Journal of King Saud University – Computer and Information Sciences 31 (2019) 105–112

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

Syntactic parsing and supervised analysis of Sindhi text

Mazhar Ali Dootio [a,b,*], Asim Imdad Wagan [c]

[a] Shaheed Zulifqar Ali Bhutto Institute of Science and Technology, Karachi, Pakistan
[b] Benazir Bhutto Shaheed University Lyari, Karachi, Pakistan
[c] Mohammad Ali Jinnah University, Karachi, Pakistan

ARTICLE INFO

Article history:
Received 28 May 2017
Revised 12 September 2017
Accepted 19 October 2017
Available online 24 October 2017

Keywords:
Sindhi parser
Sindhi WordNet
NLP
Tokenization
Machine learning
Supervised model

ABSTRACT

This research study addresses the morphological and syntactic problems of Sindhi language text by proposing an Algorithm for tokenization and syntactic parsing. A Sindhi parser is developed on basis of proposed algorithm to perform syntactic parsing on Sindhi text using Sindhi WordNet (SWN) and corpus. Results of Sindhi syntactic parsing are accumulated to develop multi-class and multi-feature based Sindhi dataset in CSV format. Three attributes of Sindhi dataset are labelled as class. All three classes are comprised with different number of categories. SVM, Random forest and K-NN supervised machine learning methods are used and trained to analyze and evaluate the Sindhi dataset. 80% of dataset is used as training set and 20% of dataset is used as test set. In this research study, 10-fold cross validation technique is applied to evaluate and validate the supervised machine learning process. The SVM classifier gives better results on class phrase and UPOS whereas Random forest gives better result on class TagStatus. Precision, recall, f-measure and confusion matrix approve the performance of all supervised classifiers. The better performance of supervised machine learning methods, support the Sindhi dataset and Sindhi online parser for future research. This study opens new doors for research on right hand written languages especially Sindhi language to solve its computational linguistics problems.

1. Introduction

The structure of right hand and left hand written languages is different from each other (Naz et al., 2013) in recognizing the syntactic tokens. Sindhi language is right hand written language and one of the oldest and morphologically rich languages of the World (Rahman, 2009), having 52 letters (Fig. 1) while English, Urdu and Arabic languages have less number of alphabetic letters. The large number of alphabetic letters shows large domain of Sindhi language having huge number of own lexicons as well as adopted lexicons from other languages like English, Arabic and Persian. This study presents novel problems and issues of Sindhi language to solve with new methodology and finally, present work to Natural languages processing system for future research.

The Grammar of Sindhi language is different and unique (Bag and vyakaran, 2015) from the grammar of other languages. The noun gender is different than the rest of languages. There are two types of gender in Sindhi language: One is masculine and second is feminine. Diacritics and adjectives make the genders in Sindhi language. This language uses intransitive passive voice verb. The presented Sindhi text ڀينگهي ۾ لڏجي ٿو (Peenghay me ludjay tho) is passive voice of intransitive verb. The active voice of this sentence is آءُ ڀينگهي ۾ لڏان ٿو (Aaun peenghay me luddaan tho) in Engish (I swing on Hammock). It is property of Sindhi language that it possesses the intransitive passive voice. This makes Sindhi language unique from the rest of languages of the world.

Most of the research studies concentrate on English text POS tagging that may be with original or universal POS tag sets. A variety of reliable resources are available for English Language text to tag Universal POS tag set, therefore, it is most facilitated language of the world. However, the on-line resources for languages other than English like Sindhi, are limited even in this digital era. To work on Sindhi syntactic parsing and UPOS tagging is not the same as to work on the English language. This difference creates research

* Corresponding author at: Benazir Bhutto Shaheed University Lyari, Karachi, Pakistan.
E-mail addresses: mazharaliabro@gmail.com (M.A. Dootio), aiwagan@gmail.com (A.I. Wagan).
Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2017.10.004
1319-1578/

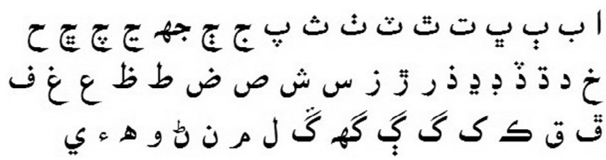

Fig. 1. Sindhi Alphabet in form of Persian-Arabic script.

question for working on Sindhi text. As there is no availability of Sindhi Corpus and Sindhi WordNet in proper way, therefore, it is a challenging and main task of this study to develop Sindhi parser on basis of self proposed algorithms to generate Sindhi tokens, to tag UPOS to Sindhi tokens, to parse the text syntactically and finally analyze statistically to solve the Sindhi linguistics problems.

Right hand written languages keep separate alphabetical, morphological and grammatical structure. Morphology of Sindhi language makes this language as the rich language. It is observed that single word of Sindhi language is showing multiple meaning and some times bunch of words show single meaning. Inflection, affix and suffix change the structure and meaning of tokens in Sindhi language. This study addresses the problems of Sindhi language by developing a Sindhi parsing tool to recognize and parse the text and finally presents the morphological statistics of the tagged Sindhi text.The developed Sindhi parser uses Persian-Arabic writing style of Sindhi language to parse the text. All forms of morphology are parsed syntactically. Sindhi parser uses Universal Part of Speech (UPOS) for mapping tags to Sindhi text. The evaluation and analysis of several tree banks demonstrates the importance of Universal tag set for different languages of the World as tag sets are language specific (Petrov et al., 2013). UPOS tags match to Sindhi POS tags with some variances like PART (Particle) tag. Sindhi language uses Adverb (zarf ظرف) for different activities including negation showing words while universal POS tag set uses PART for adverbial and possessive markers. However possessive marker (Harf-e-izafat حرف إضافت) in Sindhi text which belongs to Adposition tag (harf-e-jar حرف جر) may be used as PART universal tag.

