatize text, remove special characters, stop words, unnecessary tokens, stems, and lemmas. The table shows the normalized form of text Genesis 1:1 from tokens to Tagging. For the normalization purpose the basic text analytics executed with Python Natural Language Tool Kit (nltk) and choose Bible-King James Version as the corpus, nltk.corpus.gutenberg.words('bible-kjv.txt'). Chunking is the tokens with parts of speech tagging, Fig. 2 shows the PoS and Chunks Tagging and results contains some of the POS tags from the PoS tag list, included in TABLE 2.

Parts of Speech(PoS)	Chunks Tagging
[('In', 'IN')] [('the', 'DT')] [('beginning', 'VBG')] [('God', 'NNP')] [('created', 'VBN')] [('the', 'DT')] [('heaven', 'NN')] [('and', 'CC')] [('the', 'DT')] [('earth', 'NN')]	(S In/IN the/DT beginning/NN (PERSON God/NNP) created/VBD the/DT heaven/NN and/CC the/DT earth/NN And/CC the/DT earth/NN

Fig. 2 *PoS and Chunk Tagging*

TABLE 1

STEMMING AND LEMMATIZATION

Words	Porter Stemmer	Lancaster Stemmer	WordNet Lemmatizer	
caring	care	car	corpora	corpus
greater	greater	gre	children	child
better	better	Bet	better	good
criteria	criteria	Criter	criteria	criterion
playing	play	Play	Knives	knife

The differences between PorterStemmer and LancasterStemmer of Words and The results of the WordNet Lemmatizer.

3.2 Text classification or Text Categorization

Text classification is often termed as Document classification is the process of assigning tags or categories to text corpus. It is the task to determine the category of each document from the predefined categories [12]. Document classification is bifurcate into Content-based classification and Request-based classification. Content-based classification gives priority to specific subjects or topics in the corpus. Request-based classification classifies according to the user requests and is related to the specific groups. There are several real-world scenarios and applications in text classification like News articles categorization, Spam filtering, Music or movie genre categorization, Sentiment analysis, and Language detection.

Classification can be done by classifiers, text classifiers can be used to organize, structure, and categorize the source text. Fig. 3 shows the basic structure of classification. A classifier can take this text as an input, analyze, and then automatically assign relevant tags. So Text classification is often termed as text categorization or text tagging.

Fig. 3 *Classification*

TABLE 2

POS TAG LIST

1.	CC	Coordinating conjunction
2.	CD	Cardinal number
3.	DT	Determiner
4.	EX	Existential there
5.	FW	Foreign word
6.	IN	Preposition or subordinating conjunction
7.	JJ	Adjective
8.	JJR	Adjective, comparative
9.	JJS	Adjective, superlative
10.	LS	List item marker
11.	MD	Modal
12.	NN	Noun, singular or mass
13.	NNS	Noun, plural
14.	NNP	Proper noun, singular
15.	NNPS	Proper noun, plural
16.	PDT	Pre-determiner
17.	POS	Possessive ending
18.	PRP	Personal pronoun
19.	PRP$	Possessive pronoun
20.	RB	Adverb
21.	RBR	Adverb, comparative
22.	RBS	Adverb, superlative
23.	RP	Particle
24.	SYM	Symbol
25.	TO	To
26.	UH	Interjection
27.	VB	Verb, base form
28.	VBD	Verb, past tense
29.	VBG	Verb, gerund or present participle
30.	VBN	Verb, past participle
31.	VBP	Verb, non-3rd person singular present

