I.J. Intelligent Systems and Applications, **2021, 3, 23-34**
Published Online June 2021 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijisa.2021.03.03

Influence of GUJarati STEmmeR in Supervised Learning of Web Page Categorization

Chandrakant D. Patel
Hemchandracharya North Gujarat University, Patan, Gujarat, India
E-mail: cdpatel4phd@gmail.com

Jayesh M. Patel
Acharya Motibhai Patel Institute of Computer Studies, Ganpat University, Kherva, Gujarat, India
E-mail: drjayeshpatel.phd@gmail.com

Received: 21 November 2020; Revised: 02 January 2021; Accepted: 19 February 2021; Published: 08 June 2021

Abstract: With the large quantity of information offered on-line, it's equally essential to retrieve correct information for a user query. A large amount of data is available in digital form in multiple languages. The various approaches want to increase the effectiveness of on-line information retrieval but the standard approach tries to retrieve information for a user query is to go looking at the documents within the corpus as a word by word for the given query. This approach is incredibly time intensive and it's going to miss several connected documents that are equally important. So, to avoid these issues, stemming has been extensively utilized in numerous Information Retrieval Systems (IRS) to extend the retrieval accuracy of all languages. These papers go through the problem of stemming with Web Page Categorization on Gujarati language which basically derived the stem words using GUJSTER algorithms [1]. The GUJSTER algorithm is based on morphological rules which is used to derived root or stem word from inflected words of the same class. In particular, we consider the influence of extracted a stem or root word, to check the integrity of the web page classification using supervised machine learning algorithms. This research work is intended to focus on the analysis of Web Page Categorization (WPC) of Gujarati language and concentrate on a research problem to do verify the influence of a stemming algorithm in a WPC application for the Gujarati language with improved accuracy between from 63% to 98% through Machine Learning supervised models with standard ratio 80% as training and 20% as testing.

Index Terms: Stemming, Gujarati Language, Supervised algorithms, Machine Learning, Accuracy.

1. Introduction

In Stemming, morphologically similar words are clustered together under the hypothesis that they are semantically similar [2]–[7]. It is useful for Text Mining, Natural Language Processing (NLP) functions, Text clustering, Text categorization, Text summarization, and application of Text Mining (TM) [1], [8, 9]. It is the initial stage before applying any kind of related categorization algorithm. Additionally, word-stemming is also supportive features for the indexing and search engine, now a day. The main goal of stemming is to remove inflectional or derivational ending from word to its stem form. Stemming, a very beneficial implement under the province of Information Retrieval (IR), supported by nearly every modern indexing and search system at the time of indexing and searching.

Several inflectional and/or derivation stemming algorithms were developed for European and Asian languages [9]. Stemming is a linguistic process in which the various morphological variants of the words are mapped to their base forms [5, 10]. The algorithms or the computer programs that perform the stemming process are called stemmers. Stemmer’s designs range from the simplest techniques, such as the elimination of affixes using a list of repeated affixes, to a more complex design that uses the morphology of words in the inference process to derive a stem.

According to J B Lovins [11], the stemming algorithm can be classified as iteration, longest match, and context-free and context-sensitive. However, the current state of the art approaches of stemmer can be classified as – Language Dependent and Independent including Corpus, Statistical, and Hybrid approaches in [Fig.1]. Each type of stemmer has a typical way to find the stem from variants of the word. The power of a stemmer is the volume of saving in the mass of the vocabulary is acquired. Aggressive or Strong stemmers may decrease the size of the directory for a given corpus extremely.