2. Related work

Syntactic parsing analyzes the human text and solves computational linguistic problems. Ramteke et al. discussing the functionality of parser, defines that tokens of the sentence are very much important for the parsing because parser is analyzing the sequence of tokens ordered in the sentence to define the grammatical structure. Natural Languages processing is generally a system that is involved in segmentation, morphological analysis, lexical handling and syntactic analysis. The parsing is continuous process and seldom finishes in itself, it extracts the information concerning linguistic format of sentence for the advantage of applications of machine translation, information extraction and etc. (Nivre, 2015). According to Tsarfaty et al. (2013) parsing of text is significant as it expose the grammatical parts of sentences. The diacritics make the tokens clear and meaningful as well as finishes morphological and lexical ambiguities. At the same time diacritics change the grammatical structure of the tokens. Mahar J. describes that it is difficult job to segment the text computationally and analysis syntactically (Mahar and Memon, 2010) as Sindhi text is holding complex and compound words. The diacritics make the language strong and rich as well as creates problems for the syntactic parser. Defining this problem Shahrour et al. (2015) discuss that diacritics is tough process for Arabic automatic process due to morphology and number of causes and reasons. Viewing the previous work, done on syntactic parsing, a Sindhi parser is developed to solve the problems of Sindhi language.

3. Material and methods

This research study is empirical in nature, therefore, a Sindhi parser is developed which uses the Sindhi word processor to insert the Sindhi text. The jobs of parser are:

- To reverses the text from left side to right side as Sindhi is right hand written language,
- To generate tokens from text,
- Syntactically parse text in shape of extending tree and
- To show the statistical and morphological results of parsed data.

Results of Sindhi parser are tested and accumulated to develop a Sindhi dataset that machine learning supervised processed may be done. The purpose of machine learning process is to evaluate and analyze the dataset features and labelled classes.

3.1. Syntactic parsing of Sindhi text

Grammar of any language performs important role in making proper sentence. Proper utilization of words make the notion of sentence. A word is basic unit of the sentence, therefore, it is very much important to understand words of sentence and their grammatical and morphological structure. Morphemes are basic units of morphology by which words can be identified. For example "I eat", "She eats" are two sentences. First word "eat" is verb and simple word while second word "eats" is not simple word but complex word with addition of suffix "s". To tag and parse the word syntactically, understanding of grammar and morphology of concerned language is necessary. Syntactic parsing of the text is a method of segmentation and identifying the diverse types of phrases from the presented text. The syntactic parsing of the sentences of Sindhi text works hierarchically on basis of FIFO data structure. All the tokens are dependent on each other. In this concern, each word that comes first, is mapped with phrase and universal part of speech tag. If token is unknown then parser maps it with letter

Table 1
Features, missing values and Description of Sindhi Lexicon.

Feature Name	Missing Values	Complete Name and description
incdeclevel	0	This feature shows that either lexicon is Incremental or Decremental level
genm	0	Gender Masculine
genf	0	Gender Feminine
singular	0	Singular form
plural	0	Plural form
Vsecform	0	Verb's second form
pospol	0	Positive Polarity
negpol	0	Negative Polarity
Neutral	0	Neutral Polarity
primform	0	Primary Form
secform 4	0	Secondary Form
complexword	0	Complex word
commpoundword	0	Compound word
Reduplicatedword	0	Reduplicated word
UnigramProb	0	Uni-gram Probability
Hypernym	0	Hypernym
Hyponym	0	Hyponym
Diac	0	Diacritical words
stem	0	Stemming words.
Infinitive	0	Infinitive words
stoken	0	Sindhi token

"X" that is UPOS tag. Algorithm (1) is proposed to process the Sindhi text for syntactical parsing.

Algorithm 1: Syntactic parsing of Sindhi Text.

```
Data: Sindhi Text
Result: Syntactic Analysis of Sindhi Text
initialization;
read current Sindhi text;
Call Function ReverseArrayString();
while not at end of this document do
    Read token from SWN ;
    (Sindhi WordNet provides lexicons with proper POS
     tagging, synsets and meaning)
    Call function WordTokenization();
    if understand then
        Initialize P as a Parsing Tree with root "S";
        for Each Word =1 to N do
            Label Phrase / UPOS To Sindhi Token;
            else if Not understand then
                Tag X To Sindhi Token;
                repeat
                    break when End==0
                until 1 to N;
                ;
            end
        end
        Call function StatisticalAnalysis( )
    else
        go back to the beginning of current section;
    end
end
```

On basis of proposed algorithm (1), an online NLP tool named Sindhi parser (http://www.sindhinlp.com/sindhi_parser.php) is developed which tokenizes, tags, parses and statistically analyze the Sindhi text.

3.2. Sindhi dataset analysis

The Sindhi Data set is multi-class based categorical dataset. It has 25 attributes including class attributes. The number of records is 6841 in presenting dataset. Each token is presented with its features, tagging status and UPOS mapping. The class attributes of this dataset are Phrase, UPOS and TagStatus. Class Phrase has 8 sub classes, class UPOS has 18 sub classes and Class TagStatus has two sub classes. The correspondence of all features and classes to Sindhi lexicon is according to usage of Sindhi lexicon. The status of all features except UPOS and Uni-gram Probability is shown in related attributes in form of binary numbers. The 0 shows false correspondence of feature to Sindhi lexicon whereas 1 shows true correspondence to Sindhi lexicon. Therefore, the range of these attributes is from 0 to 1. Each instance shows UPOS article, tagging status, probability of uni-gram tokens and features of Sindhi token. The data set is pre-processed properly to analyze the missing values and ambiguities. No missing value and ambiguities are found in the dataset. Table 1 shows the attributes, number of missing values and Complete Name.

