Volume 3, Issue 10, October 2013 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Analysis of The Beatitudes (Happiness) from The Holy Bible using Fuzzy Clustering based on Contextual Branching of Artificial Neural Networks

A. Rajkumar*
Assistant Professor
Hindustan University
Chennai – 603 103, India

A. Victor Devadoss
Professor,
Loyola College,
Chennai – 600 034, India

N.D.P.Harish Prasanna
Department of Computer Science
Hindustan University
Chennai – 603103, India

Abstract— The Beatitudes, or the ten commandments that Jesus gave from above the mountainside to his disciples is a remarkable proof of concept on how a man's very state of living to his purpose of existence is linked to enlightenment of the Holy Spirit. In an attempt to study the cognitive formula behind, this paper studies the contextual perfectness of similar instances from The Holy Bible. Contextual Branching is the formation of braches around synonymously relative words. Hence by forming contextual branches around the identified layers, the similarity between two different layers can be linked and clustered using fuzzy logic thereby producing a statistical proof to the cognitive formula that led to the inception of the commandments. Here the decision is given to the contextual perfectness of The Holy Bible to make a qualitative assessment of the verses of the sacred text.

Keywords— The Beatitudes, Fuzzy Clustering, Contextual Branching, Cognitive Process, Perfectness, The Holy Bible.

I. INTRODUCTION

The process of cognition is a faculty for the processing of information, applying knowledge, and changing preferences. Cognition, or cognitive processes, can be natural or artificial, conscious or unconscious. These processes are analysed from different perspectives within different contexts. The concept of cognition is closely related to abstract concepts such as mind, intelligence. It encompasses the mental functions, mental processes (thoughts), and states of intelligent entities (humans, collaborative groups, human organizations, highly autonomous machines, and artificial intelligences). This study has determined on how such a cognitive process came into existence by comparing various other instances given in The Holy Bible. The method that is selected to do the analysis is clustering of branched data using fuzzy logic to determine a statistical value that depicts the contextual perfectness of the biblical verse. Data clustering is the process of dividing data elements into classes or clusters so that items in the same class are as similar as possible, and items in different classes are as dissimilar as possible. Depending on the nature of the data and the purpose for which clustering is being used, different measures of similarity may be used to place items into classes, where the similarity measure controls how the clusters are formed. This can be mapped to a process of thinking that determinably combines previous thoughts and actions into a context that is valid for the present situation. Here the clustering rule divides the use-case into three layers namely input layer, output layer and the hidden layer. The contextual meaning of the input layer and the output layer is determined and similar instances are pulled from the biblical script and synonymously branched and compared to form clusters between the input and the output layer. Then a third hidden layer is introduced to seed the contextual strength of the determined cluster and a value ranging between 0 and 1 is given to each use-case. Hence, if the value is higher it can be inferred as to be biblically perfect and a valid statement considering all the pertaining events of the referred context. Hence fuzzy clustering can be a potential solution to all problems where there is very little prior knowledge and requires a new dimension of a thinking for anymore new outcomes. The organization of this article is as follows: Choosing the attributes in the Methodology section, Basic notion of clustering and fuzzy clustering in following section, Results and Discussion in the third section and finally the conclusion and scope for future study is presented in the fourth and final section.

II. METHODOLOGY

In this study, the aim is to determine the level of perfectness of the biblical verses of Matthew 5:3-12. The Ten Commandments are chosen as the use-cases and are divided into three layers namely input, output and the hidden layer. The Commandments are

1. Blessed are the poor in spirit, for theirs is the kingdom of heaven. [Matthew 5:3]
2. Blessed are those who mourn, for they will be comforted. [Matthew 5:4]
3. Blessed are the meek, for they will inherit the earth. [Matthew 5:5]
4. Blessed are those who hunger and thirst for righteousness, for they will be filled. [Matthew 5:6]
5. Blessed are the merciful, for they will be shown mercy. [Matthew 5:7]
6. Blessed are the pure in heart, for they will see God. [Matthew 5:8]
7. Blessed are the peacemakers, for they will be called children of God. [Matthew 5:9]

8. Blessed are those who are persecuted because of righteousness, for theirs is the kingdom of heaven. [Matthew 5:10]
9. Blessed are you when people insult you, persecute you and falsely say all kinds of evil against you because of me. [Matthew 5:11]
10. Rejoice and be glad, because great is your reward in heaven, for in the same way they persecuted the prophets who were before you. [Matthew 5:12]

