Research Article

International Journal of Advanced Technology and Engineering Exploration, Vol 9(91)
ISSN (Print): 2394-5443 ISSN (Online): 2394-7454
http://dx.doi.org/10.19101/IJATEE.2021.875212

Identification and extraction of multiword expressions from Hindi & Urdu language in natural language processing

Vaishali Gupta[1*] and Nisheeth Joshi[2]
Department of Computer Science, IPS Academy, Institute of Engineering & Science, Indore, MP, India[1]
Department of Computer Science, Apaji Institute, Banasthali Vidyapith, Newai, Tonk, Rajsathan, India[2]

Received: 21-November-2021; Revised: 10-June-2022; Accepted: 13-June-2022

Abstract

Text can be translated from one language to another using statistical machine translation, but there are still gaps in the translations because of a lack of language resource material. Building a linguistic corpus necessarily requires the extraction of multiword expressions (MWE). MWE is a collection of words with idiomatic expression properties. However, due to its non-compositional meaning of distinctive words, identifying and extracting MWE is a time-consuming task. In this case, an automated system has been developed for the extraction of MWEs from Hindi and Urdu language sources automatically. The entire process includes tagging, pattern matching, an identification algorithm, and the extraction of MWEs from the data. Tagging each word with a unique part of speech tag is used as an input to the pattern-matching algorithm. Using pattern matching, MWE tags of specific patterns were selected, and the algorithm for automatic MWE detection was built on top of that. The conditional random field (CRF++) model was used to automatically extract the MWEs from data. Confusion matrix was used to conduct the automated evaluation of this proposed system. For Hindi and Urdu, the calculated overall accuracy is 96.82% and 96.62%, respectively.

Keywords

Bigrams, Tags, Multiword expression (MWE), Conditional random field (CRF), Confusion matrix.

1.Introduction

A class of linguistic forms called "multiword expressions" (MWEs) includes expressions that span traditional word boundaries and are found in a wide variety of languages. We must rethink linguistic processing in order to accommodate MWEs, which relies on a clear distinction between words and phrases. For natural language processing (NLP) applications, MWE handling is critical, as it raises a number of issues. This paper's primary goal is to shed light on how NLP applications like machine translation (MT) deal with MWEs. MWE processing and downstream applications such as MWE-aware parsing or MT are discussed in greater detail here. MWE handling and MWE-aware NLP applications have an insufficiency of proposed approaches. In fact, this research is motivated by the emergence of new approaches in the absence of a set of defined rules [1]. During the identification and extraction of MWE, we faced various challenges as follows [2].

Non-compositionality: Idiom like "bread and butter" is a good example of non-compositionality. This expression is used to describe someone who is sentimental and often naive, and as a result, its full meaning is obscure to those who only know the individual meanings of the words in question.

Ambiguity: Many NLP tasks face the challenge of ambiguity. For MWE processing, the choice between a compositional and a MWE reading of a sequence of words has the greatest impact, as illustrated by "I am struck by the way the rest of the world is confident of a better future. "While it's not always an exact MWE, it's a good approximation in most cases. However, the prepositional complement of the verb struck in the example is a regular prepositional complement. Syntactic analysis can help in determining that a sequence of words is a MWE in some cases. Analysis that takes by way of a MWE and thus, in this case, is an adverb is grammatically incorrect.

Variability: Because MWEs allow for a wide range of flexibility in how they are formed, the identification process is a challenge. There will be a

[*] Author for correspondence

low recall if only fixed forms are searched for, because the fixed form does not match all variations.

The MWEs are considered as a collocation which has set of words or sequence of words. These collocations have some linguistic characteristics like lexical, syntactic & semantic which impulses to turn up togetherness. In other words, MWEs consist of lexical items like multiple orthographic words such as in English: kick the bucket, by the way etc. and in Hindi: इधर-उधर (*Idhar-Udhar*), रसोईघर(*RasoiGhar*), कभीनहीं (*Kabhi Nahin*), मजाक-मस्ती (*Majak-Masti*), कद-काठी (*Kad-Kaathi*) etc. Some MWEs are expressed as a mixture of words, while others function as phrases; some of them, however, follow a set pattern. According to Baldwin [3][4], the classification or types of MWEs are as follows:

Lexicalized Phrases

i. Fixed expressions: It doesn't have any morpho-syntactic or internal modifications (e.g.: hard and fast, dark horse, down and out, break-up, break-down, by-the way).

ii. Semi-fixed expressions: Word order and construction are fixed in these expressions, but there is some lexical flexibility. For example, in the MWE "prostrate oneself," the term "oneself" contains several alternatives, such as "himself" or "herself." This category also includes compound nouns.

iii. Syntactically-flexible expression: Expressions with more than one word inserted between their constituent words have a greater range of syntactical variety. (e.g., get rid of the evidence, let the cat out of the bag—tell a secret)

Institutionalized Phrase

It is syntactically and semantically composed MWEs (e.g. Salt & Pepper, Bread & butter, traffic light, kindle excitement).

The following are examples of MWEs in English and Urdu, as translated by Google and human experts:

Example of English MWEs:

Source Sentence: (English) On the spot, he kicked the bucket in accident.

Google Translated:

(Hindi): मौके पर उन्होंने दुर्घटना में बाल्टी को लात मार दी |

(Urdu):–*موقینپرہی – انہونےحادثےمیں**بالٹیلاتماری***

(mauke par hui–unhonehaadsemeinbaaltilaatmaari)

Human Expert's Output:

(Hindi):वहदुर्घटनामेंमौकेपरहीमरगया|

(Urdu):وہ حادثے میناسی وقت مر گیا

(vahhaadsemeinusiwaqtmar gya)

In the above examples, MWE "kick the bucket" is not handled by even Google translator. It produces wrong output that is not meaningful.

Example of Urdu MWEs:

Source Sentence: (Urdu): رشید بہت مشہور ارباب سخن ہے

(Rashid bhut mashahoorarbabsukhanhai)

Google Translated:

(Hindi):राशिदबहुतप्रसिद्धअरबाबसुख़नहै

(English):The governingisvery famousspeech Rashid.

Human Expert's Output:

(Hindi):राशिदबहुतप्रसिद्धकलाकारहैं|

(English): Rashid is very famous artist.

As it can be seen that in the above example, MWE "**-ارباب سخن**arbab-sukhan" is not correctly handled by Google translator and it has generated the output that is not meaningful.