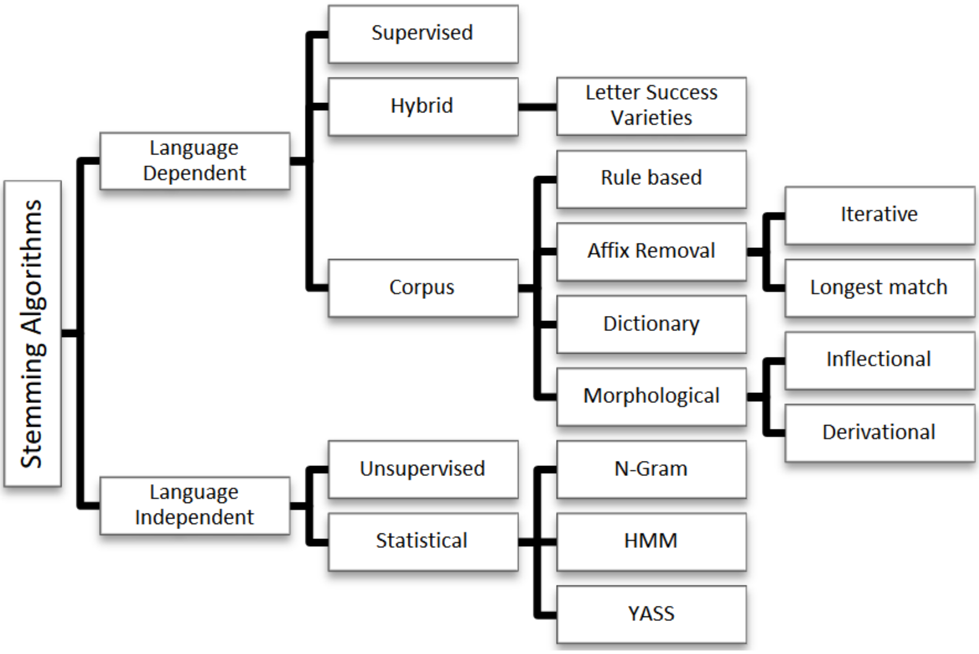

Fig.1. Categories of Stemmer algorithm

Many works have been countered on text classification in which the stemming algorithm has been used [12, 13]. The opinions of research are different from each other in the effectiveness of stemming for the aim of classification. While E. Riloff [14] and M. Spitters [15] had over that stemming might not facilitate increase classification accuracy, Buseman had discovered that morphological analysis will increase the performance for a series of classification algorithms applied to German e-mail classification. In recent work, Gustad and Bouma [16] determined that stemming doesn't systematically improve classification accuracy. Additional, however, Cohen et al. [17] found stemming advantageous whereas classifying medical documents. WPC is widely used in the area of vertical search, personalized search and recommendation system, etc. The impact of IR not only searching important terms but also massively depends on the methods of counting words which basically depends on tokenization, stemming and stop word removal process as a pre-processing step before applying text classification.[18] The researchers have following kind of research problems with research focus on influence of stemming algorithm in WPC application.

1. How the morphological rules of Gujarati words are impact.
2. Influence of stemming algorithms on WPC.
3. Which supervised models is most effective while categorization of Gujarati Web page.

The rest of this paper is structured as follows. Second section reviews earlier literature on the stemming and WPC particularly on Gujarati, Hindi and English language. Section three denotes research motivation which indicates modules wise in depth information for it with proposed methodology for WPC is presented. In section four discussed with data analysis and algorithm discussion with result on machine learning supervised models presented and implemented in this research.

2. Related Work

Numerous typical methods that accomplish stemming are in the literature for three languages, including English [Table 1], Hindi [Table 2], and Gujarati [Table 3] with comparison.

Table 1. Comparison study of English stemmer

Sr. No.	Authors Name & Reference	Data sets	Year	Approach	Evolution methods or Accuracy in %
1	Julie Beth Lovins [11]	Not Given	1968	Longest Matching	Not Given
2	Margaret A. Hafer et. al. [19]	Brown, CPC, ADI	1974	Letter Successor	Precision, Recall
3	M. F. Porter [19, 20]	Cranfield	1980	Suffixes Stripping	Precision, Recall
4	Donna Harman [22]	Cranfield, Medlars, CACM	1991	Not Given	Ranking with Noise, IDF measure
5	J. L. Dawson et. al. [23]	Not Given	1974	Not Given	Not Given
6	C. D. Paice et. al. [24]	Not Given	1994	CISI	UI, OI
7	Xu and W. Croft et. al. [5]	WEST, WSJ, WSJ91, ISM	1998	Corpus-based	Precision, Recall
8	Robert Kroverz et. al. [25]	CACL, NPL, TIME, WEST	1992	Inflectional, Derivational	Precision, Recall
9	James Allan et. al. [26]	AP89, AP90, FBIS, TREC5, TREC8	2003	Corpus-based	Precision, Recall
10	Bhamidipati Narayan L et. al. [27]	20NG, WebKB, WSJ	2007	Corpus-based	Precision
11	Vairaprakash Gurusamy et. al. [28]	Moby	2017	Performance Analysis	Accuracy

In [Table 1], a wide variety of approaches developed in six decades stemming from the English language. Most of the stemming algorithms are the corpus-based and suffixes approach which is most suitable for the English language.

Table 2. Comparison study of Hindi stemmer

Sr. No.	Authors Name & Reference	Data sets	Year	Approach	Evolution methods or Accuracy in %
1	Ananthakrishnan Ramanathan et. al. [29]	Hindi news	2003	Suffix removal, Lightweight stemmer	Promising result
2	Leah Larkey et. al. [30]	Hindi news	2003	Lightweight	Not Given
3	Amaresh Kumar Pandey et. al.[31]	EMILLE	2008	Unsupervised	Accuracy 89.9%
4	Niraj Aswani et. al. [32]	EMILLE	2010	Morphological Analyzer	Precision, Recall
5	Upendra Mishra et. al. [33]	Manually	2012	Hybrid	Accuracy 91.59%
6	Anubha Jain et. al. [34]	FIRE	2013	Hybrid	MAP 3.40%
7	Vishal Gupta [35]	EMILLE	2014	Rule-based (noun)	Accuracy 83.65%
8	Rakhi Joon et. al. [36]	CFILT	2017	MWES (N-gram)	Accuracy >90%

In [Table 2], a wide opportunity taken into stemming algorithms with rule-based, hybrid, unsupervised, n-gram, and lightweight for National language Hindi on different kinds of a corpus such as CFILT, FIRE, and EMILLE. The Hindi language got maximum stemming accuracy of 91.59% using a hybrid approach [21].

