BOHR International Journal of Smart Computing and Information Technology
2020, Vol. 1, No. 1, pp. 1–12
DOI: 10.54646/bijscit.2020.01
www.bohrpub.com

METHODS

Detecting paraphrases in the Marathi language

Shruti Srivastava* **and Sharvari Govilkar***

Department of Computer Engineering, PCE, University of Mumbai, India

***Correspondence:**
Shruti Srivastava,
shrutics2015@gmail.com
Sharvari Govilkar,
sgovilkar@mes.ac.in

Received: 10 January 2020; **Accepted:** 25 January 2020; **Published:** 05 February 2020

Paraphrasing refers to writing that either differs in its textual content or is dissimilar in rearrangement of words but conveys the same meaning. Identifying a paraphrase is exceptionally important in various real life applications such as Information Retrieval, Plagiarism Detection, Text Summarization and Question Answering. A large amount of work in Paraphrase Detection has been done in English and many Indian Languages. However, there is no existing system to identify paraphrases in Marathi. This is the first such endeavor in the Marathi Language. A paraphrase has differently structured sentences, and since Marathi is a semantically strong language, this system is designed for checking both statistical and semantic similarities of Marathi sentences. Statistical similarity measure does not need any prior knowledge as it is only based on the factual data of sentences. The factual data is calculated on the basis of the degree of closeness between the word-set, word-order, word-vector and word-distance. Universal Networking Language (UNL) speaks about the semantic significance in sentence without any syntactic points of interest. Hence, the semantic similarity calculated on the basis of generated UNL graphs for two Marathi sentences renders semantic equality of two Marathi sentences. The total paraphrase score was calculated after joining statistical and semantic similarity scores, which gives a judgment on whether there is paraphrase or non-paraphrase about the Marathi sentences in question.

Keywords: paraphrase, Marathi language statistical, semantic, Sumo metric, Universal Networking Language (UNL)

1. Introduction

Paraphrase the translation of a sentence or a paragraph into same language. Paraphrasing occurs when texts are lexically or syntactically modified to appear different, but retaining the same meaning. Paraphrase can be generated, extracted and identified. Paraphrase extraction involves collection of different words or phrases that express the same or almost the same meaning. Vocabulary plays an important role in paraphrase extraction. Paraphrase extraction helps in paraphrase generation. Paraphrase generation involves not only dictionary exercise but also changing the information sequence and grammatical structure.

Paraphrase identification is a method of detecting the variety of expressions that convey the same meaning. It poses a major challenge for numerous NLP applications. In automatic summarization, identification of paraphrases is necessary to find repetitive information in the document. In information extraction, paraphrase identification provides the most significant information whereas in information retrieval query, paraphrases are generated to retrieve better quality of relevant data. In question and answering systems, in the absence of questions from database, the answers returned for the question paraphrase are always helpful. The base of paraphrasing is semantic equivalence, which gives alternative translation in the same language. For paraphrase detection it is necessary to study the possibilities of paraphrasing at each level. Mainly there are 3 types of surface paraphrases.

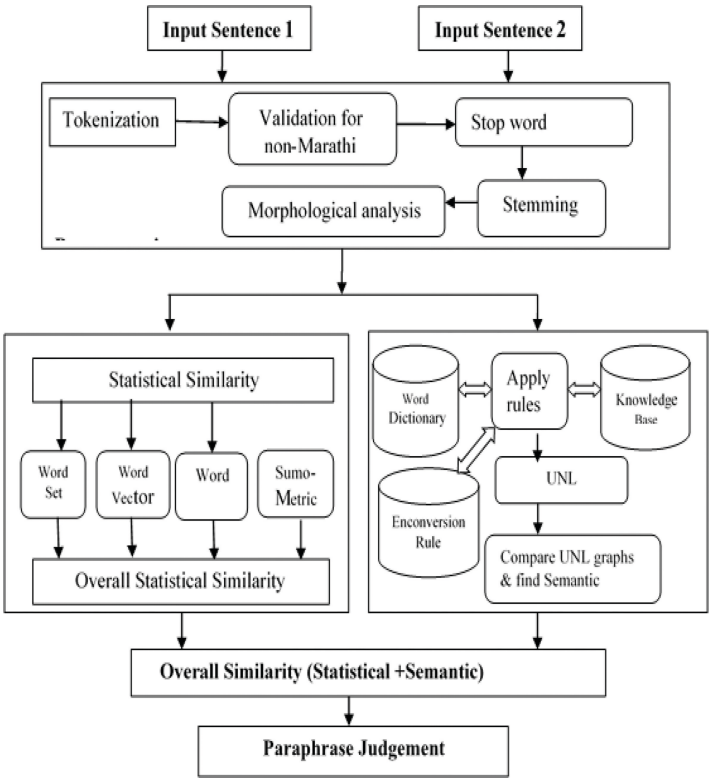

FIGURE 1 | Proposed system architecture.

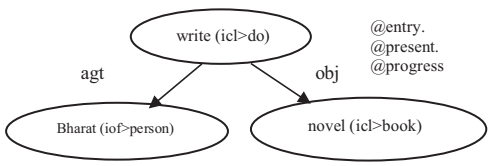

FIGURE 2 | UNL graph example.

1.1. Lexical level

Lexical paraphrases occur when synonyms appear in almost identical sentences. At lexical level, in addition to synonyms lexical paraphrasing is characterized by hypernymy. In hypernymy one word has many alternatives but only one of the words is more general or specific than the other one.

Example –synonyms (solve and resolve), (पुत्री – सुता, कन्या)

Hypernymy (reply, say), (landlady, hostess) {घर} has {सदन, शाला, आलय, धाम} hypernym

S1: मोदी सरकारने दोन पर्याय समोर ठेवले आहेत. S1: The Modi government has put forward two options.

S2: मोदी सरकारने दोन विकल्प मांडले आहेत. S2: The Modi government has given two options.