All the features which are presented in dataset of Sindhi text are significant for the analysis of Sindhi lexicon. The labelled classes are not included in the features list. The features show the actual and grammatical status of Sindhi tokens which are very much significant for NLP. Uni-gram probability is a statistical analysis of language model which is a probability distribution over sequences of words in text. The Uni-gram probability is measured to show the contribution of Sindhi tokens in the dataset. In this concern, the frequency of current token is observed in whole dataset. The statistical model is applied to measure the frequency of current token.

Supervised machine learning methods SVMs, Random forest and K-NN are utilized for evaluation and comparative analysis of Sindhi dataset. The target classes are categorized into sub-classes properly for machine learning process.

Random Forest (RF) is mixture of tree predictors and depends on the value of a random vector sampled independently and with the same distribution for all trees in the forest (Breiman, 2001). RF method trains the decision trees and gives the results of over all trees in ensemble, therefore, Random Forest is significant machine learning method that gives very good results (Singh et al., 2016). The disadvantage of Random forest method starts when number of trees increase.

Sindhi dataset is multi-feature and multi-class therefore, Several decision trees are generated randomly to built Random forest for purpose of classification of dataset. Each tree of forest is generated with dissimilar bootstrap sample taken from the dataset. Sindhi dataset is Unicode-8 based dataset, thus Random forest is trained with 80% of Sindhi training set at first. The training set is consisted of several features which are classified on basis of labelled classes. Variable collection for each split in the decision tree is arranged on basis of randomly selected subset of features in place of full feature-set. The test data set is analyzed by using the rule of each randomly generated decision tree. RF calculates the votes given by decision trees and describes prediction accordingly. The results and performance of RF are presented through confusion matrix, accuracy, precision, recall and f-score.

Support Vector Machine is supervised machine learning method for classification and regression. It is very good method for its generalization performance. It works on constructing hyperplane or set of hyperplanes for analysis of target classes. Data items which are marginalized by hyperplane and lying near the margin boundaries are called support vectors (Meyer and Wien). Non-Linear SVM transfers the data into high dimensional space. The role and performance of SVM kernel methods make the SVM robust and important machine learning method. The application of SVM is good to text classification, image processing, segmentation, multi-dimensional data and etc. According to Das et al. (2015) SVM performs its role in NLP text categorization and gives very good accuracy. Outahajala et al. (2013) recognizing the efficiency of CRF and SVM on POS Tagging describes the very good performance of SVM on subject of tokenization. SVMs are complex and take more time for training.

The SVM is selected to work on Sindhi dataset as this dataset is multi-dimensional and multi-class. The classifier generates the multiple hyper-planes to evaluate the classes of dataset. SVM RBF kernel is used to generate multi hyper-planes to divide the dataset according to classes and dimensions. SVM model is trained on basis of RBF kernel because this kernel takes decision automatically to detect the non-linear dimensions and classes of dataset. RBF kernel worked better and more accurate on Sindhi text dataset, therefore, results are acquired with better accuracy, confusion matrices, precision,recall and f-score.

K-Nearest Neighbor (K-NN) is also called distance based, instance based, lazy and memory based learning method. It is a supervised machine learning methods that works for classification. It stores all training instances and perform classification by allocating target function to its new instance. K-NN works on its nearest data values called nearest neighbors therefore, the value of K is very important in this regard. The value of K defines the number

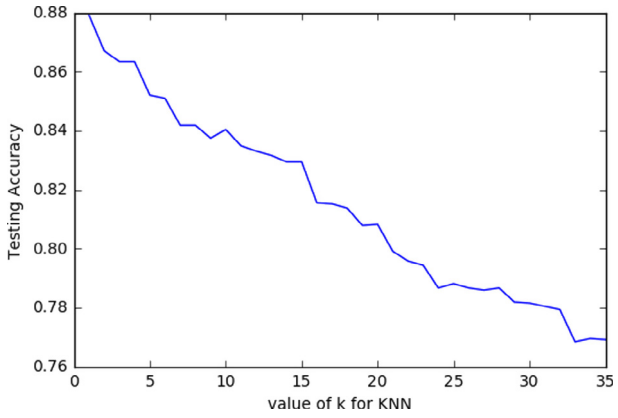

Fig. 2. Value of K for K-NN machine learning method.

of nearest neighbors to access therefore, distance between K point and targeted neighbors is calculated by distance functions like Euclidean distance function. This function measures the distance of nearest neighbors to K. It is suitable for multi-model classes (Singh et al., 2016). In this study, K is assigned value 3 whereas, performance of K is evaluated from 0 to 35. The value is assigned on the basis of testing accuracy of K-NN. Fig. (2) shows the best performance of K for K-NN using Sindhi data set.

The purpose of selecting these supervised machine learning methods is to present broad view of these methods through their performance and efficiency on non-English data set. The applications and targets of every method are found almost different from each other on this data set. The working efficiency and performance of all nominated supervised algorithms are better with medium or little difference of accuracy, precision, recall and f-measure scores.

4. Results and discussions

This research study is designed to parse the Sindhi text syntactically. Different types of sentences including small, medium and large are processed in the developed Sindhi parser. Results are obtained in form of word tokenization, Tagging and syntactic parsing.

4.1. Sindhi word tokenization

Word tokenization is basically segmentation of Sindhi text which gives results with better separate and meaningful tokens. Sindhi parser splits the text into separate tokens and assigns them sequence numbers. Fig. 3 shows result of Sindhi text word tokenization with proper labeling of sequence numbers assigned to each token. Sequence numbers in word tokenization show the order and dependency of tokens in the sentence. The sequence of tokens of the presented sentence starts from Sindhi proper noun سنڌ (Sindh) with sequence number 1 (one) and ends at Sindhi verb ٿا (Thaa) with sequence number 26 (Twenty-six) and sentence is ended with period which is at sequence number 27.