Table 3. Comparison study of Gujarati stemmer

Sr. No.	Authors Name & Reference	Data sets	Year	Approach	Evolution methods or Accuracy in %
1	Prasenjit Majumder et. al. [3]	TREC, CLEF	2007	Statistical	Precision, Recall
2	Pratik Patel et. al. [37]	EMILLE	2010	Hybrid	Accuracy 67.86%
3	Kartik Suba et. al. [38]	EMILLE	2011	Hybrid Inflectional, Rule-based derivational	HI Accuracy 90.7% RD Accuracy 70.7%
4	Juhi Ameta et. al. [39]	EMILLE	2011	Lightweight	Accuracy 91.5%
5	Jikitsha Sheth et. al. [40]	Not Given	2011	Model proposed	Not Given
6	Jikitsha Sheth et. al. [41]	EMILLE	2012	Hybrid	Accuracy 92.41%
7	Bijal Dalwadi et. al. [42]	EMILLE	2016	Affix removal	Accuracy 96.63%
8	Miral Patel et. al. [43]	MIRI	2015	Suffix stripper (noun)	Accuracy 85.83%

In [Table 3], a hybrid approach is applied for the stemming of the Gujarati language and evaluated on the EMILLE corpus. The rule-based approach is most suitable for the Gujarati language as per the result taken by the researcher that is a maximum of 96.63% for affix removal and 92.41% for the hybrid approach. The first efforts made in 2011 for the

Derivational based stemming; the accuracy was up to 70.7%. In the literature review, the researcher found very few resources for stemming processing on the Gujarati language. So, it is an urgent need to develop a mature stemming algorithm which can help for several Indian languages including Gujarati, Marathi [44], Odia, and Pashto.

3. Methods

3.1. Research Motivation

Web page classification (WPC), also known as Web Page Categorization, is the process of assigning a Web page to one or more predefined category labels [45]. The research concentrates on stemming which involves the following major modules:

A. Pre-processing for the Gujarati language web page

The first module of pre-processing mentioned in the above paragraph as the first step in web page classification process includes only a small percentage of the pre-processing sub-steps: Cleaning or normalizing the data (remove all Non-Gujarati Alphabets, Numerals, and Symbols including HTML tags), Tokenization, and generate the Gujarati words lists.

B. Identifying the stop words list

This module processed only those kinds of words which are less important for categories of the web page. The techniques of stop words identification such as static list or dynamic lists using rules, statistical or hybrid approach.

C. Finding the stem of a word

The Stemming process is language dependent as well as independent. The Information Retrieval System works accurately when there is standard stemming process for all languages. Generally, stemming algorithm is replaced or removes affixes to the word to generate the root word.

D. Evaluation of stemmer on WPC

The fourth module is performed the stemming evaluation using WPC. In the past few years, there are several stemming algorithms, and WPC methods have been developed for the non-Asian languages. Standard benchmark corpora need a predefined training/testing splits or k-fold cross validation statistical methods to allow comparison against different WPC approaches. Otherwise, a typical training/testing split of a labelled corpus is 80: 20.

3.2. Research Methodology

As per the constitution of India, 22 languages are officially included in the census of 2011. The record mentioned that the Gujarati language is scheduled language and categorized as Gujarati, Gujrao / Gujrau, Pattani, Ponchi, Saurashtra / Saurashtri, and others. The Gujarati language got the rank of 6 with 4.58% speakers in the 2011 census and that was increased by 20.40% in the decade of 2001-2011. The family of the Gujarati language mentioned as a member of Indo-Aryan. It is closely allied to Marathi, Hindi, Bengali, and Punjabi [46]. In today’s world, the Gujarati is spoken by more than 68 million people including Gujarat, Rajasthan, Karnataka, Madhya Pradesh, and Maharashtra. Moreover, it is also used beyond India like in USA, Kenya, Pakistan, Fiji, Bangladesh, Oman, South Africa, UK, Uganda, and Zambia. The Jain monk Hemchandracharya, a renowned scholar of Gujarat from the 8th century, composed the grammar book 'Siddhem Shabdanusasan – सिद्ध्र-हेम शब्दानुशासन' in addition to the Sanskrit language, which is why his praises are sung today not only in Gujarat but also in India. [46, 47].

The approach to the Gujarati WPC proposed in this paper is based on contemporary methods for performing automatic textual classification which initiates operation are remove stop words identification and stemming. This proposed method involves extracting the Gujarati language textual from the web page, encoding the document as features vector with machine learning models. This subsection gives a brief overview of the methodology used in this paper. The methodology is summarized in.

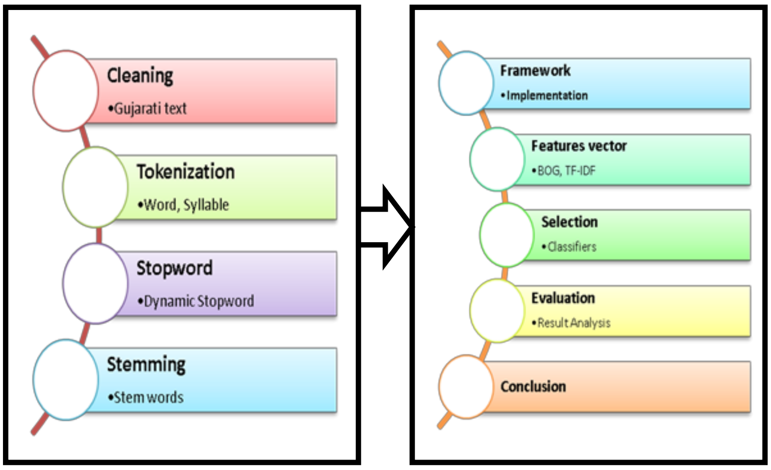

Fig.2. Overview of Research Methodology in WPC

This research methodology logically divided into two sections i.e. up to stemming and classification. The stemming section considered with several subsections such as cleaning, tokenization, stop words while classifier has features vector, selection, and evaluation with a conclusion as depicted in [Fig.2]. In the stemming section, the researcher studied relevant algorithm and approaches which are performed to identify stop words and to do stemming to get an understanding of the field and find a suitable approach and develop an algorithm that can be used for the classifier section. In the classifier section, researcher has found a suitable machine learning algorithm and there are no hard criteria for selecting a specific algorithm or method. This is followed by proposing a general method of Gujarati WPC using text classification methods. To evaluate this method, a framework for the Gujarati WPC is developed that implements the general method and provides support for several algorithms that have been considering for study. The general method performs, to establish how well several experiments are done by performing Gujarati WPC using the framework (with the several classifiers) on standard data sets.