There are two different types of Text classification are manual and automatic classification. Automated Text classification is a smart classification, which automate tasks using machine Learning and makes the whole process in fast and efficient manner. With the data overloading automatic text classification helps to extract data efficiently using classifiers like back propagation neural network (BPNN), decision trees, K-nearest neighbor (KNN), naive Bayes and Support Vector Machine (SVM) [12,13]. The aim of machine learning techniques are increasing accuracy, maximize the performance and its indispensable role in creating self-learning algorithms. Merging of traditional NLP techniques with deep learning concepts provides higher accurate results in classification [2]. Supervised machine learning and unsupervised machine learning are the two main Machine Learning techniques used for automated text classification. Supervised text classification algorithms are given a tagged or categorized text (train set) and generate models, these models when further given the new untagged text (test set), can automatically classify them. One of the popular examples of supervised classification is email-spam filtering. Supervised ML Classification algorithms are used to classify, categorize, or label data points based on trained data. Two leading algorithms are used for text classification are Multinomial Naïve Bayes and Support vector machines. The process of transforming and representing text documents as numeric vectors of specific terms that form the vector dimensions using a mathematical and algebraic model is known as Vector Space Model or Term Vector Model [3]. Only after the vector dimension transformation the algorithms or statistical methods are applicable to the text data. Feature Extraction is an important concept in Machine Learning. Features are basically numeric in nature and unique, measurable attributes. Features can be absolute numeric values or categorical features. One-hot encoding is the process of encoding the features into binary data for each category in the list. The process of extracting and selecting features is called feature extraction or feature engineering. Leading feature-extraction techniques are Bag of Words (BoW) model and TF-IDF model [13]. The Bag-of-Words model is represented as the bag or multiset of words, disregarding grammar and word order, but keeping the frequency of words. BoW uses to find the frequency of each word is used as a feature for training a classifier. It is easy to implement and understand and provides more flexibility. The limitations of the model are sparsity, which means in the case of long vectors the time and space complexity will be higher in the machine learning model, the frequent words have more power even if the words have no importance, it ignores the word orders and the sentences with same meaning but different word ordering. This model is poor in semantic changes of the text and out of vocabulary like misspelling, chat text. Bag of words cannot handle unseen words. TF-IDF is a combination of two metrics: term frequency and inverse document frequency. Term Frequency is the count of the frequency of the term vector in the document. Inverse Document Frequency is a scoring of on rare the word is across documents. TF-IDF is the product of two metrics, tfidf =tf×idf, where term frequency (tf) and inverse-document frequency (idf) represent the two metrics. Inverse document frequency is the inverse of the document frequency for each term and is computed by the ratio between the total number of documents in corpus and the document frequency for each term vector and applying logarithmic scaling on the result. $idf(w)$=1+log C/(1+$df(w)$), Where idf(w) represents the idf for the term w, C represents the count of the total number of documents in our corpus, and df(w) represents the frequency of the number of documents in which the term 't' is present. The final TF-IDF metric uses a normalized version of the tfidf matrix and is the product of tf and idf, then normalize the tfidf matrix by dividing it with the L2 norm of the matrix, also known as the Euclidean norm, which is the square root of the sum of the square of each term's tfidf weight, $tfidf$=$tfidf$/(||$tfidf$||). The advantages of tf-idf are easy to get the document similarity; it keeps the relevant word score and lowers the frequent word score. The tf-idf measures are poor in capturing document topics, measures only based on terms and words and weak handling synonyms and polysemes. That is the sentences with the same meaning cannot identify or don't show any similarity measures between the two sentences.

Both tfidf and BoW are based on words. N-gram is a solution for Bag of words, for ignoring sequence of words, next-word prediction, and misspelling. Bigram, trigram, and N-grams are the contiguous sequences of tokens. Fig. 4 and Fig. 5 shows the results of the Bigrams, Trigrams, and N-grams using Python nltk.

Bigrams	Trigrams
[('In', 'the'), ('the', 'beginning'), ('beginning', 'God'), ('God', 'created'), ('created', 'the'), ('the', 'heaven'), ('heaven', 'and'), ('and', 'the'), ('the', 'earth'), ('earth', '.'), ('.', '1'),	[('In', 'the', 'beginning'), ('the', 'beginning', 'God'), ('beginning', 'God', 'created'), ('God', 'created', 'the'), ('created', 'the', 'heaven'), ('the', 'heaven', 'and'), ('heaven', 'and', 'the'), ('and', 'the', 'earth'), ('the', 'earth', '.'), ('earth', '.', '1'),

Fig. 4 *Bigrams and Trigrams*

N-grams

```
[('In', 'the', 'beginning', 'God', 'created', 'the'),
 ('the', 'beginning', 'God', 'created', 'the', 'heaven'),
 ('beginning', 'God', 'created', 'the', 'heaven', 'and'),
 ('God', 'created', 'the', 'heaven', 'and', 'the'),
 ('created', 'the', 'heaven', 'and', 'the', 'earth'),
 ('the', 'heaven', 'and', 'the', 'earth', '.'),
 ('heaven', 'and', 'the', 'earth', '.', '1'),
 ('and', 'the', 'earth', '.', '1', ':'),
```