1.2. Phrase level

Phrasal paraphrase involves different phrases sharing the same semantic content. Phrasal paraphrases include syntactic phrases as well as pattern formation with connected factors.

Example S1: तुलसीदासांनी रामायण लिहिले. (Tulsidas wrote the Ramayana.)

S2: रामायण तुलसीदासांनी लिहिले. (The Ramayana was written by Tulsidas.)

S3: रामायणचे लेखक तुलसीदास आहेत. (The Ramayana's author is Tulsidas.)

S4:तुलसीदासांनी रामायणाची रचना केली. (Tulsidas composed the Ramayana.)

1.3. Sentence level

In this type of paraphrase, one sentence is totally replaced by another sentence, retaining the same meaning.

Example –

S1: महाराष्ट्रातील प्रत्येक प्रांताची मराठी भाषा निराळी आहे. (Each region of Maharashtra has a different Marathi language.)

S2: मराठी भाषा सर्व महाराष्ट्राची जरी एक असली तरी तीच दर बारा कोसावर बदलते. (The Marathi language is the only one in the whole of Maharashtra, the same rate varies from 12 to 12 degrees.)

2. Literature survey

This section presents various techniques for paraphrase detection in Indian languages. The techniques have been applied on various languages to detect whether two sentences were paraphrases of each other or not.

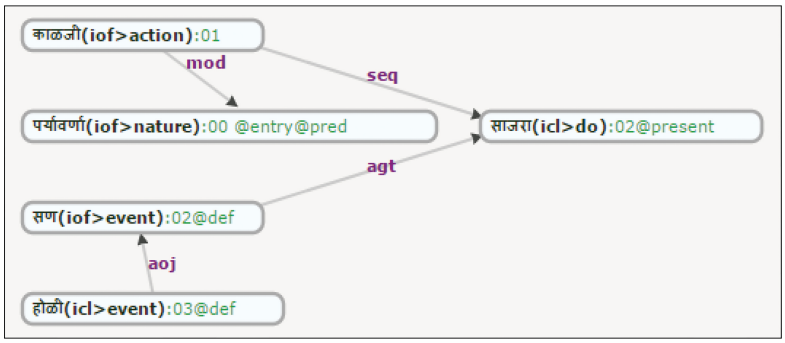

FIGURE 3 | UNL_1 graph for sentence1.

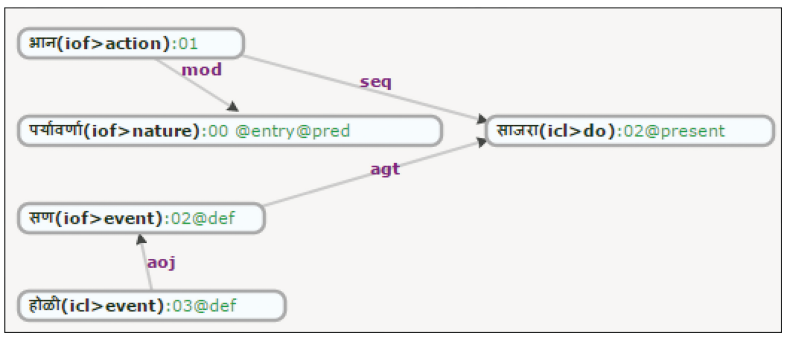

FIGURE 4 | UNL_2 graph for sentence2.

3. Proposed system

The suggested system is designed to take a set of two Marathi sentences as the input. The pre-processing can be done on sentences for collecting actual root words. Then the pre-processed data is given as input to the different statistical and semantic similarities to find paraphrases. The final similarity score is generated by combining the maximum statistical similarity score and the semantic similarity score. The final paraphrase judgment is decided based on a set threshold value which gives the decision of Paraphrase (P) or Non-Paraphrase (NP).

In this approach the system takes one original Marathi sentence and one suspicious paraphrase sentence as the input. The sentences are then pre-processed through various phases including tokenization and input validation by stop word removing, stemming and morphological analysis. After which the pre-processed data is fetched to the matrix calculation for paraphrase detection.

3.1. Pre-processing

The first step is to present two Marathi sentences in text format.

3.1.1. Sentence 1

होळीचा सण साजरा करतांना पर्यावर्णाची काळजी घ्यावी.(The environment should be taken care of while celebrating the Holi festival.)

3.1.2. Sentence 2

पर्यावर्णाचे भान ठेवून होळीचा सण साजरा करावा. (Celebrate the Holi festival keeping the environment in mind.)

The pre-processing of the documents involves the following methods:

3.1.3. Tokenization

Tokenization is a method of identification and separation of tokens from two Marathi input sentences. Lexicon is an individual word of a sentence. Word boundaries in the Marathi language are fixed as it is a segmented language. Spaces are used between words to separate one word from another. The process of separation of tokens involves removal of delimiters like white spaces and punctuation marks.

3.1.4. Validation of Devanagari script

The Marathi language is written in Devanagari script. Input sentence validation is an important phase due to language-dependent information and type of query provided to the system. The characters in Devanagari script are recognized with Unicode values from UTF-8. Comparing each and every character with the UTF-8 list, invalid Devanagari characters were removed from the sentences. The valid characters in Devanagari script were retained for the next phase.

3.1.5. Stop word elimination

Stop words are the utmost irrelevant words which delay the sentence processing. These stop words are the most

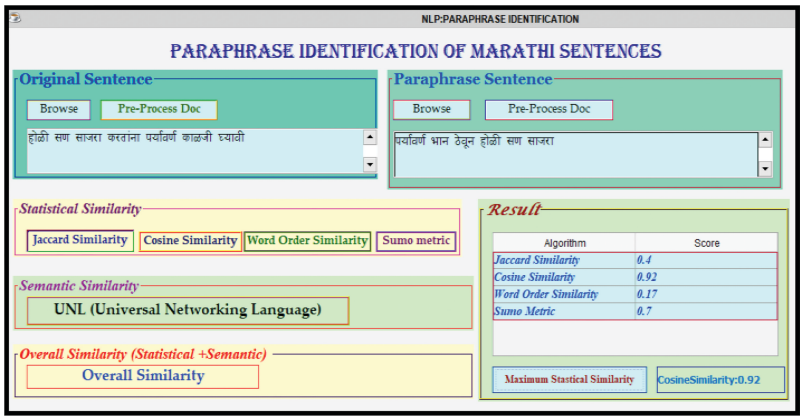

FIGURE 5 | GUI of the system.