4. Algorithm and Data Analysis

4.1. Algorithm Discussion

The algorithm has been defined with specific purpose for the given set of inputs. In this discussion, we have implemented proposed algorithm with input and desired output such as syllable tokenizer, search dynamic stop words and stemming algorithm for the given research work.

A. Syllable Tokenizer algorithm

Tokenization is a process of computer science to break the mass of texts into smaller units. In other words, tokenization is a process of segmenting a string of characters into words. The unit may consider as sentences, words, keywords, phrases, symbols but that are dependent upon application and languages [48,49].

Table 4. Sample output of Gujarati Syllable tokenizer

Word	Syllable of Word
આ	['આ']
કોવીડ	['કો', 'વી', 'ડ']
મહામારીમાં	['મ', 'હા', 'મા', 'રી', 'માં']
લોકડાઉન	['લો', 'ક', 'ડા', 'ઉ', 'ન']
જરૂરી	['જ', 'રૂ', 'રી']
પગલું	['પ', 'ગ', 'લું']
છે	['છે']

Tokenization process applies to larger text to divide into smaller text. In text mining, a tokenization is paragraph > sentences > words > syllable > character. The researcher proposed and implemented Gujarati syllable tokenizer for stemming and stop words operations basis on meaningful syllable with sample output in [Table 4].

```
Step 01    Read a word
Step 02    Mention list of all vowels, consonants and diacritics
Step 03    Declare C-tokens as list and s as string
Step 04    Read each character from word
Step 05    if character in vowel list or consonant list
           Go to Step – 06
           Otherwise
           Go to Step – 10
Step 06    if s last character is Halant ('◌્'):
           Go to Step – 07
           Otherwise
           Go to Step – 08
Step 07    Append current character into string
Step 08    Append string to C-tokens list
Step 09    Create string as empty and add current character into string
Step 10    if character in Diacritic list:
           Go to Step – 11
Step 11    Append current character into string
Step 12    Append string to C-tokens list and return C-tokens
```

B. Dynamic Stop words Identification algorithm

Stop words are frequent words that carry no information [50]. They are topic-neutral words. Removing stop words helps in saving space and simplifies further steps in classification [50, 51]. Choosing appropriate stop words is of language-specific and domain-specific choice, i.e. a word considered as stop word in one domain might be a good discriminator but not in another domain. The usefulness of stop words in web page categorization is also task-dependent [53]. A word in Gujarati language has mainly two kinds as per usage of diacritics such as non-diacritic word or diacritic word. A non-diacritic word doesn't contain Halant ('◌્') sign for example, પર ,and ઉપર while a diacritic word does contain Halant ('◌્') sign for example, અન્યજ્યારે ,, અનેત્યાં ,. To identify words based on its length, we need syllable tokenizer for word into syllable tokens. Such as "ઉપર" > "ઉ","પ","ર" and "જ્યારે" > "જ્યા","રે" Here, we consider count of syllable that is the length of that string. So, in the above example we consider the length of word 3 and 2 respectively.

```
Step 01    Read words and create a list of unique words
Step 02    Read each word from unique words
Step 03    If word is non-diacritic word
           Apply non-diacritic words rules.
           Go to Step – 05
Step 04    If word is diacritic word:
           Apply diacritic words rules.
           Go to Step – 05
Step 05    Append word to stop word list and return stop words
```

C. GUJarati STEmmeR (GUJSTER) algorithm

A word might consist of a stem or root and affixes. The stem is that the main linguistics elements because it contains that means. The role of stemmer is to spot the stem of a given word, such all the inflected words of same stem get mapped there to. Word stemming is one in every of the fundamental method exhausted applications Natural Language Processing (NLP), Information Retrieval (IR), Machine Learning or Translation (MT), Language Modelling (LM) and alternative information applications. Since inception of NLP domain, there are several approach developed to stemming the words for Indian and non-Indian language discussed in literature review [52].

```
Step 01    Input the file (s) with encoding format "UTF-8"
Step 02    Pre-processing steps apply on it
   02.01   Remove HTML tags
   02.02   Remove non-Gujarati character(s)
   02.03   Remove numbers
   02.04   Remove punctuation or symbols
   02.05   Tokenize the words and follows the syllable tokenizer algorithm
   02.06   Filter out to unique words
Step 03    Apply substitute rules on words
Step 04    Remove stop words using dynamic top words algorithm
Step 05    Apply Stemming steps on it using Rule based approach
   05.01   Apply suffixes rules
   05.02   Apply prefixes rules
   05.03   Dictionary based approaches to check stem word
Step 06    Post-processing steps apply on it
```

The text classification is widely used in Natural Language Processing (NLP). Using text classification, the researcher can train model to supervised learning. To categories the Gujarati web page, researcher used supervised learning algorithms that can predicate with labelled corpus. Using label, an algorithm can understand the patterns and correlation between words. The corpus used in this research paper is hybrid in nature. It is collected from several domains such as EMILLE, TDIL and manually. A machine learning models can't work directly on texts. So, features engineering is a process to transform texts into useful numeric features that act as input for that and will improve the quality of the model with enhanced performance. When research has text data, there are several methods to gain features that represent the texts as features sets.