Fig. 5 *N-grams*

Latent Semantic Analysis, Word embeddings, Concept Net and ontology are some solutions to overcome the drawbacks of tf-idf. Latent Semantic Analysis measures the similarity based on topic and LSA used the two-dimensional document Vectors and it is an algebraic-statistical method that extracts hidden semantics and sentences [14]. Embedding is the dense vector with similarity and Word2Vec is the popular method for word embedding. The similarity measures from the neighbor words and the multi-sense embeddings improve the performance of NLP tasks, such as part-of-speech tagging and identification of semantic relation. ConceptNet is a semantic network based on a particular domain [15]. Ontology is designed to categorize and extract the relationships between various concepts. Many texts relate researches pointing that ontology modeling can extract semantic information from text corpus [7, 9, 10, 16-23, 25]. The structural framework explains the concepts with its semantic sense and also shows the connections and relationships between concepts on the domain. The fig. 6 and fig. 7 shows the word2vec word embedding using Principal Component Analysis (PCA). PCA reduces the dimensionality of the dataset. The corpus is taken as John 1:1-3 from the Bible and Quran chapter 1: 1 -4.

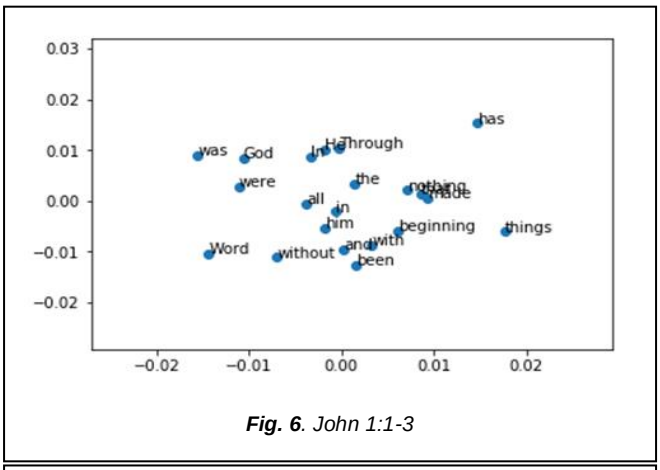

Fig. 6*. John 1:1-3*

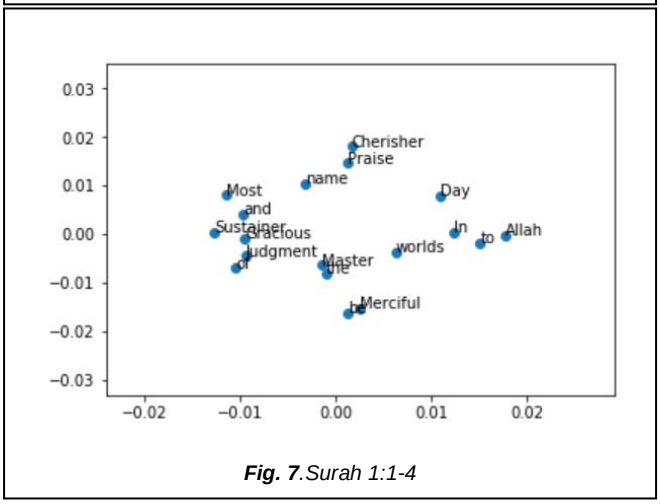

Fig. 7*.Surah 1:1-4*

3.3 Text Summarization and Information Extraction

It deals with the extraction of core concepts from a huge corpus. The demand for text summarization and information extraction increases when the entry of the concept 'information overload' and then it becoming extremely difficult to consume, process and manage. Text summarization systems categories text and create a summary in extractive or abstractive way [14]. There are different types of summarization methods such as Single document summarization, Multi-document summarization, Query focused summarization, Update summarization, Indicative summary, Informative summary, and Headline summary. Key phrase extraction, topic modeling, and automated document summarization are the most popular techniques for text summarization and information extraction. Key phrase extraction or terminology extraction is the process of extracting key important and relevant terms from raw text such that the core topics or themes of the corpus are captured in these key phrases. Key phrase extraction applicable in many areas such as Semantic web, Query-based search engines, and crawlers, Recommendation systems, Tagging systems, Document similarity, and Translation. Automated Document Summarization using automated techniques for extraction, two popular methods are Extraction-based techniques and Abstraction-based techniques.