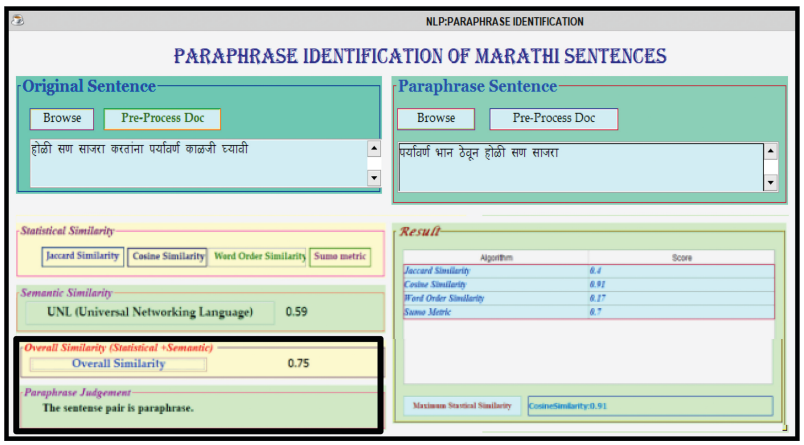

FIGURE 6 | Overall similarity and paraphrase judgment.

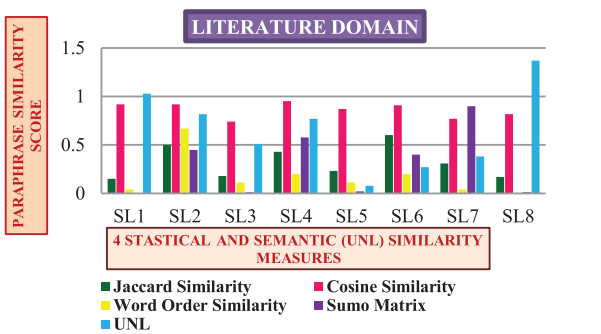

FIGURE 7 | Comparison of Statistical and semantic similarity for literature domain.

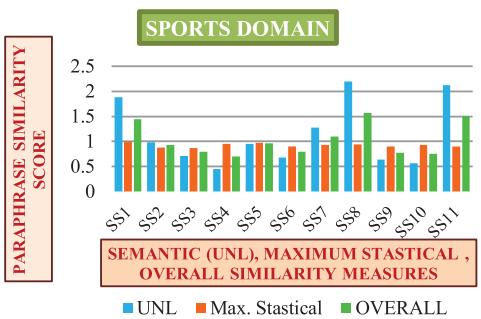

FIGURE 8 | Overall similarity measures for sports domain.

frequently occurring words in any document. The list of stop words includes articles, prepositions, and other function words. Hence to enhance the processing speed of the system, stop words need to be identified and removed properly.

3.1.6. Stemming

Stemming is a very important step in the system. In this phase, a suffix list is used to remove suffixes from words for creating the exact stem word as the stem may not be the linguistic root of the word.

3.1.7. Morphological analysis

The morphological analyzer identifies the inner structure of a word and throws up the root words from the given set of words. After stemming, the words are evaluated for any inflection. The perfect root words can be produced by generating and matching rules. Addition or replacement of characters to the inflected stem word results in the precise core word.

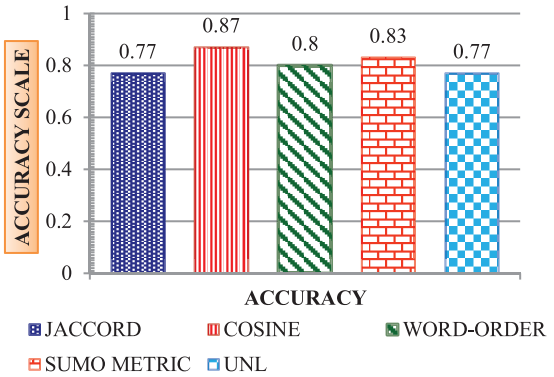

FIGURE 9 | Accuracy comparisons of all similarity techniques.

3.2. Statistical similarity

The tokens created in the tokenization process are the root words but it is necessary to distinguish the tokens. Statistical similarity is calculated by considering the tokens of both sentences and comparing them on the basis of word-set, word-order, word-vector and sumo-metric (word distance). The system uses 4 statistical methods for statistical comparisons, which are explained as follows,

3.2.1. Jaccard similarity

It is based on the similarities and differences between two word sets from the pair of sentences (1). Jaccard similarity coefficient is calculated by taking the ratio of the total number of matched unigrams to the total number of unmatched words in both the sentences. The Jaccard similarity is defined by the following equation:

$$\text{Jaccard Similarity (S1, S2)} = \frac{S1 \cap S2}{S1 \cup S2}$$

Example: Jaccard Similarity

	Sentences	Score
S1	होळी सण साजरा करतांना पर्यावर्ण काळजी घ्यावी	S1∩S2 = 4 {होळी, सण, साजरा, पर्यावर्ण}
S2	पर्यावर्ण भान ठेवून होळी सण साजरा	S1US2 = 9 **Jaccard Similarity (S1, S2) = 4/9 = 0.4**

3.2.2. Cosine similarity

Cosine similarity (1) is the most common statistical feature to measure the similarity between word vectors of two sentences. For each sentence, a word vector of root words of those sentences is formed which interprets the frequency of words in the sentences. Cosine similarity is the ratio of the dot product of those two word vectors to the product of their lengths.