If we want to accurate translated output from MT, we are using various NLP techniques such as parsing, information retrieval, text alignment, ontology creation etc. At the translation time, handling MWE is a critical task. Therefore, we have proposed experimental system architecture for the automatic identification and extraction of MWE from Hindi and Urdu language monolingual corpuses. MT is currently unable to produce high-quality results because of a lack of resources for Indian languages. Word sense disambiguation, name entity recognition, multiword expression handling, etc. are just some of the issues that arise during translation time. The MWE issue was chosen as the focus of this research in order to improve the quality of MT. For dealing with MWEs, we have a plan. It focuses on identifying and extracting various types of MWE. Various types of MWEs are mentioned in the below *Figure 1:*

Some researchers have used various approaches for handling MWEs such as rule-based approach [5], classification-based approach [6], statistical approach [7], stepwise mining methodology [8] and hybrid approach [9] etc. In our proposed research work, we have used the pipeline-based approach as like this; the 1st step is to identify the MWE and then extract these MWEs from the corpus. For identification of different types of MWEs, various strategies and processes which can be manual, semiautomatic and automatic are developed and designed. For the extraction of MWEs, various machine learning approaches are developed like for tagging the data, n-gram approach can be used and for extracting Urdu and Hindi MWEs. One can use word-net, monolingual corpus and bilingual dictionaries of Hindi, Urdu and English Languages for all these approaches. To put it another way, any method of machine learning relies on human-provided linguistic knowledge to interpret the input. In this way, the identification and extraction of MWEs in MT can be determined and the precise output produced by machines.

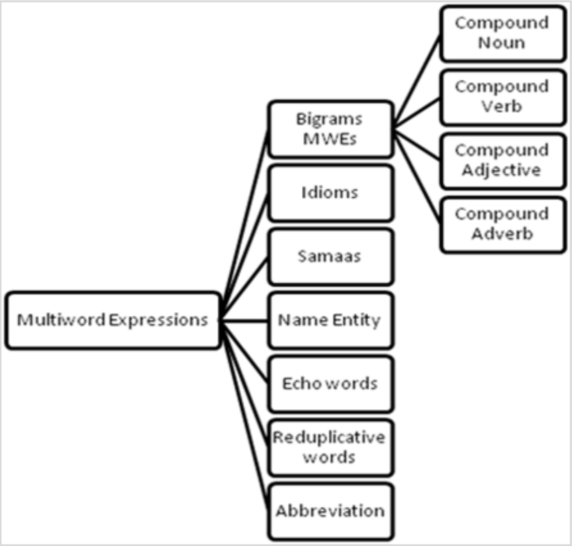


Figure 1 Classification of multiword expressions

The remaining paper is organized as follows. Section 2 outlines the important research related to handling multiword expression. Section 3 presents a proposed framework and methodology for multiword expression extraction on the basis of some specified rules and machine learning model. Section 4 presents the experiments and evaluation, Section 5 discusses the Experimental results and evaluates the proposed methods, and Section 6 concludes the paper.

2.Literature review

There has been very little research in this area, so it can be considered an unexplored domain with limited research. MWEs are typically extracted using a rule-based, semi-automatic, and automatic approach. MWE in MT has been identified as a problem by a number of different researchers [10−13]. The problem of MWEs in NLP has been addressed by some researchers, but they have not yet found a solution. Many researchers are currently focusing on this topic. Kulkarni and Finlayson (2011) [14] created a Java framework for recognising MWE (jMWE). When working with MWE, jMWE can help you create and verify the token. An MWE token detector application programming interface (API) is provided in this library, as well as a MWE index, which provides classes for building, storing, and accessing indices of valid MWE types. The "Test Harness" facility provided by the jMWE library can be used to test the performance of a MWE detector after the previous two facilities have been used. In this framework, authors have used different kinds of detection algorithms. Various types of MWE token detectors such as basic and filter resolver detectors are employed in this algorithm. Sinha (2011) [8] proposed a linguistic knowledge-based framework for the progressive mining of Hindi multi-word phrases. This study's authors discovered a slew of new MWEs in the Hindi language. MT was used to evaluate these MWEs. These were discovered through the use of a Hindi wordnet and an English-Hindi corpus. After identifying the MWEs, various extraction methods were employed. The majority of MWEs were derived from word co-occurrence and collocation in the corpus.

Chakraborty et al. (2014) [15] identified the Bengali multi-word phrases using method of semantic clustering. This method helped to identify clusters of word tokens with similar meanings in the document. We could use these clusters to see how closely related the individual words with another words. For noun-noun bigram MWEs, they employed the semantic clustering approach. In addition, well-known statistical models like pointwise mutual information (PMI) and the log likelihood ratio (LLR) were utilised. Using an Arabic corpus, Daoud et al. (2016) [16] described a method for extracting MWE. In the beginning, the researchers gathered 15.25 million Arabic tweets from 25 days of tweets. Only Arabic tweets were processed for tokenization into unigrams, bigrams, and trigrams using a language detector. The unigram lexical resources were used for stemming, and the bigram and trigram lexical resources were used to reduce noise. These lexical resources were then used to extract MWE in Arabic from those resources. They used a statistical approach to extract MWE. First, MWE candidates were retrieved, and then the ranking was assigned to those MWEs that were retrieved. There were 3,678,838 tokens found to be valid MWEs and 92 percent accuracy was found for most frequently occurring MWEs. The translation of MWE from English to Dogri has been proposed by Singh and Jamwal (2016) [17]. At the time of translation, MWE was a difficult task. As a result, they have analysed various types of MWEs and extracted from parallel corpora in order to solve the translation problem. The statistical translation system "Moses" was used to translate these MWEs from English to Dogri and vice versa. The GIZA++ and KenLM tools in the Moses toolkit are included. The word alignment and phrase transliteration training were both made easier with the GIZZ++ tool. Statistical language models were constructed using the KenLM tool. Probabilities at the sentence level were tallied. There were 83,618 sentences in the parallel corpus of English-Dogri language and about 80,000 MWE that were extracted in this study.

Singh et al. (2016) [11] proposed a Hindi and Marathi language MWE annotation scheme. They used a part of speech tagged corpus and Indo wordnet synsets to annotate their work. Only compound nouns and light verbs have been annotated with the help of a human expert. They found 3178 & 2556 valid MWE in Hindi and 1003 & 2416 valid MWE in Marathi for both compound nouns and light verbs. Any MWE system can use the annotated data of both languages as gold data. When it comes to identifying Magahi language MWEs, Kumar et al. (2017) [6] used support vector machine classification. There are approximately 75k words of Magahi language in this dataset. For this dataset, the Part of speech (POS) tags were manually entered. Eleven thousand tokens of tagged data are identified as MWE. A precision level of 81.57 percent was achieved with this method. Agrawal et al. (2018) [9] proposed a three-phase hybrid approach for the extraction of English multiword phrases. It is necessary to first separate the raw corpus into n-grams, then to filter them using Dice's coefficient and PMI method to calculate association scores, and finally to determine context similarity using the Latent Semantic Analysis method in the third stage of this three-phase procedure. A baseline and statistical analysis proposed by Joon and Singhal (2019) [18] compares the extraction of Hindi MWE with that of other languages. They included precision, recall, and f-measure in the baseline. There are several commonly used statistical measures such as the PMI, the Dice coefficient, and the modified Dice coefficient, but the relevance measures are being considered in addition to these. Measures are compared based on the frequency of MWE in the Hindi corpus, which are the most common.