3.4 Text Similarity and Clustering

The degree of closeness between two entities is measured by similarity measures, which can be any text format like documents, sentences, or even terms. Similarity measures are useful to identify similar entities and distinguishing clearly different entities from each other. Various scoring or ranking algorithms have also been invented based on these distance measures. Inherent properties or features of the entities and Measure formula and properties are the two main factors of the degree of Similarity. Lexical similarity and Semantic similarity are the two aspects of similarity measures, Term similarity and Document similarity are the areas of text similarity. Lexical similarity focused on the contents of the document syntax, structure and content and measuring their similarity based on these parameters. Semantic similarity focused on the semantics, meaning, and context of the documents and finds out the closeness these parameters to each other. Term similarity and Document similarity measures the similarity between tokens and documents respectively. The Euclidean distance, Hamming distance, Levenshtein distance, Manhattan distance, and Cosine distance and similarity are the popular metric of similarity [4].

The Euclidean distance is also known as the Euclidean norm, L2 norm, or L2 distance and is defined as the shortest straight-line distance between two points. Mathematically this can be calculated by

$$\mathrm{ed}(x,y) = \left||x-y|\right|_2 = \sqrt{\sum_{i-1}^{n}(x_i - y_i)^2}$$

The major drawback of Euclidean Distance is not sufficient for smaller distances. Hamming distance between two-term vectors of equal length is the number of positions at which the corresponding symbols are different and it computes the minimum number of substitutions required to change one string into the other.

Levenshtein distance or edit distance computes the difference between two sequences or term vectors, the advantage of Levenshtein distance over Hamming distance and Euclidean Distance the root vector and term vectors need not be same length. Manhattan distance between two points computed along axes at right angles and it is also termed as rectilinear distance, city block distance, taxi cab metric, or Minkowski's L1 distance.

Cosine similarity measures the cosine of the angle between them when they are represented as non-zero positive vectors in inner product space and it ignores the magnitude. If the term vectors having similar orientation will have scores closer to 1or cos0 indicating the vectors are very close to each other in the same direction, if similarity score close to 0 or cos90 indicate unrelated terms with a near orthogonal angle between them and similarity score close to –1 or cos180 indicate terms that are completely oppositely oriented to each other. That is the cosine similarity will be higher for the smaller angle. The advantage over other measures Cosine similarity measures irrespective of the size of the corpus.

The Jaccard similarity index is also termed as Jaccard similarity coefficient which compares members for two sets and finds the common and distinct members. The range from 0% to 100% the higher percentage shows more similarity, but it is extremely sensitive to small datasets and may result in error values. The formula to compute Jaccard similarity coefficient is $Jac(X,Y) = |X \cap Y| / |X \cup Y|$.

TABLE 3 shows the similarity measures of the two statements" John 1:1, "In the beginning, was the Word, and the Word was with God, and the Word was God" and John 1:2 says "He was in the beginning with God" using various similarity measures.

3.5 Semantic and Sentiment Analysis

Text semantics is an understanding of the meaning of text. The main areas under semantic analysis are Exploring WordNet and synsets, Named entity recognition, Analyzing lexical and semantic relations, Word sense disambiguation, and analyzing semantic representations. WordNet is a huge lexical database for the English Language and it contains a set of nouns, adjectives, verbs, and adverbs, and related lexical terms. These sets or the group of data elements are known as cognitive synonym sets or synsets synonym ring and the synsets are considered semantically equivalent for knowledge extraction and information retrieval. Each group of data entries is interconnected by means of conceptual-semantic and lexical relations. In the context of Wordnet, synset is a set of words with same and interchangeable meaning. Synsets are interlinked together with semantic context. It contains simple words but may collocations. Collocation is a combination of two or more words that closely affiliated together naturally. Named entity recognition or entity chunking/extraction, is a popular technique used in information extraction to identify and segment named entities and classify or categorize them under various predefined classes. Words have specific meaning with respect to the position, subject, context and occurrence of other words in the sentence; this is known as a sense of the word or word sense. Such type of word disambiguation is mainly categorized into Homonyms, Homographs, Homophones and many more. Homonyms are the words with same spelling or pronunciation but have different meanings. Homographs are similar to homonyms but pronunciation may different. Homophones are the same pronunciation but spelling and pronunciation are different. There are different varieties of polysemy some important forms are unidirectional, bilingual, double pivotal and contronymic. The following are the examples for homonyms taken from the Bible for the word 'bear', in each sentence the meaning of bear is different. Genesis chapter 4 verses 13 say "And Cain said unto the LORD: 'My punishment is greater than I can bear."