$$\text{Cosine Similarity (S1, S2)} = \frac{S1.S2}{|S1|\,|S2|}$$

Hence, Cosine similarity (S1, S2) = 0.91

3.2.3. Word order similarity

The vectors of the two sentences were considered for calculating the Word order similarity between them. The vectors of two sentences were constructed using the following equations:

$$L(S_a) = \{ (w_{a1},w_{a2}),(w_{a1},w_{a3}),.....,(w_{a(i-1)},w_{ai})\}$$

$$L(S_b) = \{ (w_{b1},w_{b2}),(w_{b1},w_{b3}),.....,(w_{b(i-1)},w_{bi})\}$$

where vector $(w_{a1},w_{a2},.....,w_{ai})$ and vector $(w_{b1},w_{b2},.....,w_{bi})$ were constructed from sentence tokens S_a and S_b, respectively. w_x is before w_y in $(w_x, w_y) \in L\ (Sa)\ U\ L(S_b)$. The similarity calculation between S_a and S_b can be done (1) on the basis of the following equation:

$$\text{WordOrder(Sa, Sb)} = \frac{|L\ (Sa) \cap L(Sb)|}{|L\ (Sa)\ UL(Sb)|}$$

L (Sa) ∩ L(Sb) = { होळी सण,होळी साजरा, सण साजरा, *होळी पर्यावर्ण*, होळी सण,होळी साजरा, सण साजरा, *होळी पर्यावर्ण*, } = 6

$$\text{WordOrder}\,(Sa, Sb) = \frac{|L\ (Sa) \cap L\ (Sb)|}{|L\ (Sa)\ UL(Sb)|} = \frac{6}{30} = 0.17$$

3.2.4. Sumo metric

The Sumo-Metric (2) is on the word distance. This metric finds the special lexical links between the root words of a pair of Marathi sentences. This metric not only identifies paraphrases but also identifies the exact and quasi-exact matches. It can be considered as 1-gram exclusive overlap.

S1 and S2 are a pair of sentences given as an input to the system. Let x and y be the numbers of words in those two sentences. The number of words in the intersection of the two sentences excluding repetitions is represented by λ .

To calculate the Sumo-Metric S (.,.), first evaluate the function S(x, y, λ)

$$S(x, y, \lambda) = \alpha \log 2\left(\frac{x}{\lambda}\right) + \beta \log 2\left(\frac{y}{\lambda}\right)$$

where α, β ∈ [0, 1] and α + β = 1. Now the Sumo-Metric S (.,.) can be calculated by the following equation:

$$S(S1, S2) = \begin{cases} S(x, y, \lambda) & \text{if } S(x, y, \lambda) < 1.0 \\ e^{-k*S(x,y,\lambda)} & \text{otherwise} \end{cases}$$

5α and β are two core components involved in the calculation. The exact and quasi exact match pairs are gradually penalized by log2 (.) function as for equal pairs the sumo-metric result is exactly zero.

Sentence 1: होळी सण साजरा करतांना पर्यावर्ण काळजी घ्यावी

Sentence 2: पर्यावर्ण भान ठेवून होळी सण साजरा

λ = number of words in the intersection of the two sentences = {होळी, सण, साजरा, पर्यावर्ण} = 4

Hence, if α and β = 0.5

Then, $S(x, y, \lambda) = 0.5 * \log2\left(\frac{7}{4}\right) + 0.5 * \log2\left(\frac{6}{4}\right) = \mathbf{0.7}$

3.3. Semantic similarity

Universal Networking language (UNL) generates a graph of semantic representation of sentences. In this method, UNL relation signifies semantic information in a sentence by drawing a hyper-graph. In this hyper-graph nodes represent concepts and arcs signify relations. The hyper-graph contains a set of directed binary relations between two concepts of a sentence.

UNL relation involves mainly 3 important terms which are as follows:

1. Universal Words (UWs): These are the root words that signify word meanings.

2. Relation Labels: These labels **t**ag the various interactions between Universal Words.

3. Attribute Labels: Give extra information about the Universal Words present in a sentence.

Two UNL graphs are compared for similarity, indirectly comparing the semantic content of two sentences. This comparison is useful for finding the semantic similarity score.

3.3.1. UNL expression

The directed graph generated in this method is known as a "UNL Expression" or "UNL Graph," which gives a long list of binary relations.

The general format of UNL Expression is as follows:

<relation>**:[**<**scope-ID**>**](**<**from-node**>**,**<**to-node**>**)**

A UW is described in each of <from-node> and <to-node>.

An example of the UNL expression of the sentence "Bharat is writing a novel."

agt[write(icl>do)@entry.@present.@progress, Bharat (iof>person)]

obj[write(icl>do)@entry.@present.@progress, novel(icl>book)]

3.3.2. UNL En-conversion process

UNL en-conversion is a process of translating natural language text into UNL for syntactic and semantic evaluation.

The En-converter scans the sentence from left to right, and then the word dictionary is used for matching morphemes, which will be the candidate morphemes for further processing. Using defined rules, a syntactic tree is built and a UNL format semantic network is designed for the input sentence until all the words are inputted.