4.2. Syntactic parsing of Sindhi text

The Sindhi parser is tested with different sentences of Sindhi text. The parser for syntactic parsing works properly, it parses the Sindhi text including complex, compound and reduplicated Sindhi words syntactically and gives better results. Fig. 4 shows the syntactic parsing of Sindhi text. Each token of the sentence is tagged with proper labeling of phrase and universal POS tag. The

سنڌ، 1 ،سنڌودريا، 2 ،جي، 3 ،ٻنهي، 4 ،ڪپن، 5 ،تي، 6 ،تي، 7 ،آباد، 8 ،
آهي، 9 ،ء، 10 ،پاڪستان،11 ،جو، 12 ،صوبو، 13 ،آهي، ،،، 14 ،هتان،
15 ،جا، 16 ،ماڻهو،17 ،ايندڙ، 18 ،هر، 19 ،ماڻهو، 20 ،کي، 21 ،دل، 22 ،
سان، 23 ،پليڪار، 24 ،چون،25 ،ٿا، 26 ،.، 27

Fig. 3. Sindhi text word Tokenization.

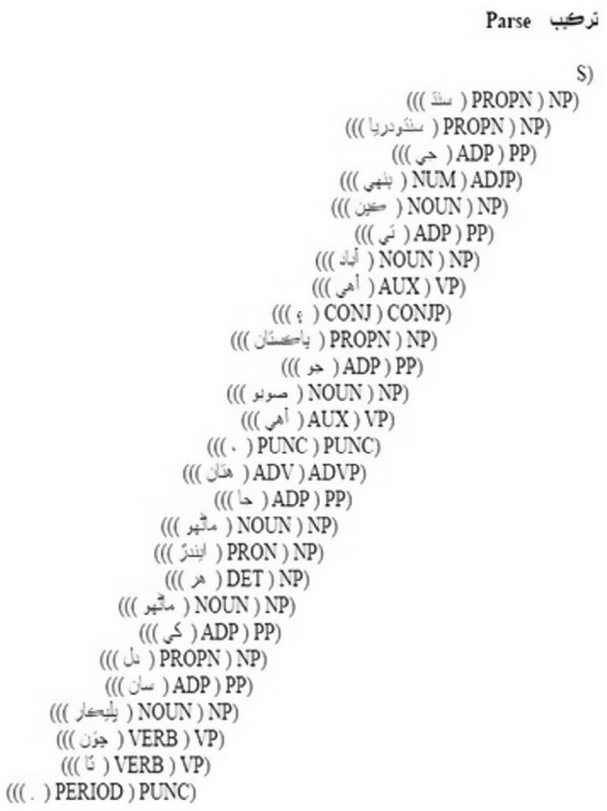

Fig. 4. Syntactic Parsing of Sindhi Text.

proposed algorithm for syntactic parsing identifies tokens correctly and maps the phrase and UPOS tags to all identified tokens. The parsing tree extends hierarchically in shape of extending tree.

The statistical analysis is important and significant part of online Sindhi parser. It analyzes the tokens morphologically and grammatically. Morphological analyzer shows the number of morphological forms which corresponds to Sindhi tokens whereas UPOS analysis shows the number of UPOS articles which are mapped to Sindhi tokens. This analysis shows the richness and complexity of language. Table 2 shows the morphological analysis of syntactically parsed Sindhi text. The morphological analysis shows different forms of Sindhi tokens. It shows good number of secondary or bound form of morphological words which shows richness of Sindhi language.

The use of Diacritics in Sindhi text change the meaning and understanding of words and sentences. There is an example sentence of Sindhi text that shows the words with diacritics and

Table 2
Statistical and Morphological analysis of Syntactically Analyzed Sindhi Text.

Total Number of Sindhi tokens in processed text: 27 Execution time using Intel i3 machine with 2 GB RAM: 0.05731 s		
Morphological Words	Total words	Percentage
Simple Words	15	55.56
Complex Words	7	25.93
Compound Words	3	11.11

without diacritics. (He was selling and purchasing goods in market and that woman, which is his wife, was helping him in his business). The Sindhi word هُو (He) with diacritic, is Pronoun in Sindhi grammar and used for some one who is masculine gender. The first portion of sentence ends with auxiliary verb هو (was) without diacritics which shows the action happened in past. An other Sindhi word هُوءَ (She) is secondary morphological form of simple morphological form هو which is used with diacritic here for feminine gender. The Fig. 5 shows syntactic parsing of Sindhi text with and without diacritics.

4.2.1. Result analysis

The Sindhi parser maps phrases and UPOS to tokens accurately and parses syntactically. To approve the accuracy and performance of parser, the generated Sindhi data set is assessed and analyzed by supervised machine learning methods: SVM non-linear, Random forest and K-NN. These machine learning methods perform better on the multi-class dataset. The dataset is divided into 80% training set and 20% test set. The training set is used to train the classification model and test set is used for proper evaluation of model. The cross validation technique is used with 10-folds to validate the training and assess the test set for proper prediction. The process of cross validation continues till 10 folds to analyze and validate each randomly partitioned part of Sindhi dataset for training and test sets. Finally, it counts the error rate. The cross validation technique confirms the noteworthy classification results which are obtained through supervised machine learning methods. Machine learning process classifies the true and false annotated Sindhi data and recognizes the original features of Sindhi text.