Text simplification for Urdu has been done automatically by Qasmi et al. (2020) [19]. Using an unsupervised approach, the system was built to simplify complex Urdu text. Conditional random field-based part of speech tagged Urdu data was used, and the word2vec model was used to embed the words. This year, Han et al. (2020) [20] made available multi-lingual and bilingual MWE corpora extracted from root parallel corpora. German-English and Chinese-English MWE pairs in the collections are 3,159, 226 and 143, 042 after filtering. In MT experiments, they tested the quality of the extracted bilingual MWEs. Using MWEs in MT has resulted in improved translation performance on MWE terms in qualitative analysis and better overall evaluation scores on both German-English and Chinese-English language pairs, according to their preliminary experiments. When it comes to Persian, Fleischhauer (2020) [21] argued against treating all "bare noun + verb" sequences in contemporary Persian alike. For the purposes of this study, criteria were presented that could be used to separate light verb constructions from other predication construction types that looked similar on the surface. The primary goal of Goyal and Goyal (2020) [22] was to create an automated tool for extracting MWE from a parallel corpus of English and Punjabi. The rule-based approach, linguistic approach, statistical approach, and many other approaches were used in this tool to identify and extract MWEs from monolingual and parallel corpus of English and Punjabi and achieved more than 90% f-score value in some types of MWEs. Edition 1.2 of the PARSEME shared task for identifying verbal MWE was presented by Ramisch et al. (2020) [23]. Learning from previous editions suggests that the most difficult part of verbal multiword expression (VMWE) is identifying test cases that have never been seen in the training data. VMWEs that cannot be seen are the focus of this edition. They've divided annotated corpora into test corpora and provided non-annotated raw corpora to be used by complementary discovery methods, so the test corpora contain around 300 unseen VMWEs.

In the Persian language, Marszałek-Kowalewska (2021) [24] presented a work on the identification of MWEs. It aims to find loanwords in Persian and their Persian-language equivalents proposed by the academy of Persian language and literature in MWEs with specific lemmas. In order to find these MWEs, four association measures (AMs) are used and then evaluated. This is followed by an analysis of the list of MWEs, and a comparison of expressions with loanwords and their equivalents is made available. Emotion detection in code-mixed Twitter data was investigated by Tan et al. (2021) [25] using MWEs extracted from WordNet and WordNet Bahasa. Chinese character decomposition for neural MT with MWEs was proposed by Han et al. (2021) [26]. Researchers examined the impact of Chinese decomposition embedding and the accuracy with which these decompositions capture the original character sequences' meaning in this study. If the combination of decomposed MWE can improve the model learning, they investigated this further. Machine learning can be used to extract MWE from corpora, according to Jamwal et al. (2022) [27]. Simple heuristics are employed by the MWE, which consider the co-occurrence of MWEs. A new method for extracting MWE using supervised machine learning for the Dogri language has been proposed. When applied to a variety of test datasets, the proposed method achieves respectable levels of precision, recall, and F-score. Iwatsuki et al. (2022) [28] proposed a new method that is able to handle a wide range of formulaic expressions and their spans. Their approach sees a sentence as having a formulaic and non-formulaic component. Then, instead of trying to extract formulaic expressions from a whole corpus, by extracting them from each sentence, different forms can be dealt with simultaneously. In order to avoid the problem of diversity, they compared the extracted expressions to an already-existing set of terms. They also proposed a new extraction method that used named entities and dependency structures to remove the non-formulaic portion of a sentence. The concreteness ratings for 62,889 MWEs presented by Muraki et al. (2022) [29] were compared to the existing concreteness ratings for single words and two-word expressions. As the first large dataset of ratings for MWEs, these new ratings should prove useful to researchers in the fields of language acquisition and processing, as well as NLP and textual analysis.

3.Proposed methodology

A comprehensive framework developed in this paper which was used to identify and extract MWEs from Hindi and Urdu corpora. In order to determine the probability of each MWE tag, a stochastic approach is used, and the MWEs with the highest probability are extracted. Using tagged corpora of Hindi and Urdu, we created the MWE tagset in phase one. Phase-2 involved developing an algorithm to identify MWEs and phase-3 saw the automatic extraction of these MWEs. The final phase of performance evaluation is completed using both human and automated evaluations at this point. *Figure 2* depicts the proposed framework.

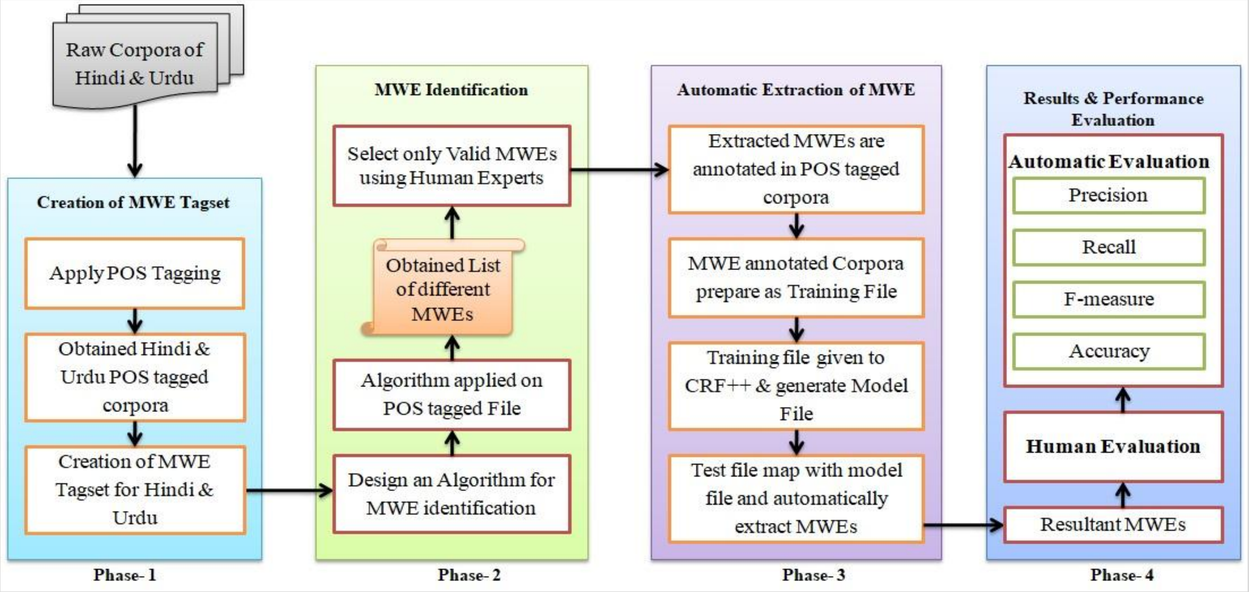

Raw Corpora of Hindi & Urdu
Creation of MWE Tagset
Apply POS Tagging
Obtained Hindi & Urdu POS tagged corpora
Creation of MWE Tagset for Hindi & Urdu
Phase- 1
MWE Identification
Select only Valid MWEs using Human Experts
Obtained List of different MWEs
Algorithm applied on POS tagged File
Design an Algorithm for MWE identification
Phase- 2
Automatic Extraction of MWE
Extracted MWEs are annotated in POS tagged corpora
MWE annotated Corpora prepare as Training File
Training file given to CRF++ & generate Model File
Test file map with model file and automatically extract MWEs
Phase- 3
Results & Performance Evaluation
Automatic Evaluation
Precision
Recall
F-measure
Accuracy
Human Evaluation
Resultant MWEs
Phase- 4