In Bag of Words (BOW), the orders of terms in sentences are not considered and for this application, it is not important that orders terms can classify well than unordered terms. The researcher has selected TF-IDF vectors mechanism to generate the features engineering of the document in the corpus. The valid reasons for selection are...It is simple and easy to understand the results for the particular domain. It is very fast process and takes less waiting. The researcher can implement features generation process to resolve the over fitting problem. In addition, Joshi Hardik et. al. [54] have compared various IR (Information Retrieval) models and concluded that TF-IDF (Term Frequency / Inverse Term Frequency) model performed well for the Gujarati newswire corpus.

A numeric features can easily predicate by machine learning models. So, the researcher must create a mapping for each category with numerical value. The machine learning model quality will prove by the test datasets while training datasets. The researcher has selected randomly 80:20 ratios for the training and test, respectively. The researcher also provide the hyper parameter tuning process to cross validation on the training data and fit with final model and then going to evaluate the model with unseen data to obtain some metrics which is less biased as possible.

4.2. Data Analysis

The stemmer datasets which include all kinds of characters with Gujarati and non-Gujarati for the testing are 1028 files and the average length of sentence in the corpus is 13 to 15 and the average word length is 7. These all kinds of datasets are hybrid approach because few of the resources are collected from internet and others are collected or crafted manually. The datasets collected from EMILLE, TDIL and other sources.

Table 5. Statistics of Datasets

Type	Features	Count
Without Filter (Non-Gujarati & Gujarati Characters)	Total Sentences	862865
	Total Words	11287001
	Total Unique Words	656806
With Filter (Gujarati Characters)	Total Sentences	839466
	Total Words	11080306
	Total Unique Words	485949

The statistics of words are as per the length with the help of the Syllable word algorithm mentioned in [Table 5]. The minimum length of Gujarati words is one and the maximum length of Gujarati words is fourteen in our corpus.

Table 6. Statistics of words per length

Length of Word	Number of unique words	Percentage
1	980	0.250698
2	28956	7.407351
3	91500	23.40698
4	121941	31.19422
5	89859	22.98719
6	41739	10.67742
7	12635	3.23221
8	2632	0.673302
9	526	0.134558
10	92	0.023535
11	27	0.006907
12	18	0.004605
13	3	0.000767
14	1	0.000256

The categories of datasets with lexical diversity of each category mentioned in [Table 6] and lexical diversity calculate based on unique words and total words in [Table 7].

Table 7. Average of lexical diversity of datasets

Category	Files in Category	Average Lexical Diversity in all words	Average Lexical Diversity on Gujarati words
Astrology	10	65.71%	64.43%
Business	15	53.61%	50.53%
Crime	10	69.01%	67.18%
Education	8	45.58%	35.99%
Health	21	52.67%	41.11%
Legal	8	46.24%	37.44%
News	136	51.23%	47.39%
Politics	5	76.98%	74.04%
Sports	15	57.98%	56.26%
Spritual	10	57.48%	56.24%
AsianNet	16	35.39%	31.11%
General_News	37	43.70%	40.03%
NationaL_News	235	55.60%	51.20%
Regional_News	216	37.75%	34.92%
Social	6	53.87%	44.78%
Supplement_News	280	41.16%	37.63%
Grand Total	1028	46.73%	42.90%

5. Results and Discussion

The researcher has tested several machine learning models to work out which one may fit better to the information and properly capture the relationships across the points and their labels. The researcher got only used classic machine learning models rather than deep learning models thanks to the insufficient amount of knowledge. The researcher got, which might probably result in over fit models that doesn't generalize well on unseen data. The researcher has tested & evaluated the following models such as Random Forest, Support Vector Machine, K Nearest Neighbors, Multinomial Naïve Bayes, Multinomial Logistic Regression, and Gradient Boosting.

Table 8. Classification report of multinomial naive bayes in balanced multi-categories models

Classification Report				
Category	**Precision**	**Recall**	**F1-Score**	**Support**
0	1.00	1.00	1.00	1
1	1.00	1.00	1.00	1
2	1.00	0.67	0.80	3
3	1.00	0.50	0.67	4
5	0.50	1.00	0.67	1
9	0.50	1.00	0.67	2
Accuracy			**0.75**	12
Macro Average	0.83	0.86	0.80	12
Weighted Average	0.88	0.75	0.76	12

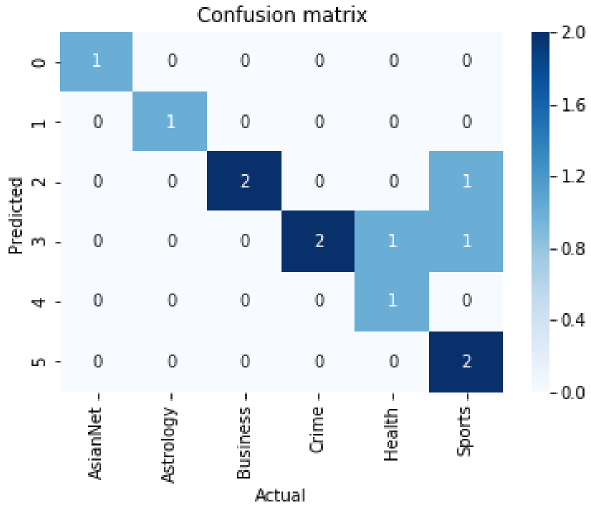

Fig.3. Confusion matrix of Multinomial Naive Bayes in Balanced Multi-categories Models