3.3.3. UNL graph-based similarity

The generated UNL graphs of two sentences are compared for computing semantic similarity using the formula given below. The average of relation match scores and universal word match scores is the actual relation score. The matching attributes of the corresponding universal word decides the universal word match score.

$$\text{Semsim}(S1, S2) = \frac{S1 \cap S2}{S1 \cup S2} + \sum_{a \in S1, b \in S2} \left(0.75 * \frac{[[a.w1 = b.w1 \& a.w2 = b.w2]]}{|S1 \cup S2|} + 0.75 * \frac{[[a.r = b.r \& a.Ew1 = b.Ew1 \& a.Ew2 = b.Ew2]]}{|S1 \cup S2|} + 0.6 * \frac{[[a.Ew1 = b.Ew1 \& a.Ew2 = b.Ew2]]}{|S1 \cup S2|}\right)$$

Example: Input Sentence1: होळी सण साजरा करतांना पर्यावर्ण काळजी घ्यावी

————–UNL expression for Sentence1————————

{unl} mod (काळजी (iof>action):01, पर्यावर्ण(iof>nature):00 @entry@pred)

seq (काळजी (iof>action) : 01, साजरा(icl>do):02@present) agt (सण (iof>event):02@def, साजरा(icl>do):02@present)

aoj (होळी (icl>event):03@def, सण(iof>event):02@def) {/unl}

————-UNL expression for Sentence1————————

Input Sentence 2: पर्यावर्ण भान ठेवून होळी सण साजरा

————–UNL expression for Sentence2————————

{unl} mod (भान (iof>action):01, पर्यावर्ण (iof>nature):00 @entry@pred) seq (भान (iof > action):01,साजरा(icl>do):02 @present)

agt (सण (iof>event):02@def,साजरा(icl>do):02@present) aoj (होळी (icl>event):03@def,सण(iof>event):02@def) {/unl}

————–UNL expression for Sentence2————————

UNL_1 and UNL_2 denote the UNL graph for Sentence 1 and for Sentence 2, respectively. Hence, the semantic Similarity score can be calculated using following formula,

$$S1 \cap S2 = \{\text{होळी, सण, साजरा, पर्यावर्ण}\} = 4$$

$$S1 \cup S2 = \{\text{होळी, सण, साजरा, पर्यावर्ण, काळजी, भान}\} = 6$$

$$\text{Hence, Semsim}(S1, S2) = 0.59$$

3.4. Overall similarity

The scores of maximum statistical similarity and semantic similarity are combined using the following formula for the overall similarity calculation of two sentences, Sim (S1, S2) = $p^*Stat_{sim}(S1, S2)+q^*Sem_{sim}(S1, S2)$

The values of p and q are considered in such a way that, $p + q = 1$; $p, q \in [0, 1]$ as p and q are the constants that represent involvement of each part to the overall similarity calculation,

3.5. Paraphrase judgment

As the proposed system produced an overall similarity score of 0.75, it can easily be concluded that the Marathi sentence pair taken for paraphrasing is identified as a paraphrase pair. Taking into consideration all 5 statistical and 1 semantic measures, it can be detected that there are variation in every similarity measure score. However, the overall similarity score provides an enhanced outcome for the paraphrase detection process.

4. Results and discussion

4.1. Working of the system

The input to the Paraphrase Identification system will be two Marathi text documents in ".txt" format. One document will be the original sentence and the other will be the suspicious paraphrase sentence.

Since the paraphrase detection system is designed exclusively for Marathi sentences, all characters that do not belong to Devanagari script must be eliminated to save processing time and memory. The pre-processing of sentences undergoes further processing steps like tokenization, stop word removal, stemming and morphological analysis and will result in validated output.

The overall similarity calculation is done by adding values of maximum statistical similarity and semantic similarity. The values of both similarities are multiplied by a smoothing factor before addition to take into account the contribution of each similarity to overall similarity. Whether paraphrasing is involved in the given sentence pair or not is decided on the basis of the score of overall similarity. If the score is greater than 0.8, then the sentences are paraphrases of each other. On clicking on the overall similarity button, the system generates the overall similarity score, on the basis of which a judgment on paraphrase is generated.

4.2. Experimental analysis

All sentence pairs from a dataset of five domains were tested domain-wise and similarity-wise. Graphs were created for comparative study of each similarity measure and to find the best for the system. The best method taking it into consideration while calculating overall similarity score for the system.

4.2.1. Domain-wise statistical and semantic similarity

Sentence pairs from the political, sports, film, literature and miscellaneous domains were tested separately for both types of similarities. Statistical similarity deals with statistical structure between words of the sentences, whereas semantic similarity deals with the meaning. For the literature domain and the miscellaneous domain, statistical similarity score lies between 0 and 1. The semantic similarity score lies between 0 and 2 for literature domain and ranges from 0 to 3 for the miscellaneous domain.

4.2.2. Overall similarity

Out of 4 statistical similarities, maximum statistical similarity is calculated for each sentence pair. Combination of the semantic similarity and maximum statistical similarity is considered as overall similarity. The overall similarity score is used for setting the threshold value, based on which paraphrase (P) and non-paraphrase pairs (NP) are identified.

4.3. Threshold value generation

From the different similarity scores thus obtained, it is necessary to highlight a classical problem in classi?cation - thresholds. After computation and comparison of all similarity measures, it is essential to decide upon some threshold value based on which the system decides whether a sentence pair is a paraphrase or not.

Thresholds are parameters that facilitate the process of evaluation. In the evaluation process, no threshold value is pre-defined. Instead, from overall comparison of each similarity on each domain, the best threshold is computed upon which each domain is agreed. Threshold computation may be a classical issue of function maximization or optimization.

The threshold values play important role in paraphrase judgment as sentence pairs can be divided into 3 groups as per their overall similarity scores viz, Paraphrase (P), Semi-paraphrase (SP) and Non-Paraphrase (NP). **Table 3** gives an example of overall similarity score of sentence pair and the role of threshold in deciding whethe it is a Paraphrase (P), Semi-paraphrase (SP), and Non-Paraphrase (NP).

4.4. Performance evaluation

The performance of the system is evaluated on the basis of 4 basic evaluation measures. These measures are accuracy, precision, recall and F-measure. For calculation of these four performance measures, the four parameters TP, TN, FP, and FN must be calculated first.

The performance of the system was evaluated by creating a confusion matrix and it is given in **Table 4**.

46 sentence pairs were classified as similar and 12 as dissimilar out of 58 similar sentence pairs by the proposed system. Out of the actual 16 dissimilar sentence pairs, the 15 sentence pairs were classified as dissimilar and 1 as similar in the used data-set. In above table, True positive (TP) and True Negative (TN) are the observations that are correctly

TABLE 1 | Summary of literature review.