Figure 2 Comprehensive framework for extraction of MWEs

3.1Phase-1 (Creation of MWE Tagset)

A.Collection of corpus

The raw corpora of both Hindi and Urdu are required initially for the identification and extraction of MWEs from text. In addition, we're compiling idiomatic expressions in Hindi and Urdu from a variety of sources. The following are the specifics of this collection-

(i)Raw corpora of Hindi & Urdu

The tourism, health, and agricultural domain-based corpus of 60k sentences are compiled using online English-language resources. These English phrases are translated into Hindi as well in Urdu, which are mentioned in below *Table 1*.

Table 1 Details of English, Hindi and Urdu corpora

Corpus	Hindi	Urdu	English
Health	25,000	25,000	25,000
Tourism	25,000	25,000	25,000
Agriculture	10,000	10,000	10,000
Total	60,000	60,000	60,000

These corpora were processed manually through unified IL POS tagset [30] for generating the tagged corpora of each word for the respective dataset and this process took approx4 months. The Tagged corpus which was generated by the human expert was not giving the huge number of tagged corpora. So that, we have developed automatic POS tagger, by which lots of raw corpus was tagged automatically. Then, we have obtained the total 60k sentences of tagged Hindi and Urdu corpora.

(ii)Collection of Hindi & Urdu idiomatic expression

An idiomatic expression is a MWE that cannot be predicted entirely or partially from its constituent words. The "MWE_IP" tag represents idioms. For example: In Hindi, छींटाकशीकरना (*chheetaKashikarna*), जैसादेशवैसाभेष (*jaisadeshvisabhesh*) and in Urdu,سوتے نصیب جاگے (*sotenaseebjage*), دل کا غبارباہر نکلنا (*dilkaghubaarbaharnikalna*).

Idiomatic idioms in Hindi and Urdu are collected using a manual technique from text books, websites, novels, and grammar books. 3394 &2807 idioms are collected in Hindi and Urdu language respectively. Some examples are shown in following *Table 2*.

Table 2 Examples of Hindi and Urdu idioms

Idioms	Meaning
नाकऊँचीहोना (*naakunchihona*)	To be honourable
बढ़-चढ़करबोलना (*badh-chadhkarbolna*)	Overwhelmingly speak
آبرو میں بٹّا لگانا (*aabrumeinbattalagna*)	Stain upon one's honour
اپنا اللو سیدھا کرنا (*apnaulluseedhakarna*)	To cheat

B. POS Tagging

For machine learning, the sentences were tagged by developing a POS tagger which was based on conditional random fields. In the process, by applying some features the training file of data set was created and further the resultant training file was used with template file. According to some specific rules of conditional random field (CRF++) template file was created.

Therefore, in this way machine learned with the help of training file and template file which was created using CRF++ model and then one model file was generated. Through this model file, POS tags were assigned for test datasets. Detailed process of POS tagging and training the machine is discussed in research paper of Khan et al (2019) [31] Kaur and Saini [32].

C. MWE Tagset

With compositional structure of words, some rules have been created for identification of MWEs. According to these rules of MWE identification [33−37], the following *Table 3* shows the standard MWE tagset for both the languages.

Table 3 Tagset of MWEs for Hindi and Urdu language

S. No.	MWE Tags	Category of Tag
1	MWE_C_N	Noun with a Compound Form
2	MWE_C_V	Verb with a Double Meaning
3	MWE_C_ADJ	Adjective Compound
4	MWE_C_ADV	Adverbial Compound
5	MWE_ECH	Words that are repeated as echo
6	MWE_A	Words that have been shortened as abbreviation
7	MWE_RP	Redundant words
8	MWE_IP	Idiomatic expressions
9	MWE_C_S	Samaas is a word that is used only in Hindi
10	MWE_NE	Entities are given names

3.2Phase-2 (MWE Identification)

An algorithm for identifying various types of MWEs was developed using rule-based patterns in this framework. Some of the strategies are manual, semi-automatic, and automatic. Rules and strategies are laid out in the following *Table 4*.

A. Rules for MWE pattern

In the following *Table 4*, different types of MWE tags are displayed with examples and some rules are also defined for compound words. By using automatic approach, the compound words were identified, and reduplicative, echo and abbreviated words were identified by using semiautomatic approach. Also, manually Samaas and idiomatic phrases for both the languages were identified.

Table 4 Rules for MWE tagset

MWE Tagset	Rules (Tag Categories)	Examples of Hindi and Urdu
Compound Noun (MWE_C_N)	Noun + Noun Noun + Verb Noun + Adjective Noun + Adverb	पूजा-अर्चना*(pooja-archana)* सर-दर्द*(sar-dard)* کتب- خانہ*(kutub-khana)* غلط- فہمیاں*(galat-fahmiya)*
Compound Verb (MWE_C_V)	Verb + Verb	सोचे-समझे*(soche-samjhe)* ناچنے- گانے*(nachne-gaane)*
Compound Adjective (MWE_C_ADJ)	Adjective + Noun Adjective + Verb Adjective + Adjective	साफ़-सुथरे*(saaf-suthre)* पौष्टिक-आहार*(poshtik-aahar)* مکمّل ہونا*(mukamal-hona)* احتیاط برتیں*(ahtyaat-bartein)*
Compound Adverb (MWE_C_ADV)	Adverb + Adverb Adverb + Verb Adverb + Noun	निवारक-दर्द*(nivarak-dard)* तेज-चलना*(tej-chalna)* رجانہ-ورجس*(rojana-varjis)* پیدل-چلنے*(paidal-chalne)*
MWE_RP	Reduplicative Tags	धीरे-धीरे*(dheere-dheere)* ठीक-ठाक*(theek-thaak)* चुप-चाप*(chup-chaap)* آہستہ – آہستہ*(aahista-aahista)* رفتہ-رفتہ*(raftah-raftah)*
MWE_ECH	Echo words	अलग-थलग*(alag-thalag)* टेढा-मेढा*(tedha-medha)* چائے-وائے*(chaay-vaay)* چھوٹی- موٹی*(chhoti-mooti)*
MWE_A	Abbreviated Words.	कि.मी.*(ki.mi.)* ई.पू. (ii.pu.) ٹی.بی. *(T.B.)* ٹی.وی. *(T.V.)*
MWE_IP	Idiomatic phrases	سرقلمکرنا*(sarqalamkarna)* دلباغباغہونا*(dilbaghbaghhona)* अन्धेकेहाथबटेर*(andheke hath bater)* एकपंथदोकाज*(ekpanth do kaaj)*
MWE_C_S	Samaas	लौहपुरुष*(lauhpurush)* रसोईघर*(rasoighar)* पीताम्बर*(peetambar)*

B. MWE identification Algorithm

The tagset of MWEs is used to develop the MWE identification algorithm. The following approach is used to detect various types of bigrams, MWEs, and compound words in Hindi and Urdu corpora. This algorithm is applied on corpora of 60,000 sentences and gets the millions of bigrams or MWEs. All bigrams were not accurate so unique and valid MWEs were identified manually for both Hindi and Urdu. Total 1,892 MWEs for Hindi and 1,475 MWEs for Urdu were extracted. Here, idioms are also identified from following algorithm [38].