The confusion matrices of Sindhi dataset are derived through applied machine learning methods. The columns of each matrix show the predicted data and rows show the true data. The diagonal components of matrix describes the number of data values which show the true and predicted labels. The increased number of true values shows the high number of predicted values which presents the better performance of machine learning classifier. Confusion matrices evaluate the quality of the output of a classifiers on Sindhi data set labeling all three classes: Phrase, UPOS and TagStaus.

Fig. 6 shows the confusion matrix which evaluates the quality of the output of a SVM non linear classifier on Sindhi data set labeling class Phrase. The class Phrase is multi-labelled class which shows the status of phrases tagged to Sindhi text. The matrix gives high number of positive values, therefore, the results of SVM non-linear are better than the random forest and K-NN classifiers. False values are observed in Interjection phrase (INTJP).

Fig. 7 shows the confusion matrix which assesses the output of a Random forest classifier on Sindhi data set labeling class Phrase. The matrix shows good number of positive values whereas, false values are also observed in Adverbial phrase (ADVP), Propositional phrase (PP) and Interjection phrase (INTJP).

Fig. 8 describes the confusion matrix which considers the output of a K-NN classifier on Sindhi data set labeling class Phrase. Matrix shows good number of positive or true values along with some false values which are observed in Adjective phrase (ADJP) and Interjection phrase (INTJP).

Fig. 9 describes the confusion matrix which evaluates the output of a SVM non linear classifier on Sindhi data set labeling class UPOS. Matrix shows better results with high number positive values nevertheless, some false values are found in punctuations and symbols.

Fig. 10 describes the confusion matrix which assesses the output of a Random forest classifier on Sindhi data set labeling class UPOS. Matrix describes high number of positive or true values whereas, false values are shown in Adverbial Tag (ADV).

S)
(((هُو) PRON) NP)
(((بازار) NOUN) NP)
(((۾) ADP) PP)
(((سامان) NOUN) NP)
(((جي) ADP) PP)
(((خريدوفروخت) NOUN) NP)
(((ڪندو) VERB) VP)
(((هو) AUX) VP)
(((۽) CONJ) CONJP)
(((هُوءَ) PRON) NP)
(((مائي) NOUN) NP)
(((جيڪا) CONJ) ConjP)
(((هن) PRON) NP)
(((جي) ADP) PP)
(((زال) NOUN) NP)
(((آهي) AUX) VP)
(((هنجو) PRON) NP)
(((ڏنڊي) NOUN) NP)
(((۾) ADP) PP)
(((ساڻ) VERB) VP)
(((ڏيندي) VERB) VP)
(((هئي) AUX) VP)

Fig. 5. Syntactic Parsing of Sindhi Text with and without diacritics.

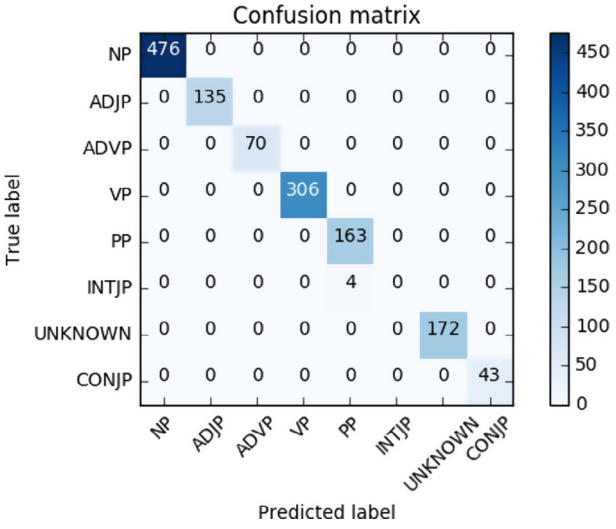

Fig. 6. SVM Non-linear Confusion Matrix on Sindhi data set labeling class Phrase.

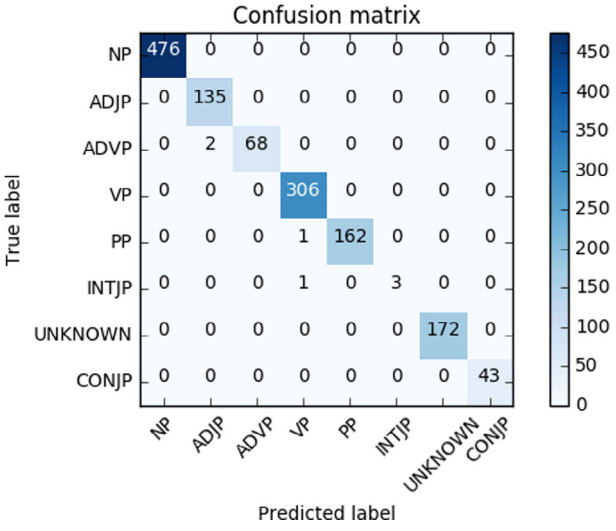

Fig. 7. RF Confusion Matrix on Sindhi data set labeling class Phrase.

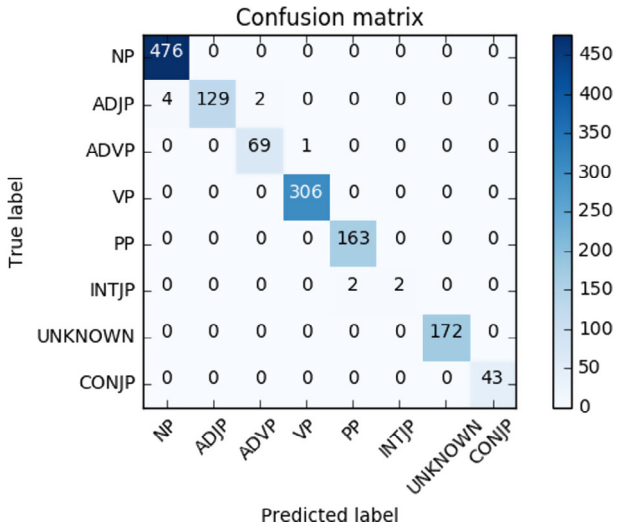

Fig. 8. K-NN Confusion Matrix on Sindhi data set labeling class Phrase.