Each of these attributes [A_1, A_2.., A_{10}] are selected and categorized into their levels [Input (I_1-I_{10}), Output (O_1-O_{10}) and Hidden (H_1-H_{10})]. The categorized levels are then converted to tags [$I_1T_{1\text{-}n}$, $I_2T_{1\text{-}n}$,.. $I_{10}T_{1\text{-}n}$]. The tags are further used to pull similar instances from The Holy Bible and a link chain [$L(I_n,B_1\text{-}B_x)$], is formed. The determined link chain is now contextually branched and extensively mapped at various levels of synonymy with the base tags. The same process is carried out for the output layer also. Now both the context branch maps are clustered using fuzzy logic and a context strength is given to each cluster. In order to prove the concept a mean value is taken by again clustering the common clusters of Input layer and Output layer with that of the determined context branch of the hidden layer.

III. PRELEMINARIES

A. Lexical Chaining

A lexical chain is a sequence of related words in writing, spanning short (adjacent words or sentences) or long distances (entire text). A chain is independent of the grammatical structure of the text and in effect it is a list of words that captures a portion of the cohesive structure of the text. A lexical chain can provide a context for the resolution of an ambiguous term and enable identification of the concept that the term represents. The structure is Index, Meaning and Chain.
E.g., Formation of Output Link Chain ($L(O_n,B_1\text{-}B_x)$)

Index	Meaning	Link Chain
1	"Kingdom of God"	"Spiritual World" → "unseen" → "Spiritual Domain" → "Heaven"
2	"Thirst of Righteousness"	"Morality" → "Righteous" → "Search for Good" → "Fulfilment"
3	"Children of God"	"Holy" → "Spiritual Hold" → "descendants" → "Prophecy" →"Peace"

Algorithm:

1. For each noun instance
   For each sense of the noun instance
   Compute all scored meta-chains
2. For each noun instance
   For each meta-chain to which the noun belongs
   Keep word instance in the meta-chain to which it contributes most
   Update the scores of each other meta-chain

Lexical Filtered Tagging

Lexical filtering is a feature which allows the user to constrain the lexical selection to only those items that contain a certain property, for example, the realizing an item as a definition. The idea is that the user provides an input like idx constraints: [definition:A_x], which means that the lexical selection must include exactly one tree with the property [definition:A_x] in its interface. This mechanism works as pre-processing step after lexical selection and before polarity automaton construction, in conjunction with the Extra-Polarities mechanism. What we do is
Preprocess the lexically selected trees; any tree which has a desired property (e.g. definition:A_x) in its interface is assigned a positive polarity for that property (+definition:A_x)
Add all the index constraints as negative extra polarities (-definition:A_x)

The tags that are marked as a negative extra polarity are ignored as they are just present to link the words to build the sentence. They do not make any sense when considered for lexical chaining as they are just link words.

Fuzzy c-means Clustering

In fuzzy clustering, every point has a degree of belonging to clusters, as in fuzzy logic, rather than belonging completely to just one cluster. Thus, points on the edge of a cluster, may be in the cluster to a lesser degree than points in the center of cluster. The FCM algorithm attempts to partition a finite collection of n elements into a collection of c fuzzy clusters with respect to the determined contextual definition.

$$X = \{x_1, \ldots, x_n\} \quad (1)$$

Considering the branched data, the algorithm returns a list of c cluster centers $C = \{c_1, \ldots, c_c\}$ and a partition matrix

$$W = w_{ij} \in [0, 1], i = 1, \ldots, n, j = 1, \ldots, c \quad (2)$$

Where each element w_{ij} tells the degree to which element x_i belongs to cluster c_j. Like the k-means algorithm, the FCM aims to minimize an objective function. With fuzzy c-means, the centroid of a cluster is the mean of all points, weighted by their degree of belonging to the cluster.

$$c_k = \frac{\sum_x w_k(x)x}{\sum_x w_k(x)}. \quad (3)$$

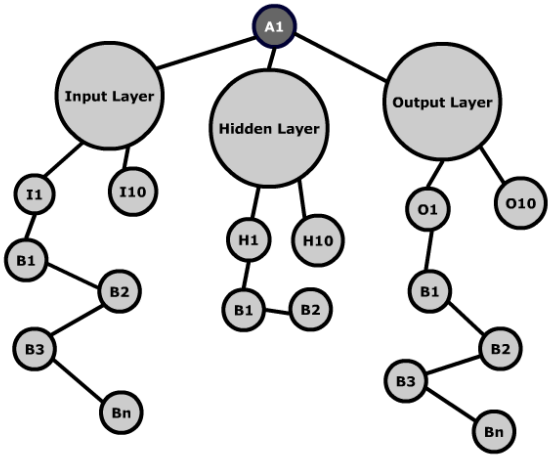
Hence the standard function is given by

$$w_k(x) = \frac{1}{\sum_j \left(\frac{d(\text{center}_k, x)}{d(\text{center}_j, x)}\right)^{2/(m-1)}}. \quad (4)$$

For m equal to 2, this is equivalent to normalizing the coefficient linearly to make their sum 1. When m is close to 1, then cluster center closest to the point is given much more weight age than the others, and the algorithm is similar to k-means.