Algorithm for Compound Tags

Step 1: Take an input file as tagged Hindi or Urdu Corpora.
Step 2: Apply the MWE rules on input file.
Step 3: Perform pattern matching-

(a) Extract as MWE_C_N tag if tagged words are mapped with "NN+NN" or "NN+JJ" or "NN+RB" or "NN+VM" pattern rules.
(b) Extract as MWE_C_V tag if tagged words are mapped using "VM+VM" or "VM+VAUX" pattern rules.
(c) Extract as MWE_C_ADJ tag if tagged words are mapped with "JJ+JJ" or "JJ+NN" or "JJ+VM" pattern rules.
(d) Extract as MWE_C_ADV tag if tagged words are mapped with "RB+RB" or "RB+NN" or "RB+VM" pattern rules.
Step 4: Obtain a list of all MWE tags for compound types.

Algorithm for Idiomatic Tags

Step 1: Take an input file of Hindi or Urdu raw corpora.
Step 2: Design a Knowledgebase of Hindi/Urdu Idioms from various sources.
Step 3: Use the idioms database to do surface matching.
Step 4: If surface matching of idioms is possible in the input file, those idioms are retrieved and labelled as MWE_IP.
Step 5: Again, and again look for surface matching.
Step 6: When the entire corpus has been processed for surface matching, the algorithm is terminated.

While we've got the complete list of MWEs, some of them are invalid and should not be included. As a result, we need a Hindi and Urdu language expert in order to create a list of valid MWEs.

3.3Phase-3 (Automatic extraction of MWE)

MWE tagged corpora are created by replacing a list of MWEs with POS tagged corpora. A training file for machine learning can then be created by applying feature extraction to the corpora. After training, we can test our unannotated data and automatically extract the multiword expression. Prepare the training file first for automatic multiword expression extraction, and then use CRF++ models to process the training file. Overall flowchart for automatic extraction of MWEs from text is presented in following *Figure 3.*

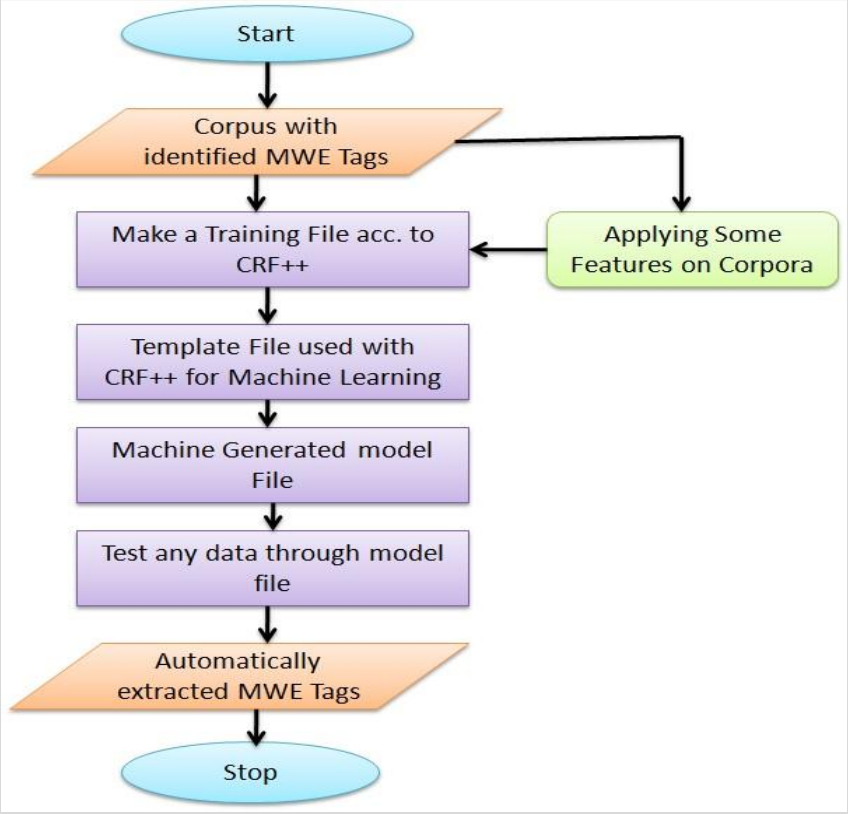

Start
Corpus with identified MWE Tags
Make a Training File acc. to CRF++
Applying Some Features on Corpora
Template File used with CRF++ for Machine Learning
Machine Generated model File
Test any data through model file
Automatically extracted MWE Tags
Stop

Figure 3 Flowchart of proposed approach for automatic extraction of MWE

A. Preparation of training& test file:

If we want to train the system for automatic extraction of MWEs through machine learning models then we must follow the constraint of those models. These constraints are as follows-

- Training file must have some features.
- For training, huge amount of data set required.
- In training file, there must be separation between sentences.
- It should be designed as prescribed format given by CRF++ model.

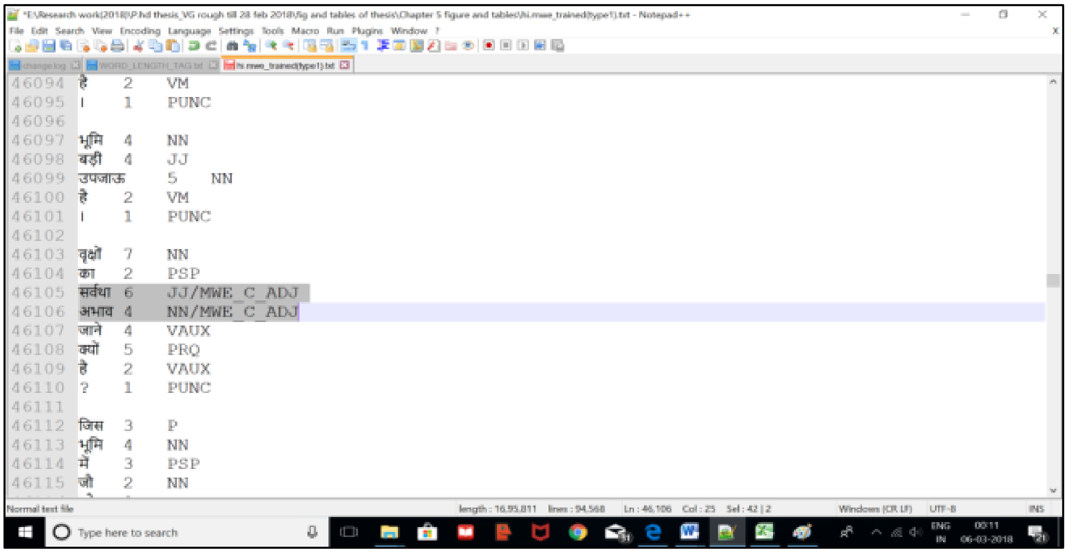
Here, we are creating different types of training files by applying some features. Through these training files, we will train the system and then we will test the test files with respect to generated model file. Here, we are designed three types of training file as follows:

(i)Training File (Type-I):

This type of training file consists of two features. The following words are connected with these characteristics:

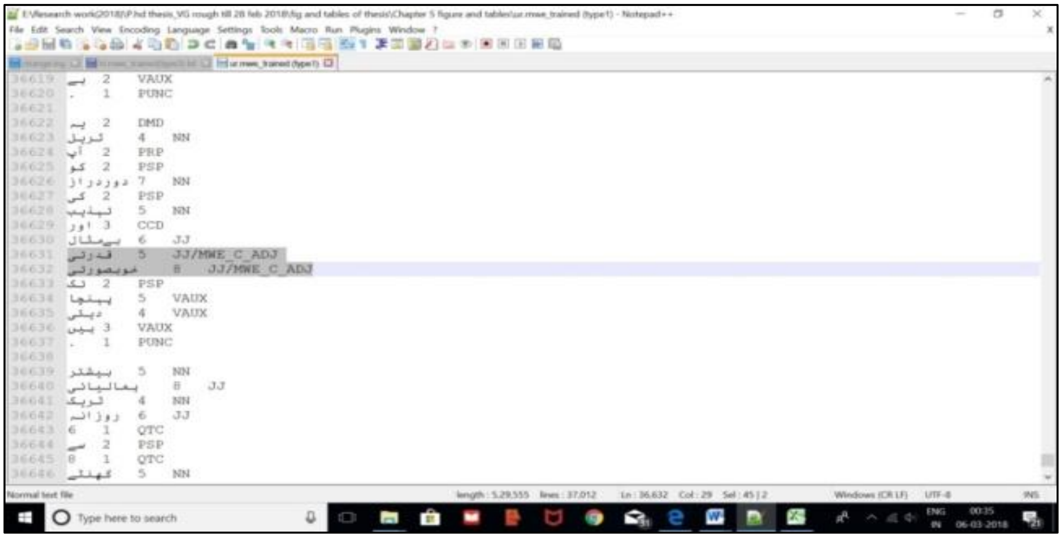
- Determine the length of the input words.
- Associate the tag of input words

Then format of training is as follows in *Figures 4* and *5*.



```
है	2	VM
।	1	PUNC

भूमि	4	NN
बड़ी	4	JJ
उपजाऊ		5	NN
है	2	VM
।	1	PUNC

वृक्षों	7	NN
का	2	PSP
सर्वथा	6	JJ/MWE_C_ADJ
अभाव	4	NN/MWE_C_ADJ
जाने	4	VAUX
क्यों	5	PRQ
है	2	VAUX
?	1	PUNC

जिस	3	P
भूमि	4	NN
में	3	PSP
जो	2	NN
```

Figure 4 MWE tagged Hindi corpora sample training file (Type-I)



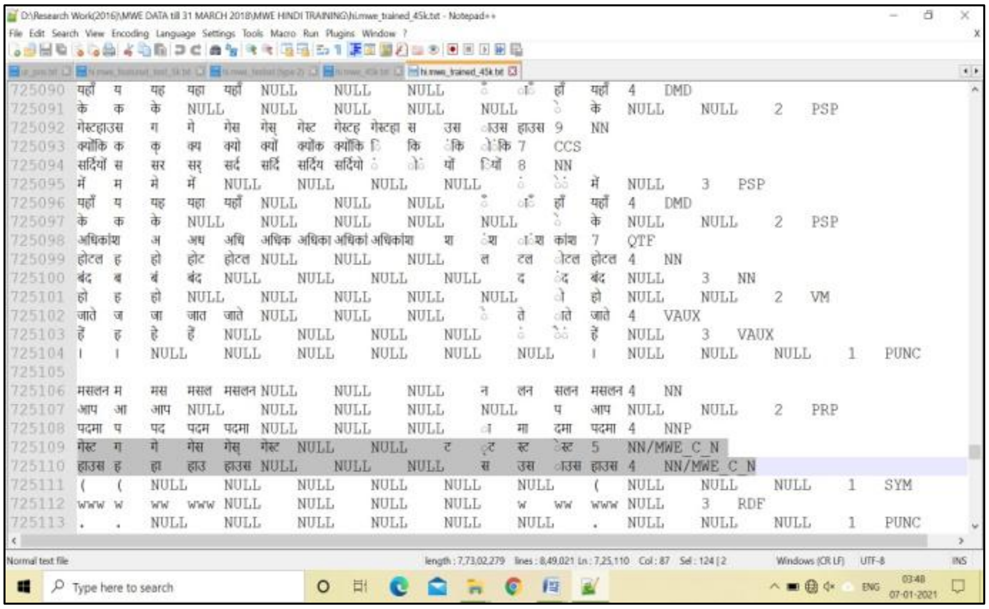
```
ہے	2	VAUX
۔	1	PUNC

یہ	2	DMD
ٹریل		4	NN
آپ	2	PRP
کو	2	PSP
دوردراز	7	NN
کی	2	PSP
تہذیب		5	NN
اور	3	CCD
بےمثال		6	JJ
قدرتی		5	JJ/MWE_C_ADJ
خوبصورتی		8	JJ/MWE_C_ADJ
تک	2	PSP
پہنچا		5	VAUX
دیتی		4	VAUX
ہیں	3	VAUX
۔	1	PUNC

بیشتر		5	NN
ہمالیائی		8	JJ
ٹریک		4	NN
روزانہ		6	JJ
6	1	QTC
سے	2	PSP
8	1	QTC
گھنٹے		5	NN
```

Figure 5 MWE tagged Urdu corpora sample training file (Type-I)

(ii)Training file (Type-II): Second training file is created by applying 13 features such as-

- Extract prefixes up to 4 characters in length (1st to 4th feature)
- Extract the suffixes up to a maximum length of 7 characters (5th to 11th feature)
- Work out how long each word is (12th feature)
- Assign the MWE tag to words as a final feature (13th feature)