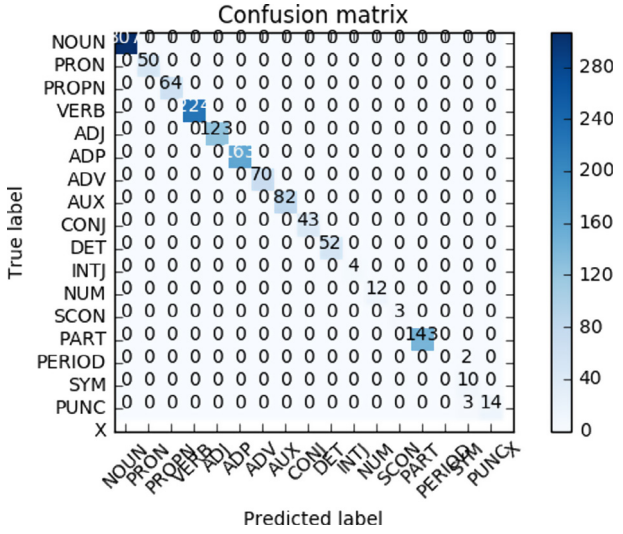

Fig. 9. SVM Non-linear Confusion Matrix on Sindhi data set labeling class UPOS.

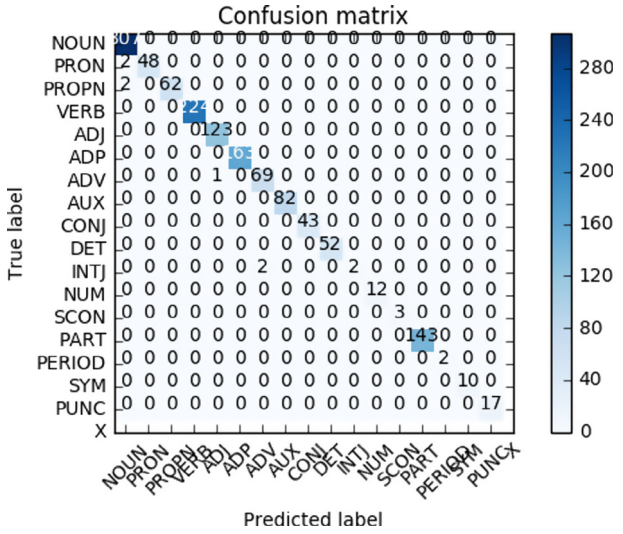

Fig. 10. Random Forest Confusion Matrix on Sindhi data set labeling class UPOS.

Fig. 11 describes the confusion matrix which assesses and evaluate the functionality of a K-NN classifier on Sindhi data set labeling class UPOS. Matrix shows better results with high number positive values. All the tags of UPOS are found accurate in mapping to Sindhi text.

TagStatus is target class of Sindhi dataset which shows the status of UPOS. If Sindhi lexicon is tagged properly with UPOS tag than it shows true value with number digit 1 else it shows false value with number digit 0. The data set is updated according to status of UPOS tagging to Sindhi dataset. Fig. 12 describes the confusion matrix which considers the output of a SVM non linear classifier on Sindhi data set labeling class TagStatus. Matrix shows high number true values. No false value is found in the matrix.

Fig. 13 describes the confusion matrix which evaluate the output of a Random Forest classifier on Sindhi dataset labeling class TagStatus. Matrix describes good number of true values, nevertheless, false values are found which shows the error rate of classification.

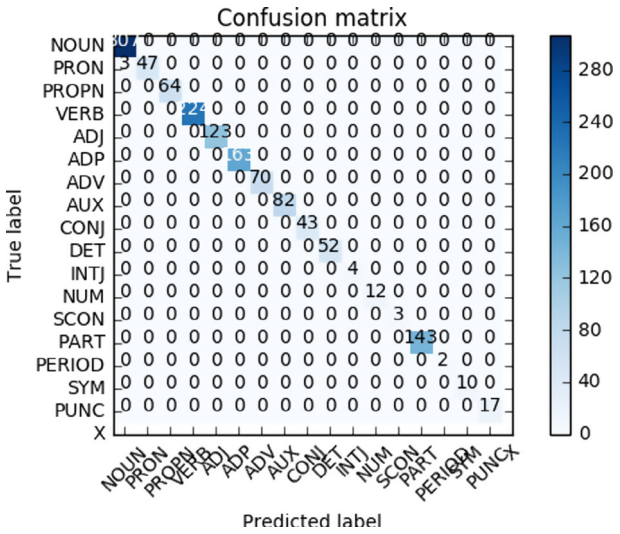

Fig. 11. K-NN Confusion Matrix on Sindhi data set labeling class UPOS.

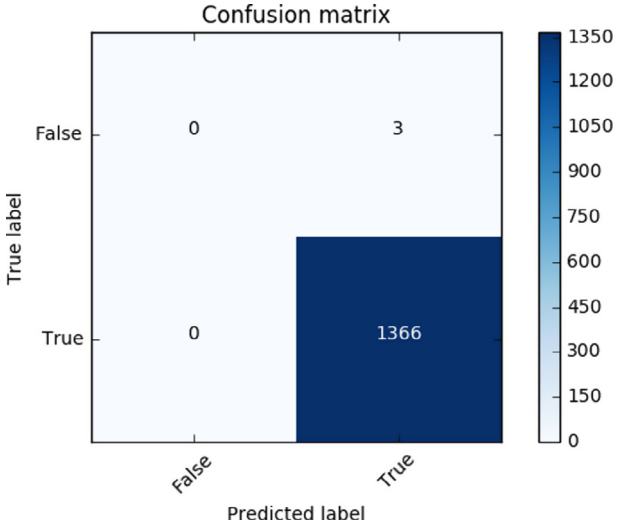

Fig. 12. SVM Non-linear Confusion Matrix on Sindhi data set labeling class TagStatus.