IV. RESULTS AND DISCUSSIONS

By intensive lexical chaining and Contextual Branching an exhaustive link chain is created for each and every tag of the considered layers. By comparing all the synonymous definition of the link chains with the input layer versus the output layer clusters are formed using the FCM Algorithm and then again clustered with the link chain of the hidden layer to determine the statistical mean value which depicts the perfectness of the biblical verses.



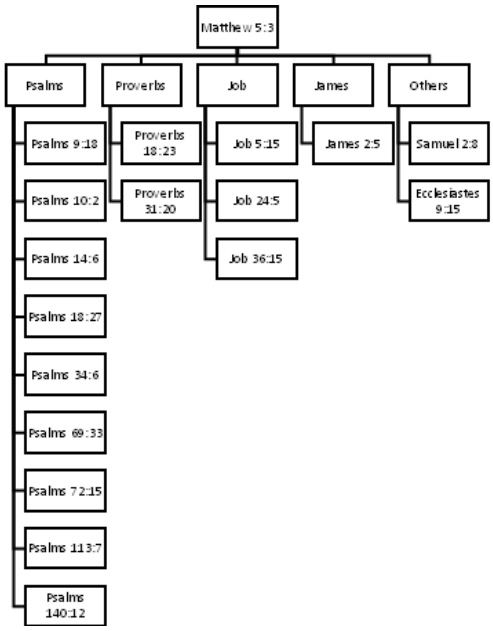
(Fig :A- Formation of lexical chain for layer attributes)

The Commandments are divided into the three basic layers of the neural networks model i.e. Input Layer, Output layer and The Hidden Layer.

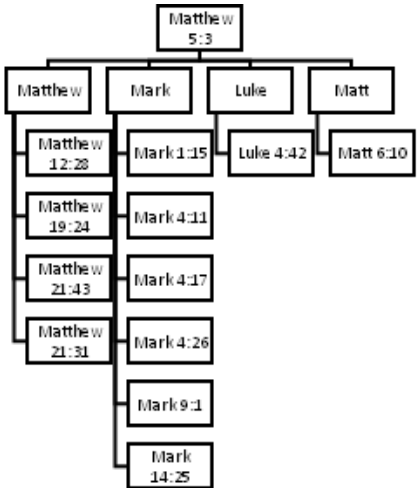
S. No	Input layer	Hidden Layer	Output layer
1	Poor In Sprit	Humble /Empty /Hopeless/To Looking His Word	Kingdom Of God
2	Mourn	Contrite/Repentance/Salivation	Inherit The Earth
3	Meekness	Obedient/Patience	Inherit The Earth

Consider Case 1:
INPUT LAYER: Poor in spirit
OUTPUT LAYER: Kingdom of god
HIDDEN LAYER: Humble, Empty, Hopeless, To Looking His Word
DISAMBIGUATION OF INPUT HIDDEN & OUTPUT LAYER:
Input Layer Tags = I = "Poor", "in", "Spirit."
Hidden Layer Tags = H = "Humble", "Empty", "Hopeless", "To", "Looking", "His", "Word".
Output layer Tags = O = "Kingdom", "of", "God".

FILTERED TAGGING:
Input layer:
Lexically accepted: "Poor", "Spirit".
Ignored: "in" (Reason: Connector)
Hidden Layer:
Lexically Accepted: "Humble", "Empty", "Hopeless", "Looking", "Word".
Ignored: "To", "His" (Reason: Reference Connector)
Output Layer:
Lexically Accepted: "Kingdom", "God".
Ignored: "of" (Reason: Reference Connector)
Lexical chaining:
contextual branching & re-chaining:
Context Branch is done by utilizing the existing Bible reference system e.g. Matthew-5:10.
Input Layer Context Branch:



OUTPUT LAYER CONTEXT BRANCH:

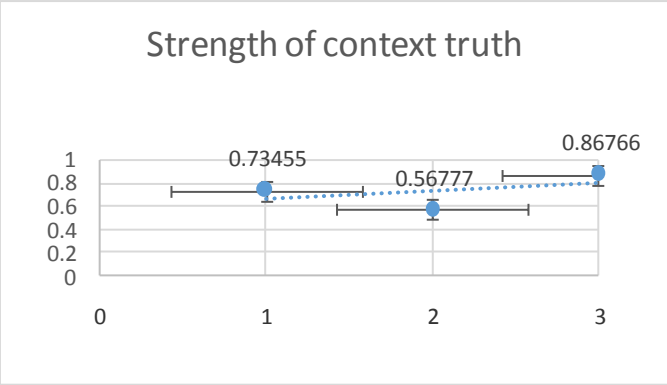


ALGORITHM TO FORM THE LINK CHAIN AND DETERMINE THE MEMBERSHIP VALUES OF THE DETERMINED CLUSTERS:

Step 1: Start
Step 2: Divide the attributes(A_1,A_2,..A_{10}) into Input layer ($I_{1\text{-}10}$), Output Layer($O_{1\text{-}10}$) and Hidden Layer($H_{1\text{-}10}$).
Step 3: Convert the categorized layers into separate tags ($I_1T_{1\text{-}n}$,$I_2T_{1\text{-}n}$,..$I_{10}T_{1\text{-}n}$)
Step 4: The tags are then used to pull out similar instances from the scripture (B_1,..B_x).
Step 5: The instances are then linked with the determined tags and a link chains are formed ($L(I_n,B_1\text{-}B_x)$), ($L(O_n,B_1\text{-}B_y)$), ($L(H_n,B_1\text{-}B_z)$).
Step 6: The link chains for the induced attribute set is further classified and minimised using lexical filtering.
Step 7: The resultant graphs are mapped and clustered across a chart of the linked clusters against value ranging from 0-1 is noted depending on its contextual linkage (C-Center Value).
Step 8: The cluster C-centers are re-estimated to form class members of the closest or the strongest cluster values.
Step 9: Repeat step 8 untill the clusters can no longer be classified and reduced.
Step 10: Stop
The context branch linkage between the link chains ($L(I_1,B_1\text{-}B_x)$), ($L(O_2,B_1\text{-}B_y)$), ($L(H_3,B_1\text{-}B_z)$) are thus minimized and re-fixed n number of times to determine the average mean value of the strenth of truth of the context . The fixing is stoped when the error rate hits below the threshold value which is 0. Three sets of randomly chosen iteration value sets are taken to determine the average value of the context strength.

Iteration	Input Layer	Output Layer	Hidden Layer
12	0.72467	0.86590	0.56494
47	0.73434	0.86721	0.56844
89	0.74464	0.86987	0.56993
Average	0.73455	0.86766	0.56777

In this case the Input layer scores a value of 0.73455, Output Layer a value of 0.86766 and the Hidden layer scores a value of 0.56777. Hence the use case can be given an average score of 0.72332.



V. CONCLUSION

Thus the context strength of the considered usecase – 1 [Matthew 5:3] with reference to The Holy Bible is said to be 0.72332. Similarly the context strength can be analysed for all other Beatitudes by following the context branching and clustering method provided the attributes should be divided into Input layer, Output layer and Hidden Layers. Hence, Fuzzy Clustering using artificial neural network model has been successfully used to analyze and qualify The Ten Commandments or The Beatitudes- Matthew 5:3-12, The Holy Bible.

VI. SUGGESTIONS

The truth of The Holy Bible is obvious to anyone willing to fairly investigate it. The Holy Bible is uniquely self-consistent and extraordinarily authentic. It has been confirmed countless times by archaeology and other sciences. It possesses divine insight into the nature of the universe and has made correct predictions about distant future events with perfect accuracy. When Christians read The Holy Bible, they cannot help but recognize the voice of their Creator. The Bible claims to be the Word of God, and it demonstrates this claim by making knowledge possible. It is the standard of standards. The proof of the Bible is that unless its truth is presupposed, we couldn't prove anything at all.

ACKNOWLEDGEMENT

I Wish to thank my professor praveen prakash for his valuable support and encouragement

REFERENCES

[1] *The Holy Bible*, The bible society of India.
[2] Efficiently computed lexical chains as an intermediate representation for automatic text summarization HG Silber, KF McCoy - Computational Linguistics, 2002 - MIT Press.
[3] Learning in artificial neural networks: A statistical perspective. H White - Neural computation, 1989 - MIT
[4] Mapping context and content: the BrainMap model. PT Fox, JL Lancaster - Nature Reviews Neuroscience, 2002 nature.com