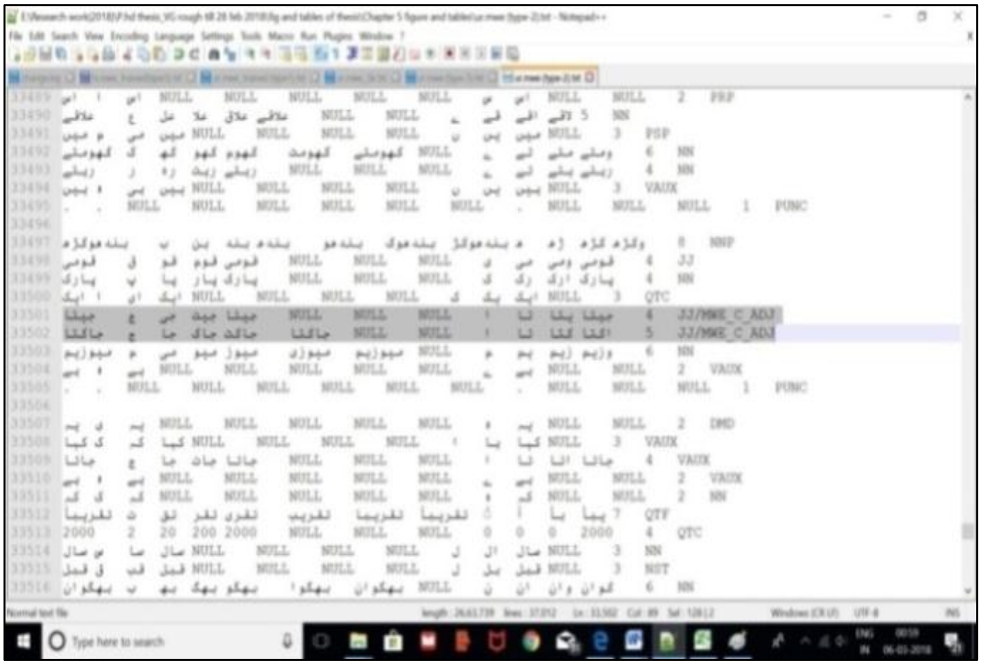
Through use of these features to the MWE tagged file, we obtain the training file shown in *Figures 6* and *7*


D:\Research Work(2016)\MWE DATA till 31 MARCH 2018\MWE HINDI TRAINING\hi.mwe_trained_45k.txt - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
hi.mwe_trained_45k.txt
Type here to search
ENG
03:48
07-01-2021

```
यहाँ   य    यह   यहा   यहाँ   NULL   NULL   NULL   ँ    ाँ    हाँ    यहाँ   4   DMD
के    क    के   NULL  NULL  NULL   NULL   NULL   े    के   NULL   NULL   2   PSP
गेस्टहाउस   ग    गे   गेस   गेस्   गेस्ट   गेस्टह  गेस्टहा  स   उस   ाउस   हाउस   9   NN
क्योंकि  क    क्   क्य   क्यो   क्यों   क्योंक   क्योंकि  ि   कि   ंकि   ोंकि   7   CCS
सर्दियों  स    सर   सर्   सर्द   सर्दि   सर्दिय   सर्दियो  ं   ों   यों   ियों   8   NN
में    म    मे   में   NULL   NULL   NULL   NULL   ं    ें    में   NULL   3   PSP
यहाँ   य    यह   यहा   यहाँ   NULL   NULL   NULL   ँ    ाँ    हाँ    यहाँ   4   DMD
के    क    के   NULL  NULL  NULL   NULL   NULL   े    के   NULL   NULL   2   PSP
अधिकांश  अ    अध   अधि   अधिक  अधिका  अधिकां  अधिकांश   श   ंश   ांश   कांश   7   QTF
होटल   ह    हो   होट   होटल   NULL   NULL   NULL   ल   टल   ोटल   होटल   4   NN
बंद    ब    बं    बंद   NULL   NULL   NULL   NULL   द   ंद    बंद   NULL   3   NN
हो    ह    हो   NULL  NULL   NULL   NULL   NULL   ो   हो   NULL   NULL   2   VM
जाते   ज    जा   जात   जाते   NULL   NULL   NULL   े   ते   ाते   जाते   4   VAUX
हैं    ह    है    हैं    NULL   NULL   NULL   NULL   ं   ैं    हैं    NULL   3   VAUX
।     ।    NULL  NULL  NULL   NULL   NULL   NULL   ।   NULL   NULL   NULL   1   PUNC

मसलन  म    मस   मसल  मसलन  NULL   NULL   NULL   न   लन   सलन   मसलन  4   NN
आप   आ    आप   NULL  NULL   NULL   NULL   NULL   प   आप   NULL   NULL   2   PRP
पदमा   प    पद   पदम   पदमा   NULL   NULL   NULL   ा   मा   दमा   पदमा   4   NNP
गेस्ट   ग    गे   गेस   गेस्   गेस्ट   NULL   NULL   ट   ्ट   स्ट   ेस्ट   5   NN/MWE_C_N
हाउस   ह    हा   हाउ   हाउस   NULL   NULL   NULL   स   उस   ाउस   हाउस   4   NN/MWE_C_N
(     (    NULL  NULL  NULL   NULL   NULL   NULL   (   NULL   NULL   NULL   1   SYM
www   w    ww   www   NULL   NULL   NULL   NULL   w   ww   www   NULL   3   RDF
.     .    NULL  NULL  NULL   NULL   NULL   NULL   .   NULL   NULL   NULL   1   PUNC
```

Normal text file
length : 7,73,02,279 lines : 8,49,021 Ln : 7,25,110 Col : 87 Sel : 124 | 2
Windows (CR LF) UTF-8 INS

Figure 6 MWE tagged Hindi corpora sample training file (Type-II)