TABLE 3

SIMILARITY MEASURES

NAME OF MEASURE	SIMILARITY
Cosine similarity	0.3380617018914066
Eucledean distance	[[0.]] [[5.29150262]]
Jaccard similarity	1.2
TF-IDF	[0.0, 0.70710677]

Various similarity measures and variations in the similarity of text.

Genesis chapter 16 verses 11 say that "And the angel of the LORD said unto her: 'Behold, thou art with child, and shalt

bear a son;" Exodus chapter18 verses 22 say "so shall they make it easier for thee and bear the burden with thee."
Exodus chapter 20 verses 16 say that "Thou shalt not bear false witness against thy neighbor."
1 Samuel chapter17 verses 34 "And David said unto Saul: 'Thy servant kept his father's sheep; and when there came a lion, or a bear, and took a lamb out of the flock,"
Hosea chapter9 verses 16 says "Ephraim is smitten, their root is dried up, they shall bear no fruit; "

Sentiment analysis is also known as Opinion Mining, it is contextual mining of text which includes methods, techniques, and tools for the detection and extraction of subjective information, such as opinion and attitudes, from source material [19]. Sentiment analysis is categorized into symbolic and sub-symbolic approaches, the former methods include the use of lexicons, ontologies, and semantic networks to encode the polarity associated with words and multiword expressions and now merged with machine learning techniques that perform sentiment analysis and classification based on word frequencies and semantics [24-26]. Sentiment Analysis depends upon the polarity and subjectivity. Polarity measures sentiment through positive polarity or negative polarity, and intensity (compound) towards the input text, that is the emotions or feelings expressed in a sentence. Polarity ranging from -1.0 to 1.0 represents negative and positive statement respectively. Subjectivity is an explanatory text which must be analyzed in context, which ranges from 0 (very objective) to 1.0 (very subjective). If a statement shows high negative polarity with .989 and a very high compound score of -0.998, which means that the statement is very negatively intense.
TextBlob is a Python library helps to find the polarity and subjectivity measures easily. Table 1 shows the polarity and subjectivity measures of the following statements,
Statement 1:"not a very great victory",
Statement 2:"a very great victory",
Statement 3: from Quran, Surah 14:7: "If you are grateful, I will surely increase you [in favor]" and
Statement 4: from Bible, Job 42:15: "and in all the land were no women found so fair as the daughters of job"

4 Results and Discussion

The literature review includes the references of studies already conducted, most of the studies focused on the similarities and overlapping between the corpora, TABLE 3 listed the results of various similarity measures and Fig. 8 and Fig. 9 shows the results of the similarity measures. The similarity measures may vary with respect to the change in length of the corpus and frequency of term vectors. The overlapping of the corpora considers only the term vectors and frequencies and not on the basis of semantics. Fig.4 and Fig.5 show the examples for the Bigrams, Trigrams, and N-grams, the solution for the prediction of next words and Fig. 6 and Fig.7 shows the results of Word2Vec done by gensim Python library. In ontology modeling the proper noun extraction eliminated some core points and central concepts of the corpora.

The following figures are the results of the tf-idf with BoW of the corpus psalms 23 from the bible, comparison of results with a fraction of text from psalms 23:4 using gensim Python library.
Fig. 8 shows tf-idf similarity for a fraction of text from psalms 23:4