SNo	Paper		Algorithm	Accuracy	*F*-score
Language: English					
1	**Title**	Paraphrase detection using semantic relatedness based on Synset Shortest path in WordNet	Semantic relatedness approach	71.1%	81.1%
	Author	Lee and Cheah (3)			
2	**Title**	Learning paraphrase identification with structural alignment	Alignment-based Approach	78.6%	85.3%
	Author	Liang et al. (4)			
3	**Title**	Paraphrase detection based on identical phrase and similar word matching	SimMat metric combined with 8 MT metrics	77.6%	83.9%
	Author	Nguyen-Son et al. (5)			
4	**Title**	Paraphrase detection using string similarity with synonyms	Use of synonyms in text similarity metrics.	70.6%	80.4%
	Author	Lee and Cheah (6)			
5	**Title**	Re-examining machine translation metrics for paraphrase identification	Combination of 8 MT metrics.	77.4%	84.1%
	Author	Madnani et al. (7)			
6	**Title**	Paraphrase identification using weighted dependencies and word semantics	Calculated similarity and dissimilarity scores using dependencies.	72.41%	81.32%
	Author	Lintean and Rus (8)			
7	**Title**	A semantic similarity approach to paraphrase detection	JCN word similarity with matrix	74.1%	82.4%
	Author	Fernando and Stevenson (9)			
8	**Title**	A metric for paraphrase detection	Paraphrase detection using SumoMetric	78.19%	80.92%
	Author	Cordeiro et al. (10)			
9	**Title**	Paraphrase identification on the basis of supervised machine learning techniques	Combination of lexical and semantic features	76.6%	79.6%
	Author	Kozareva and Montoyo (11)			
10	**Title**	CFILT-CORE: semantic textual similarity using universal networking language	Described a syntactic and word level similarity method combining with a semantic extraction method which was based on Universal Networking Language (UNL) for finding semantic similarity score between a pair of sentences.		
	Author	Dan and Bhattacharyya (1)			
Language : Hindi					
11	**Title**	A novel approach to paraphrase Hindi sentences using NLP	Creating paraphrase by applying synonym and antonym replacement.		
	Author	Sethi et al. (12)			
12	**Title**	Paraphrase detection in Hindi language using syntactic features of phrase	Random forest classifier used for classification and Leven-shtein Distance method used for similarity score calculation for identifying paraphrases in the Hindi language		
	Author	Bhargava et al. (2)			
13	**Title**	Hindi paraphrase detection using natural language processing techniques and semantic similarity computations	Discovered synonym WordNet using a semantic similarity metric and classification done by paraphrase detection decision tree classifier.		
	Author	Vani and Deepa (13)			
14	**Title**	A novel paraphrase detection method in Hindi language using machine learning	Random forest classifier and different machine learning algorithms were used for removing overlapping words and normalized IDF scores for paraphrase detection were calculated.		
	Author	Saini (14)			
Language : Malayalam					
15	**Title**	Malayalam paraphrase detection	Procured CUSAT Malayalam Wordnet for calculating sentence similarity using matching tokens, lemmas and synonyms.		
	Author	Sindhu and Mary Idicula (15)			

(Continued)

TABLE 1 | (Continued)

SNo	Paper		Algorithm	Accuracy	*F*-score
16	**Title**	Detecting paraphrase in Indian languages-Malayalam	Calculated cosine similarity score using model named Bag of Word and a threshold for calculating paraphrase score.		
	Author	Manju and Mary Idicula (16)			
17	**Title**	Paraphrase identification using Malayalam sentences	Detecting paraphrase by Statistical measure and semantic analysis		
	Author	Mathew and Mary Idicula (17)			
Language : Tamil					
18	**Title**	Detection of Paraphrases on Indian Languages	Tamil Shallow parser used for extracting sixteen diverse morphological features.		
	Author	Thangarajan et al. (18)			
Languages (Tamil, Malayalam, Hindi, Punjabi)					
19	**Title**	Detecting paraphrases in Indian languages based on gradient tree boosting	Employed a Cosine, Dice Distance, Jaccard Coefficient and METEOR features and Gradient Boosting Tree supervised classification method.		
	Author	Kong et al. (19)			
20	**Title**	Language independent paraphrases detection	Probabilistic Neural Network (PNN) was used to train Jaccard, Cosine Similarity and length normalized Edit Distance language independent feature-set to detect paraphrases in Indian languages.		
	Author	Sarkar et al. (20)			
21	**Title**	Detecting paraphrases in Indian languages using multinomial logistic regression model	Described lexical and semantic level similarities between two sentences using a trained multinomial logistic regression model for paraphrase identification and to attain maximum f-measure		
	Author	Sarkar (21)			
22	**Title**	Paraphrase detection in indian languages - a machine learning approach	The team worked on Hindi, Tamil, Punjabi and Malayalam sentences. The paraphrase decision is taken into consideration by various similarity measures, machine translation evaluation metrics and machine learning framework.		
	Author	Saikh et al. (22)			

TABLE 2 | Similarity measure score.

Types of similarity	Measure	Value	Similarity measure score to consider
Statistical similarity measures	Jaccard Similarity (Word set)	0.4	Maximum Statistical Similarity $Stat_{sim}$(S1,S2) = 0.91
	Cosine Similarity (Word Vector)	**0.91**	
	Word order Similarity	0.17	
	Sumo Metric	0.7	
Semantic similarity measure	**UNL graph based Similarity**	**0.59**	Semantic Similarity Sem_{sim} (S1,S2) = 0.59
Overall Similarity	**If p = 0.5, q = 0.5, then Sim (S1, S2) = 0.5*0.91 + 0.5*0.59 = 0.75**		

predicted and shown in green color. False positive (FP) and False Negative (FN) must be minimized in order to get a good system so they are shown in red.