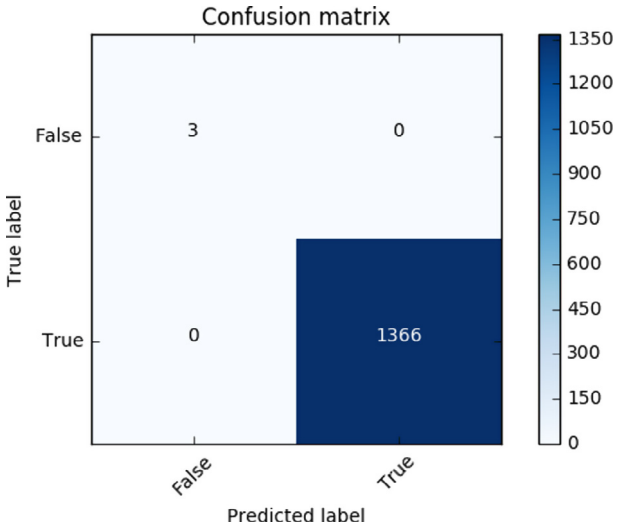

Fig. 13. Random Forest Confusion Matrix on Sindhi data set labeling class TagStatus.

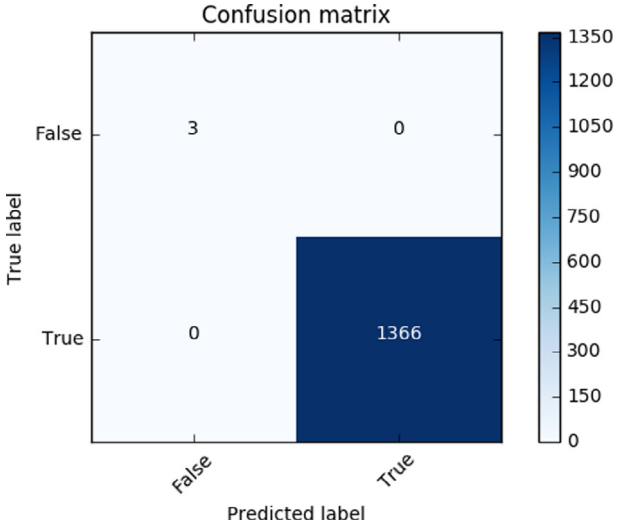

Fig. 14. K-NN Confusion Matrix on Sindhi data set labeling class TagStatus.

Fig. 14 describes the confusion matrix which assess the output of a K-NN classifier on Sindhi dataset labeling class TagStatus. Matrix shows good number of true values along with some false values.

The evaluation of accuracy rate is very much important for any supervised machine learning study. It recognizes the performance of machine learning classifiers, therefore, the accuracy of all supervised machine learning methods, approves the better results of confusion matrices. Table 3 shows the accuracy rate of supervised machine learning methods SVM non-linear, random forest and K-NN on class Phrase, class UPOS and class TagStatus.

The performance of SVM non-linear gives better results on phrase and UPOS classes of Sindhi dataset, whereas, random forest gives better results on class TagStatus which confirm the better performance of Sindhi parser. Fig. 15 shows comparative analysis of accuracy of all three classes.

The precision and Recall analysis is evaluation of relevant data retrieved from relevant and irrelevant data available in dataset basically. The F-measure is the single measure of precision and recall. Sindhi dataset is analyzed properly using machine learning

Table 3
Performance of supervised machine learning methods on Sindhi data set targeting class Phrase, UPOS and TagStatus.

Accuracy rate (in %) acquired through labelled classes			
Method	Class Phrase	Class UPOS	Class TagStatus
SVM-Nonlinear	99.92	99.74	99.96
K-NN	99.45	99.70	99.96
Random Forest	99.70	99.56	99.99

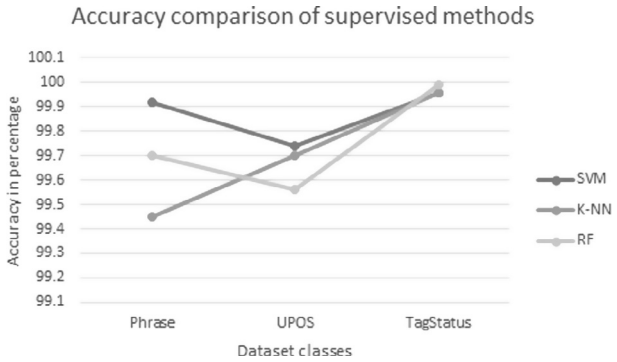

Fig. 15. Comparison of Accuracy rates produced by supervised classifiers through labelled classes.

Table 4
P.R.F results show performance of SVM Non-Linear on Sindhi dataset targeting labeled classes.

Measures	Class Phrase	Class UPOS	Class TagStatus
Precision	99	100	100
Recall	100	100	100
F-Score	100	100	100

Table 5
P.R.F results show performance of Random Forest on Sindhi dataset targeting labeled classes.

Measures	Class Phrase	Class UPOS	Class TagStatus
Precision	100	100	100
Recall	100	100	100
F-Score	100	100	100

Table 6
P.R.F results shows performance of K-NN on Sindhi dataset targeting labeled classes.