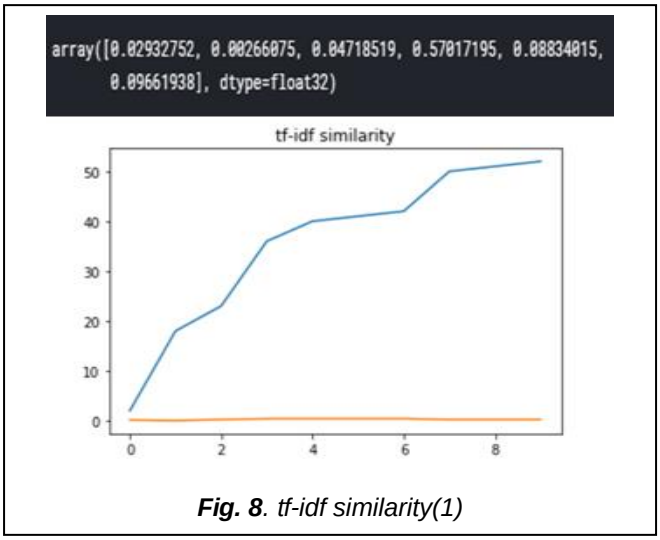

***Fig. 8**. tf-idf similarity(1)*

Fig. 9 shows Tf-idfsimilarity for the text from psalms 23:4

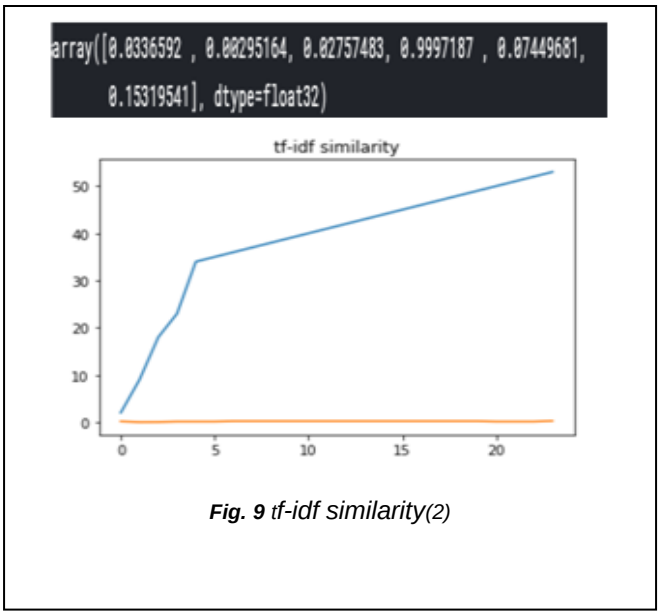

***Fig. 9** tf-idf similarity(2)*

TABLE 4
SENTIMENT ANALYSIS

Statement No.	POLARITY	SUBJECTIVITY
1	0.3076923076923077	0.5769230769230769
2	1.0	0.9750000000000001
3	0.5	0.8888888888888888
4	0.7	0.9

The polarity and the subjectivity of the sentence show the nature of the sentence, which is Positive, Negative or Neutral and its values ranging from -1 to 1.

5 CONCLUSION AND FUTURE WORKS

Automatic text analytics is a common task, but the knowledge

extraction from religious books is challenging because of the extraction of the inner and inter interpretations, unstructured corpora and the polysemic nature in semantics of sentences. Traditional and existing methods are not sufficient to extract meaningful insights from Holy books. Machine Learning and Deep learning techniques like Recurrent Neural Network and Long Short-Term Memory (LSTM) networks can extract better semantic information from religious texts.

REFERENCES

[1] M. Anandarajan et al., Practical Text Analytics, Advances in Analytics and Data Science 2, © Springer Nature Switzerland AG 2019

[2] Muhammad Abulaish, Jahiruddin, and LipikaDey, DeepTextMiningforAutomaticKeyphraseExtractionfromText Documents , J. Intell. Syst.20(2011), 327–351 DOI 10.1515/JISYS.2011.017 © de Gruyter 2011

[3] Tomas Mikolov et al, Efficient Estimation of Word Representations in Vector Space, arXiv:1301.3781v3 [cs.CL] 7 Sep 2013

[4] Salha Hassan Muhammed, "An Automatic Similarity Detection Engine Between Sacred Texts Using Text Mining and Similarity Measures" (2014). Thesis. Rochester Institute of Technology.

[5] Daniel Merrill McDonald , A Text Mining Analysis of Religious Texts, The Journal of Business Inquiry 2014, 13, Issue 1 (Special Issue), 27-47 http:www.uvu.edu/woodbury/jbi/articles/ ISSN 2155-4072

[6] MayuriVerma , Lexical Analysis of Religious Texts using Text Mining and Machine Learning Tools , International Journal of Computer Applications (0975 – 8887) Volume 168 – No.8, June 2017

[7] Ta'a .A, Abed Q. A., & Ahmad M. (2018). Al-Quran ontology based on knowledge themes. Journal of Fundamental and Applied Sciences, 9(5S), 800.