True positive (TP) and True Negative (TN) values occur when actual results obtained from the system agree with the predicted results.

False positives (FP) and False negatives (FN) occur when the actual results contradict with the predicted results.

4.4.1. Accuracy

Accuracy is the ratio of correctly predicted observation to the total observations.

4.4.2. Precision

Precision is obtained by dividing the number of correctly predicted paraphrases by the number of predicted paraphrases. High precision relates to the low false positive rate.

4.4.3. Recall

Recall calculation is done taking the ratio of correctly predicted paraphrases to the all referenced paraphrases in actual class.

TABLE 3 | Threshold selection.

Sentence pair	Overall similarity score	Threshold-value	Paraphrase judgment
Sentence 1: पुस्तक हा माणसाचा श्रेष्ठ मित्र आहे (Books are man's best friends.) **Sentence 2:** पुस्तक मनुष्याच्या जवळचा मित्र आहे Books are close friends of man.	1.22 (SM1 from miscellaneous domain)	(score $\geq$ 0.5)	Paraphrase (P)
Sentence 1:महाराष्ट्रातील प्रत्येक प्रांताची मराठी भाषा निराळी आहे Each region of Maharashtra has a different Marathi language. **Sentence 2:**मराठी भाषा सर्व महाराष्ट्राची जरी एक असली तरी तीच दर बारा कोसावर बदलते Marathi language is the only one in the whole of Maharashtra, the same rate varies from 12 to 12 degrees.	0.48 (SL5 from literature domain)	(score $<$ 0.5 && score $>$ 0.4)	Semi-paraphrase (SP)
Sentence 1: होळीच्या वेळी झोळी सद्गुणांनी भरावी Beans are to be filled with virtues at the time of Holi. **Sentence 2:**होळीसाठी वाईट रुढ़ी व प्रथांची मोळी बांधावी. Build bad rallies and rituals for Holi.	0.0 (SM4 from miscellaneous domain)	(score = 0.0)	Non-paraphrase (NP)

4.4.4. F-measure

F-measure considers false positives and false negatives into account and it is the weighted average of Precision and Recall. Accuracy gives the best value when the values of false positives and false negatives are similar. When the cost of these two is different, then values of both Precision and Recall are considered from which in turn F-measure is calculated.

Although there is deviation in the similarity score of each method, the overall similarity score proved to be an improved result. This system considered all the statistical and semantic concerns.

The data obtained from the above table prove that the proposed system performs efficiently for Paraphrase Identification of Marathi sentences.

TABLE 4 | Confusion matrix for performance evaluation.

Actual results	Predicted results				Total dataset
	Similar		Dissimilar		
Similar	46	TP	12	FN	58
Dissimilar	1	FP	15	TN	16

TABLE 5 | Performance analysis of overall similarity.

Measure	Value
Accuracy	0.82
Precision	0.98
Recall	0.79
F-measure	0.89

5. Dataset and testing

5.1. Dataset

The dataset used in this work consist of sentence pairs from the political,film,sports,literautre,miselleneous domains.

There are 74 Marathi sentence pairs in the dataset and each sentence pair has a binary decision accompanying with it which gives the judgment on whether it is a paraphrase pair or not.

5.2. Testing

To test the accuracy and F-measure, the dataset was divided into 75% and 25% for training and testing, respectively. The performance measures (Accuracy, Precision, Recall, and F-Measure) were evaluated for different similarity measures.

A quartile is a form of quantile. 25% is set as the first quartile (Q1) as it is the mid of the smallest number and the median of the data set. The median of the data-set is the second quartile (Q2). The mid value between the median and the highest value of the data-set is considered as the third quartile (Q3).

The decision of paraphrase of the sentence pair depends on the score of first quartile. If the similarity score is greater than Q1, then the sentence pair is labeled as paraphrase, otherwise non-paraphrase. On the basis of this decision accuracy, precision, recall, and F-measure were calculated and compared for each similarity measure.

In paraphrase identification of Marathi sentences, 5 similarity measures were performed to check the similarity score of the sentences. The similarity measures were performed on 58 sentence pairs to identify the highest score among 4 statistical similarities and the performance of semantic similarity.

TABLE 6 | Performance of the similarity measures.

Similarity measure	Min	Quartile 25%	Quartile 50% Median	Quartile 75%	Max
Jaccard	0	0.18	0.29	0.4	0.86
Cosine	0.74	0.885	0.91	0.94	1
Word order	0	0.065	0.12	0.24	4
Sumo metric	0	0.01	0.04	0.72	1
Semantic (UNL)	0	0.235	0.38	0.79	2.58
Overall	0.42	0.58	0.64	0.87	0.87

TABLE 7 | Performance measure analysis.

Measure	Jaccard	Cosine	Word-order	Sumo metric	UNL	Overall
Accuracy	0.77	0.87	0.8	0.83	0.77	0.82
F-measure	0.85	0.92	0.87	0.89	0.85	0.89
Recall	0.77	0.95	0.75	0.86	0.77	0.79
Precision	0.94	0.9	1	0.92	0.94	0.98

5.2.1. Analysis of performance measures

As the first quartile value is determined to be the threshold for each similarity, the confusion matrix is created and values of performance measures are calculated according to it.

From **Table 6** it is clear that cosine similarity gives the best score among all similarity methods. Hence for finding paraphrases in Marathi Sentences, the Cosine similarity method is effectively accurate.

After evaluating all similarity measures, it can be clearly seen that cosine similarity is the best method among all statistical similarity techniques for finding similarity. When considering all sentences in the given dataset, other statistical techniques also performed well for different sentence pairs. Word-order similarity is the best to find dissimilar pairs.

For taking dissimilarity into consideration, the overall similarity score is 0 if any of the statistical similarity or semantic similarity values is 0. Otherwise, it is calculated by combining the maximum statistical similarity score and the semantic similarity score.