Table 9. Classification report of logistic regression in balanced single category model

Classification Report				
Category	**Precision**	**Recall**	**F1-Score**	**Support**
0	0.93	0.97	0.95	29
1	0.96	0.93	0.94	27
2	1.00	1.00	1.00	23
3	1.00	1.00	1.00	25
Accuracy			**0.97**	104
Macro Average	0.97	0.97	0.97	104
Weighted Average	0.97	0.97	0.97	104

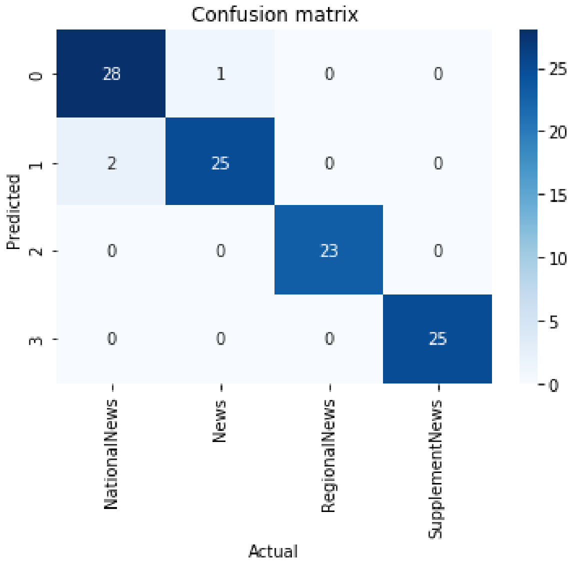

Fig.4. Confusion matrix of Logistic Regression in Balanced Single category model

Each one of them has multiple hyper parameters that must be tuned. The researcher has followed the subsequent methodology when defining the most effective set of hyper parameters for every model. Firstly, the researcher has decided which hyper parameters, wish to tune for every model, taking under consideration those which will have more influence within the model behaviour, and considering that a high number of parameters which require lots of computational time.

Then, researcher has defined a grid of possible values and performed a Randomized Search using 3-Fold Cross Validation (with 50 iterations). Finally, once researcher has get the model with the most effective hyper parameters, performed a Grid Search using 3-Fold Cross Validation cantered in those values so as to exhaustively search within the hyper parameter space for the most effective performing combination. The researcher has followed this system because with the randomized search able to cover a way wider range of values for every hyper-parameter without incurring in really high execution time. Once researcher has narrowed down the range for everyone, where to concentrate on search and explicitly specify every combination of settings to do.

The reason after electing K = 3 as the number of folds and 50 repetitions in the randomized search originated from the trade-off between smaller performance time and testing a high number of mixtures. When electing the superlative model in the process, the researcher has elected the accuracy as the evaluation metric. To check the performance on test sets that is totally unseen data, the researcher need to train the model with hyper parameters tuning process including cross validation and fitting the model on train sets. In WPC problems, there are several performance metrics which will be accustomed to gain insights on how the model is performed. The researcher has classified the datasets into balanced and unbalanced with same category (news) or multi-category documents (Asiannet, Astrology, Business, Crime, Health, Sports and Spiritual). There is a summary of the various models and their evaluation metrics on multi categories depicted in [Table 8] and [Table 9] with confusion matrix values as [Fig.3] and [Fig.4].

5. Conclusions

Now, the conclusion of this research paper denotes that the influence of stemmer is more effective on web page categorization on Gujarati language. The few classification reports of above selected supervised models are mentioned above with confusion matrix. These same techniques can apply to other languages using mapping of the language because most of the Indian languages are derived from the Indo-Aryan. The stemmer can implement using unsupervised model through machine learning. The researcher didn’t find any limitation of this proposed work but supposed to limited with features then to enhance this proposed techniques basis on positive feedback. The report of classification indicates Accuracy 75% to 97% for the WPC in Gujarati language web page. The confusion matrix indicates the error related to classification because stemming operation provides opportunities to achieve target output.

References

[1] C. D. Patel and J. M. Patel, “GUJSTER: a Rule based stemmer using Dictionary Approach,” *IEEE - International Conference on Inventive Communication and Computational Technologies (ICICCT 2017) GUJSTER:*, no. IEEE, pp. 496–499, 2017.

[2] D. Harman, “How effective is suffixing?,” *Journal of the American Society for Information Science*, vol. 42, no. 1, pp. 7–15, 1991.

[3] P. Majumder, M. Mitra, S. Parui, and G. Kole, “YASS: Yet Another Suffix Stripper,” *ACM Transaction of Information*

Systems, vol. 25. pp. 18–37, 2007.

[4] W. B. Frakes and R. Baeza-yates, *Information Retrieval : Data Structures & Algorithms*. 2004.

[5] W. B. Croft and J. Xu, “Corpus-specific stemming using word form co-occurence,” pp. 147–159, 1995.

[6] J. Anjali Ganesh, “A Comparative Study of Stemming Algorithms,” *IJCTA*, vol. 2, no. 2004, pp. 1930–1938, 2011.

[7] Neha Garg, R.K. Gupta,"Exploration of Various Clustering Algorithms for Text Mining", International Journal of Education and Management Engineering, Vol.8, No.4, pp.10-18, 2018.