[8] Mohamed Osman Hegazi et al, Fine-Grained Quran Dataset, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 6, No. 12, 2015

[9] Ramona Cristina Popaet al, ONTOLOGY LEARNING APPLIED IN EDUCATION: A CASE OF THE NEW TESTAMENT , The European Proceedings of Social & Behavioural Sciences EpSBS, Edu World 2016 7th International Conference, ISSN: 2357-1330

[10] Ramona Cristina Popa et al, Extracting Knowledge from the Bible: A Comparison between the Old and the New Testament, International Conference on Automation, Computational and Technology Management (ICACTM) Amity University ©2019 IEEE

[11] Daniel C. Elton et al, USING NATURAL LANGUAGE PROCESSING TECHNIQUES TO EXTRACT INFORMATION ON THE PROPERTIES AND FUNCTIONALITIES OF ENERGETIC MATERIALS FROM LARGE TEXT CORPORA, arXiv:1903.00415v1 [cs.CL] 1 Mar 2019

[12] Mingyang Jiang, Text classification based on deep belief network and softmax regression, Neural Computing and Applications, January 2018.

[13] Young-Man Kwon et al, The Performance Comparison of the Classifiers According to Binary Bow, Count Bow and Tf-Idf Feature Vectors for Malware Detection, International Journal of Engineering & Technology, 7 (3.33) (2018) 15-22

[14] MakbuleGulcinOzsoy and FerdaNurAlpaslan , Text summarization using Latent Semantic Analysis, Journal of Information Science 1–13 ,2011

[15] Piero Andrea Bonatti et al, Knowledge Graphs: New Directions for Knowledge Representation on the Semantic Web, Dagstuhl Reports, Vol. 8, Issue 09, pp. 29–111,2018.

[16] Monika Rani, Amit Kumar Dhar, O.P. Vyas, Semi-automatic terminology ontology learning based on topic modeling, Engineering Applications of Artificial Intelligence, Volume 63, August 2017

[17] Kyoung Soon Hwang et al, Autonomous Machine Learning Modeling using a Task Ontology, 2018 Joint 10th International Conference on Soft Computing and Intelligent Systems and 19th International Symposium on Advanced Intelligent Systems.

[18] Victoria Vysotsk, Development of Information System for Textual Content Categorizing Based on Ontology, CEUR-WS.org/ Vol-2362/ 2019

[19] DiegoDeU˜na et al, Machine Learning and Constraint Programming for Relational-To-Ontology Schema Mapping, Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18)

[20] Mohammad M Alqahtani , Developing Bilingual Arabic-English Ontologies of Al-Quran, Proceedings of ASAR'2018 Arabic Script Analysis and Recognition. ASAR'2018 Arabic Script Analysis and Recognition, Alan Turing Institute, The British Library, London UK. IEEE , pp. 96-101.

[21] Dejing Dou et al, Semantic Data Mining: A Survey of Ontology-based Approaches , Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015)

[22] RaziehAdelkhah, The Ontology of Natural Language Processing, 5th International Conference on Web Research (ICWR) 978-1-7281-1431-6/19/$31.00 ©2019 IEEE

[23] A.B.M. ShamsuzzamanSadi et al, Applying Ontological Modeling on Quranic "Nature" Domain , 7th International Conference on Information and Communication Systems (ICICS) 2016

[24] MikaV.Mäntyläa et al , The evolution of sentiment analysis—A review of research topics, venues, and top cited papers, ComputerScienceReview27(2018)16–32

[25] Mauro Dragoni et al, OntoSenticNet: A Commonsense Ontology for Sentiment Analysis, IEEE Intelligent Systems Published by the IEEE Computer Society May/June 2018

[26] DoaaMohey El-Din Mohamed Hussein , A survey on sentiment analysis challenges Journal of King Saud University – Engineering Sciences (2016)

[27] panelMangiKang et al, Opinion mining using ensemble text hidden Markov models for text classification Expert Systems with Applications, Volume 94, 15 March 2018,

[28] SheelaPandey and Sanjay K. Pandey, Applying Natural Language Processing Capabilities in Computerized Textual Analysis to Measure Organizational Culture, Organizational Research Methods 1-33 The Author(s) 2017