Measures	Class Phrase	Class UPOS	Class TagStatus
Precision	99	100	100
Recall	99	100	100
F-Score	99	100	100

supervised model, thus, precision and recall techniques are used to assess the Sindhi dataset. The precision rate has evaluated by true values retrieved from relevant data whereas recall has evaluated by true values retrieved from all data. The results show significant performance of supervised model. Tables 4–6 show class-wise precision, recall and f-score of true data which are retrieved from relevant and irrelevant data of Sindhi data set.

Fig. 16 shows precision, recall and f-score measures and their differences on assessment of Sindhi dataset. The high number of relevant values shows the better results of machine learning model.

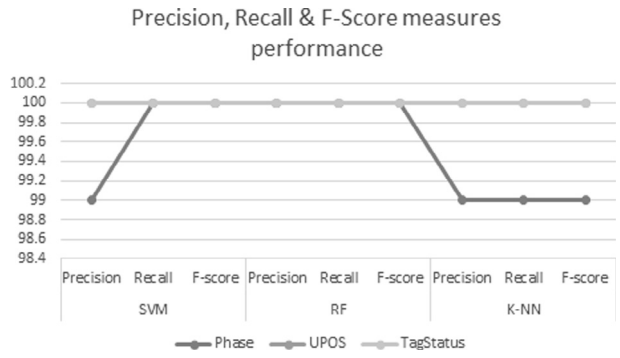

Fig. 16. Comparison of P.R.F through supervised classifiers on Sindhi dataset targeting all labeled classes.

5. Conclusion

The physical boundaries are not barriers in this digital era. Therefore, understanding and translation of languages are important in this regard. computational linguistics perform a vital role in the provision of a platform for understanding of dissimilar languages of the world to connect the people of different countries. This research study is planned to address the computational linguistic problems of Sindhi language. Viewing the above mentioned problems, a Sindhi parser (http://www.sindhinlp.com/) is developed that works on basis of proposed algorithms. The rule based system is followed to develop the Sindhi parser. This parser generates tokens from the Sindhi text, syntactically parse that text and finally presents the statistical and morphological analysis. The results of parser are accumulated to develop a dataset. The supervised machine learning methods SVM non-linear, random forest and K-NN are utilized for the comparative analysis and evaluation of dataset. The performance of supervised methods are acquired on basis of labeled classes. SVM non-linear gives better accuracy, precision, recall and f-measure results than the random forest and K-NN. The confusion matrices evaluate the performance of supervised classifiers on Sindhi dataset and show the better performance. These results show the better and acceptable performance of Sindhi parser.

This research work may be extended to work more on universal dependencies and syntactic ambiguities of Sindhi text using NLP tools and techniques.

Acknowledgment

Presented research study is produced from my research thesis "Mapping Universal POS Tag set to Sindhi Text" which is submitted at SZABIST Karachi Sindh Pakistan. I acknowledge the support of Program coordinator Dr. Hussnain Mansoor Ali Khan and department head Dr. Imran Amin in provision of resources and environment.

References

Bag, M.K., vyakaran, Sindhi., 2015. Sindhi Adabi Board Jamshoro Sindh Pakistan.
Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32. https://doi.org/10.1023/A:1010933404324.
Das, B.R., Sahoo, S., Panda, C.S., Patnaik, S., 2015. Part of speech tagging in odia using support vector machine. Procedia Comput. Sci. 48, 507–512. https://doi.org/10.1016/j.procs.2015.04.127.
Mahar, J.A., Memon, G.Q., 2010. Sindhi part of speech tagging system using wordnet. Int. J. Comput. Theory Eng. 2 (4), 538.
Meyer, D., Wien, F.T., Support vector machines. The Interface to libsvm in package e1071.
Naz, S., Hayat, K., Razzak, M.I., Anwar, M.W., Akbar, H., 2013. Arabic script based character segmentation: a review. In: Computer and Information Technology (WCCIT), 2013 World Congress on, IEEE, pp. 1–6. https://doi.org/10.1109/WCCIT.2013.6618741.
Nivre, J., 2015. Towards a universal grammar for natural language processing. In: Computational Linguistics and Intelligent Text Processing. Springer, pp. 3–16. https://doi.org/10.1007/978-3-319-18111-01.
Outahajala, M., Zenkouar, L., Benajiba, Y., Rosso, P., 2013. The development of a fine grained class set for amazigh pos tagging. In: Computer Systems and Applications (AICCSA), 2013 ACS International Conference on, IEEE, pp. 1–8. https://doi.org/10.1109/AICCSA.2013.6616440.
Petrov, S., Das, D., McDonald, R., 2013. A universal part-of-speech tagset, arXiv preprint arXiv:1104.2086.
Rahman, M.U., 2009. Sindhi morphology and noun inflections. In: Proceedings of the Conference on Language & Technology, pp. 74–81.
Ramteke, S., Ramteke, K., Dongare, R., Lexicon parser for syntactic and semantic analysis of devanagari sentence using hindi wordnet. Int. J. Adv. Res. Comput. Commun. Eng. 3 (4).
Shahrour, A., Khalifa, S., Habash, N., 2015. Improving arabic diacritization through syntactic analysis. In: EMNLP. pp. 1309–1315.
Singh, A., Thakur, N., Sharma, A., 2016. A review of supervised machine learning algorithms. In: Computing for Sustainable Global Development (INDIACom), 2016 3rd International Conference on, IEEE, pp. 1310–1315.
Tsarfaty, R., Seddah, D., Kübler, S., Nivre, J., 2013. Parsing morphologically rich languages: introduction to the special issue. Comput. Linguistics 39 (1), 15–22. https://doi.org/10.1162/COLIa00133.