[8] C. D. Patel and J. M. Patel, “Improving a Lightweight Stemmer for Gujarati Language,” *International Journal of Information Sciences and Techniques*, vol. 6, no. 1, pp. 135–142, 2016.

[9] C. D. Patel and J. M. Patel, “A Review of Indian and Non-Indian Stemming : A focus on Gujarati Stemming Algorithms,” *International Journal of Advanced Research*, vol. 3, no. 12, pp. 1701–1706, 2015.

[10] V. Gupta and G. S. Lehal, “A survey of common stemming techniques and existing stemmers for indian languages,” *Journal of Emerging Technologies in Web Intelligence*, vol. 5, no. 2, pp. 157–161, 2013.

[11] J. B. Lovins, “Development of a stemming algorithm,” *Mechanical Translation and Computational Linguistics*, vol. 11, no. June, pp. 22–31, 1968.

[12] J. Leskovec, A. Rajaraman, and J. D. Ullman, *Mining of Massive Datasets*, Edition-. Cambridge: Cambridge University Press, 2014.

[13] J. Han, M. Kamber, and J. Pei, *Data Mining Concept and Techniques*, 3rd Editio. Morgan Kaufmann Publishers is an imprint of Elsevier, 2012.

[14] E. Riloff, “Little words can make a big difference for text classification,” *SIGIR Forum (ACM Special Interest Group on Information Retrieval)*, pp. 130–136, 1995.

[15] M. Spitters, “Comparing feature sets for learning text categorization,” *Proceedings of the Sixth Conference on Content-Based Multimedia Access (RIAO 2002)*, pp. 1124–1135, 2000.

[16] T. Gaustad and G. Bouma, “Accurate Stemming of Dutch for Text Classification,” *Computational Linguistics in the Netherlands 2001*, pp. 1–14, 2016.

[17] A. M. Cohen, J. Yang, and W. R. Hersh, “Retrieval of Biomedical Documents,” *Medical Informatics*, pp. 1–9, 2004.

[18] M. Panda, “Developing an Efficient Text Pre-Processing Method with Sparse Generative Naive Bayes for Text Mining,” *International Journal of Modern Education and Computer Science*, vol. 10, no. 9, pp. 11–19, 2018.

[19] M. A. Hafer and S. F. Weiss, “Word Segmentation by Letter Successor Varieties,” *Information Storage and Retrieval*, vol. 10, pp. 371–385, 1974.

[20] M. Porter, “Snowball: A language for stemming algorithms,” *http://snowball.tartarus.org/texts/introduction.html*, pp. 1–15, 2001.

[21] M. F. Porter, “An algorithm for suffix stripping,” *Program*, vol. 14, no. 3, pp. 130–137, 1980.

[22] H. Donna, “How effective is suffixing?,” *Journal of the American Society for Information Science*, vol. 42, no. 1, pp. 7–15, 1991.

[23] J. L. Dawson, “Suffix Removal and Word conflaction,” *LLC Buletin*, vol. 2, no. 3, pp. 33–46, 1974.

[24] C. D. Paice, “An Evaluation Method for Stemming Algorithms,” *In Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 42–50, 1994.

[25] R. Krovetz, “Viewing Morphology as an Inference Process,” *16th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 191–202, 1993.

[26] J. Allan and G. Kumaran, “Stemming in the language modeling framework,” *CIIR Technical Report*, vol. IR-289, no. June, p. 455, 2003.

[27] N. L. Bhamidipati and S. K. Pal, “Stemming via distribution-based word segregation for classification and retrieval,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 37, no. 2, pp. 350–360, 2007.

[28] V. Gurusamy and S. K. K. Nandhini, “Performance Analysis : Stemming Algorithm for the English Language,” *IJSRD - International Journal for Scientific Research & Development*, vol. 5, no. 05, pp. 1933–1938, 2017.

[29] A. Ramanathan and D. D. Rao, “A Lightweight Stemmer for Hindi,” *Proceedings of the EACL 2003 Workshop on Computational Linguistics for South Asian Languages*, pp. 43–48, 2003.

[30] L. S. Larkey, M. E. Connell, and N. Abduljaleel, “Hindi CLIR in thirty days,” *ACM Transactions on Asian Language Information Processing*, vol. 2, no. 2, pp. 130–142, 2003.

[31] A. K. Pandey and T. J. Siddiqui, “An unsupervised hindi stemmer with heuristic improvements,” *Proceedings of SIGIR 2008 Workshop on Analytics for Noisy Unstructured Text Data, AND’08*, pp. 99–105, 2008.

[32] N. Aswani and R. Gaizauskas, “Developing Morphological Analysers for South Asian Languages: Experimenting with the Hindi and Gujarati Languages.,” pp. 811–815, 2010.

[33] U. Mishra and C. Prakash, “MAULIK: An Effective Stemmer for Hindi Language,” *International Journal on Computer Science and Engineering*, vol. 4, no. 5, pp. 711–717, 2012.

[34] A. Jain and S. Das, “Hindi stemmer @ fire-2013,” *ACM International Conference Proceeding Series*, pp. 4–6, 2013.

[35] V. Gupta, “Hindi Rule Based Stemmer for Nouns,” *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 4, no. 1, pp. 1–4, 2014.

[36] R. Joon and A. Singhal, “Analysis of MWES in Hindi Text Using NLTK,” *International Journal on Natural Language Computing*, vol. 6, no. 1, pp. 13–22, 2017.