6. Conclusion and future scope

Mostly the research has been done for English and Indian regional languages such as Hindi, Malayalam, Punjabi and Tamil. However, no paraphrase detection work has been yet done for the Marathi language. This is the first footstep toward detecting Paraphrases in Marathi sentences. The development of social media in Marathi makes available massive data in the Marathi language. Analysis of social data like daily tweets in Twitter is a field of growing interest for different purposes. Identifying paraphrases should be widely explored to help many more arenas of the Natural language processing field.

The proposed system provides a paraphrase identification tool for Marathi sentences using incorporation of both statistical and semantic features. The detailed analysis and performance comparison of all statistical methods concludes that the cosine similarity measure outperforms all the other statistical measures. Semantic similarity carried out through a UNL method plays an important role in achieving 82% accuracy and 89% F-measure. This score proves that the designed tool is extremely effective in Paraphrase Identification of Marathi Sentences.

The system can be enhanced for capturing the similarity between Marathi paragraphs and Marathi documents. There is possibility of improving statistical methods to detect more structural similarity types. Misspelled words in the sentences will yield wrong results even if the sentences are similar. Pre-processing steps can be enriched to capture spelling mistakes and to process accordingly. Semantic similarity measure scores can be improved by considering synonyms of universal words. The current model has been experimented on fewer data sets, which can be extended for more data. The model constructed can also be changed from static to online Paraphrase Identification tool for Marathi documents.

References

1. Dan A, Bhattacharyya P. CFILT-CORE: Semantic Textual Similarity Using Universal Networking Language, 2013 Association for Computational. (2013).
2. Bhargava R, Baoni A, Jain H. *BITS_PILANI@DPIL-FIRE 2016: Paraphrase Detection in Hindi Language Using Syntactic Features of Phrase*. Kolkata: DPIL (2016).
3. Lee JC, Cheah Y. Paraphrase detection using semantic relatedness based on Synset shortest path in WordNet. In: *Proceedings of the International Conference on Advanced Informatics: Concepts, Theory and Applications*, Penang (2016).
4. Liang C, Paritosh P, Rajendran V, Forbus KD. Learning paraphrase identification with structural alignment. In: *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI-16)*, New York, NY (2016).
5. Nguyen-Son H, Miyao Y, Echizen I. Paraphrase detection based on identical phrase and similar word matching. In: *Proceedings of the 29th Pacific Asia Conference on Language*, Shanghai (2015).
6. Lee JC, Cheah Y. Paraphrase detection using string similarity with synonyms. In: *Proceedings of the 4th Asian Conference on Information Systems, ACIS 2015*. (2015).
7. Madnani N, Tetreault J, Chodorow M. Re-examining machine translation metrics for paraphrase identification. In: *Proceedings of the Conference of the North American Chapter of the ACL: 2012 Association for Computational Linguistics*, Montreal, CA (2012).
8. Lintean M, Rus V. Paraphrase identification using weighted dependencies and word semantics. In: *Proceedings of the 22nd International FLAIRS Conference*. (2009).
9. Fernando S, Stevenson M. A semantic similarity approach to paraphrase detection. In: *Proceedings of the 11th Annual Research Colloquium of the UK Special Interest Group for Computational Linguistics*. Princeton, NJ: Citeseer (2008).

10. Cordeiro J, Dias G, Brazdil P. A metric for paraphrase detection. In: *Proceedings of the International Multi-Conference on Computing in the Global Information Technology (ICCGI'07)*. Guadeloupe: IEEE (2007).
11. Kozareva Z, Montoyo A. Paraphrase identification on the basis of supervised machine learning techniques. In: *Proceedings of the 5th International Conference on NLP (FinTAL 2006), Advances in Natural Language Processing*, Turku (2006).
12. Sethi N, Agrawal P, Madaan V, Singh SK. A novel approach to paraphrase Hindi sentences using natural language processing. *Indian J Sci Technol*. (2016). ***VOLPGQ.
13. Vani K, Deepa G. *ASE@DPIL-FIRE2016: Hindi Paraphrase Detection using Natural Language Processing Techniques & Semantic Similarity Computations*. Kolkata: DPIL (2016).
14. Saini A. *Anuj@DPIL-FIRE2016: A Novel Paraphrase Detection Method in Hindi Language using Machine Learning*. Kolkata: DPIL (2016).
15. Sindhu L, Mary Idicula S. *CUSAT_NLP@DPIL-FIRE2016:Malayalam Paraphrase Detection*. Kolkata: DPIL (2016).
16. Manju K, Mary Idicula S. *CUSAT TEAM@DPIL-FIRE2016: Detecting Paraphrase in Indian Languages-Malayalam*. Kolkata: DPIL (2016).
17. Mathew D, Mary Idicula S. *Paraphrase Identification of Malayalam Sentences–An Experience*. (2013).
18. Thangarajan R, Kogilavani SV, Karthic A, Jawahar S. *KEC@DPIL-FIRE2016: Detection of Paraphrases on Indian Languages*. Kolkata: DPIL (2016).
19. Kong L, Hao Z, Chen K, Han Z, Tian L, Qi H. *HIT2016@DPIL-FIRE2016: Detecting Paraphrases in Indian Languages Based on Gradient Tree Boosting*. Kolkata: DPIL (2016).
20. Sarkar S, Pakray P, Saha S, Das D, Bentham J, Gelbukh A. *NLP-NITMZ@DPIL-FIRE 2016: Language Independent Paraphrases Detection*. Kolkata: DPIL (2016).
21. Sarkar K. *KS_JU@DPIL-FIRE2016: Detecting Paraphrases in Indian Languages Using Multinomial Logistic Regression Model*. Kolkata: DPIL (2016).
22. Saikh T, Naskar SK, Bandyopadhyay S. *JU_NLP@DPIL-FIRE2016: Paraphrase Detection in Indian Languages–A Machine Learning Approach*. Kolkata: DPIL (2016).