[37] P. Patel, K. Popat, and P. Bhattacharyya, “Hybrid Stemmer for Gujarati,” *Proceedings of the 1st Workshop on South and Southeast Asian Natural Language Processing (WSSANLP),the 23rd International Conference on Computational Linguistics (COLING), Beijing*, no. August, pp. 51–55, 2010.

[38] K. Suba, D. Jiandani, and P. Bhattacharyya, “Hybrid Inflectional Stemmer and Rule-based Derivational Stemmer for Gujarati,” *Proceedings of the 2nd Workshop on South Southeast Asian Natural Language Processing (WSSANLP)*, pp. 1–8, 2011.

[39] J. Ameta, N. Joshi, and I. Mathur, “A Lightweight Stemmer for Gujarati,” *In Proceedings of 46th Annual National Convention of Computer Society of India.*, 2011.

[40] J. R. Sheth and B. C. Patel, “Stemming Techniques and Naïve Approach for Gujarati Stemmer,” *International Conference in Recent Trends in Information Technology and Computer Science (ICRTITCS - 2012) Proceedings published in International Journal of Computer Applications*, pp. 9–11, 2012.

[41] J. Sheth and B. Patel, “Dhiya: A stemmer for morphological level analysis of Gujarati language,” *Proceedings of the 2014 International Conference on Issues and Challenges in Intelligent Computing Techniques, ICICT 2014*, pp. 151–154, 2014.

[42] B. Dalwadi and N. Desai, “An Affix Removal Stemmer for Gujarati Text,” *IEEE*, pp. 2296–2299, 2016.

[43] M. Patel, “A Suffix Stripper for Gujarati Noun,” *Discovery*, vol. 47, no. 218, pp. 95–101, 2015.

[44] Sharvari S. Govilkar, J. W. Bakal, Sagar R. Kulkarni,"Extraction of Root Words using Morphological Analyzer for Devanagari Script", International Journal of Information Technology and Computer Science, Vol.8, No.1, pp.33-39, 2016.

[45] X. Qi and B. D. Davison, “Web page classification: Features and algorithms,” *ACM Computing Surveys*, vol. 41, no. 2, p. 31, 2009.

[46] P. Bhati, *Hand book of Gujarati Grammer*. 1889.

[47] G. of Gujarat, ભાષા વિવેક, Second. Gandhinagar: ભાષા નિયામકની કચેરી, 2010.

[48] Mavajibhai, વ્યાકરણ પરિચય. 2010.

[49] G. Grefenstette and P. Tapanainen, “What is a Word, What is sentence? Problems of Tokenization,” *Rank Xerox Research Centre*, vol. 3, p. 9, 1994.

[50] L. H. Patil and M. Atique, “A novel approach for feature selection method TF-IDF in document clustering,” *2013 3rd IEEE International Advance Computing Conference (IACC)*, pp. 858–862, 2013.

[51] M. A. Hafer and S. F. Weiss, “Word segmentation by letter successor varieties,” *Information Storage and Retrieval*, vol. 10, no. 11–12, pp. 371–385, 1974.

[52] L. Hao and L. Hao, “Automatic identification of stop words in chinese text classification,” *Proceedings - International Conference on Computer Science and Software Engineering, CSSE 2008*, vol. 1, pp. 718–722, 2008.

[53] T. K. H. T. K. Ho, “Fast identification of stop words for font learning and keyword\nspotting,” *Proceedings of the Fifth International Conference on Document Analysis and Recognition. ICDAR ’99 (Cat. No.PR00318)*, 1999.

[54] Pareek, Jyoti and J. H. Pareek Jyoti, “Evaluation of some Information Retrieval models for Gujarati Ad hoc Monolingual Tasks,” *VNSGU Journal of Science & Technology*, vol. 3, no. 2, pp. 176–181, 2012.

Authors’ Profiles

Chandrakant D. Patel was born on May 19, 1981.He received the BCA and MCA degree from Hemchandracharya North Gujarat University, Patan in 2002 and 2005, respectively. He having 14+ years’ experience in Academic with UG and PG Courses. Prior to joining Acharya Motibhai Patel Institute of Computer Studies (PG Department), he worked with Shri C. J. Patel College of Computer Studies (UG Department), Visnagar from 2005 to 2011. His areas of interest include Web Page Categorization and Stemmer in Indian languages. He has also written a book for Hemchandracharya North Gujrat University BCA students titled “Computer Organization” in 2011. He has completed OP sponsored by UGC at ASC, Rajkot. He has 2 national and 6 international papers in his credit

Dr. Jayeshkumar M. Patel, having rich experience of 16 Years in Academics, Industry, Research and International exposure, is holding Doctorate in ERP (Computer Science) from North Gujarat University. Rewarding his research work, he has been awarded “Career Award For Young Teachers” from AICTE, Delhi. He is working as a recognised Ph.D. guide at G.T.U., H. N. G.U., Ganpat University and also with many other reputed universities. He has good number of research under his name and presented more than 47 research papers in International and National Journals and Conferences. He has delivered number of expert talk in SANDHAN Programme and UGC Sponsored Programme. He is also the member of board of studies and selection committee of different universities.

How to cite this paper: Chandrakant D. Patel, Jayesh M. Patel, "Influence of GUJarati STEmmeR in Supervised Learning of Web Page Categorization", International Journal of Intelligent Systems and Applications(IJISA), Vol.13, No.3, pp.23-34, 2021. DOI: 10.5815/ijisa.2021.03.03