File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Type here to search

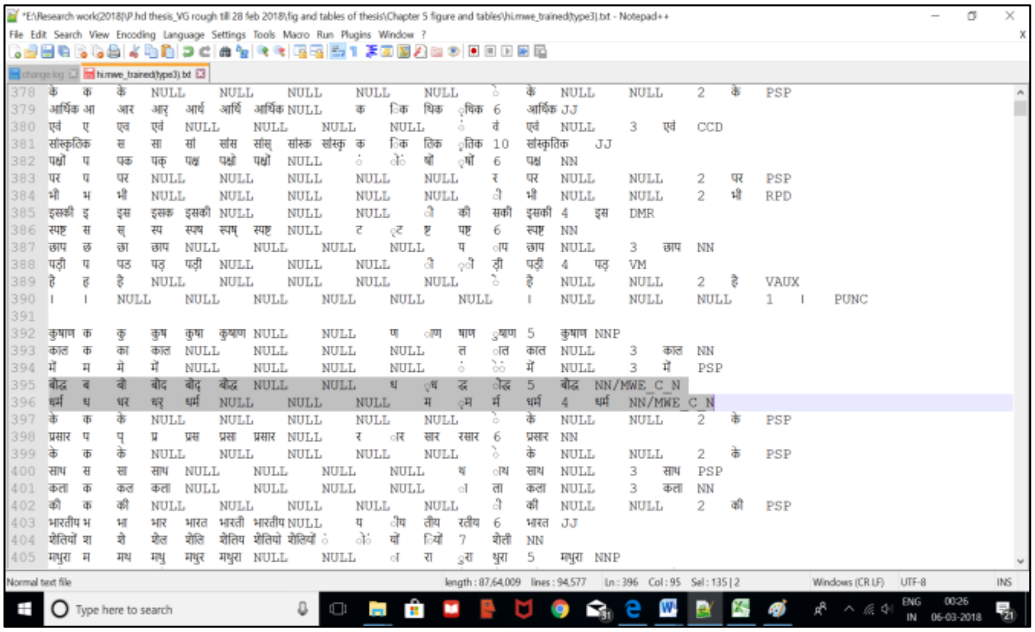
```
اس   ا   اس   NULL   NULL   NULL   NULL   NULL   س   اس   NULL   NULL   2   PRP
علاقے   ع   عل   علا   علاق   علاقے   NULL   NULL   ے   قے   اقے   لاقے   5   NN
میں   م   می   میں   NULL   NULL   NULL   NULL   ں   یں   میں   NULL   3   PSP
گھومنے   گ   گھ   گھو   گھوم   گھومن   گھومنے   NULL   ے   نے   منے   ومنے   6   NN
زیبنے   ز   زی   زیب   زیبن   NULL   NULL   NULL   ے   نے   بنے   یبنے   4   NN
ہیں   ہ   ہی   ہیں   NULL   NULL   NULL   NULL   ں   یں   ہیں   NULL   3   VAUX
.   .   NULL   NULL   NULL   NULL   NULL   NULL   .   NULL   NULL   NULL   1   PUNC

[illegible]   8   NNP
قومی   ق   قو   قوم   قومی   NULL   NULL   NULL   ی   می   ومی   قومی   4   JJ
پارک   پ   پا   پار   پارک   NULL   NULL   NULL   ک   رک   ارک   پارک   4   NN
ایک   ا   اي   ایک   NULL   NULL   NULL   NULL   ک   یک   ایک   NULL   3   QTC
جیتا   ج   جی   جیت   جیتا   NULL   NULL   NULL   ا   تا   یتا   جیتا   4   JJ/MWE_C_ADJ
جاگتا   ج   جا   جاگ   جاگت   جاگتا   NULL   NULL   ا   تا   گتا   اگتا   5   JJ/MWE_C_ADJ
میوزیم   م   می   میو   میوز   میوزی   میوزیم   NULL   م   یم   زیم   وزیم   6   NN
ہے   ہ   ہے   NULL   NULL   NULL   NULL   NULL   ے   ہے   NULL   NULL   2   VAUX
.   .   NULL   NULL   NULL   NULL   NULL   NULL   .   NULL   NULL   NULL   1   PUNC

یہ   ی   یہ   NULL   NULL   NULL   NULL   NULL   ہ   یہ   NULL   NULL   2   DMD
کہا   ک   کہ   کہا   NULL   NULL   NULL   NULL   ا   ہا   کہا   NULL   3   VAUX
جاتا   ج   جا   جات   جاتا   NULL   NULL   NULL   ا   تا   اتا   جاتا   4   VAUX
ہے   ہ   ہے   NULL   NULL   NULL   NULL   NULL   ے   ہے   NULL   NULL   2   VAUX
کہ   ک   کہ   NULL   NULL   NULL   NULL   NULL   ہ   کہ   NULL   NULL   2   NN
تقریبا   ت   تق   تقر   تقری   تقریب   تقریبا   تقریبا   ً   ا   با   یبا   7   QTF
2000   2   20   200   2000   NULL   NULL   NULL   0   00   000   2000   4   QTC
سال   س   سا   سال   NULL   NULL   NULL   NULL   ل   ال   سال   NULL   3   NN
قبل   ق   قب   قبل   NULL   NULL   NULL   NULL   ل   بل   قبل   NULL   3   NST
بھگوان   ب   بھ   بھگ   بھگو   بھگوا   بھگوان   NULL   ن   ان   وان   گوان   6   NN
```

Figure 7 MWE tagged Urdu corpus sample training file (Type-II)

We've been using the above training file with the template file to train the CRF++ model, then generating the model file after learning.

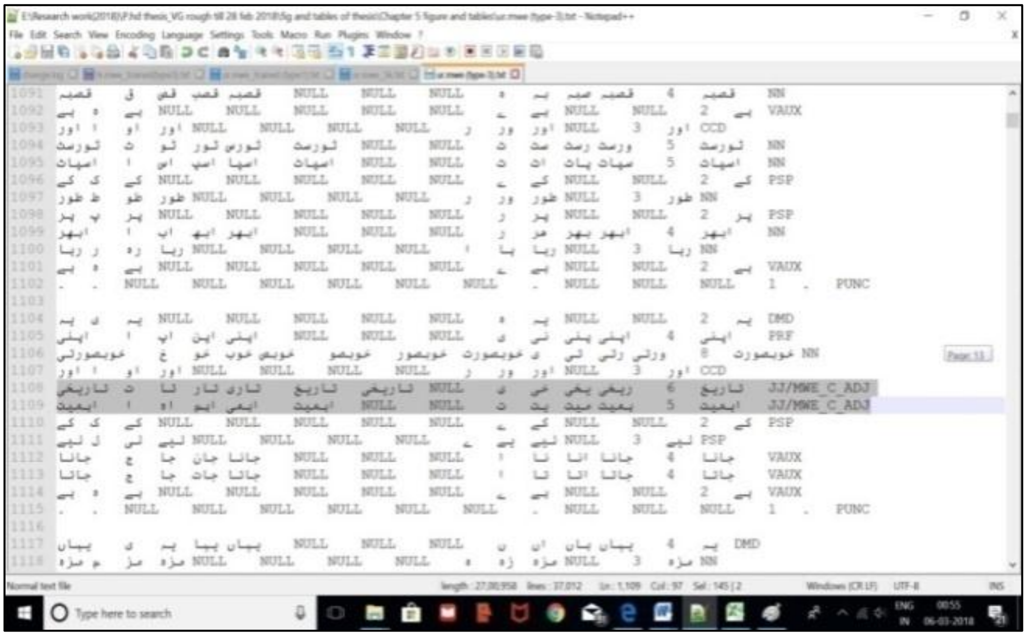
(iii)Training file (Type-III): In this type of training file, we are associated one another feature in training file type-II. Here another feature is

- Associate the root form of input word as a 13th feature and last feature will be associated as a tag. So, we can see additional feature "root" in 13thcolumn [39, 40] of below training file in *Figure 8* and *9*.


File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
change log
hi.mwe_trained(type3).txt
Type here to search
ENG IN 00:26 06-03-2018

```
*E:\Research work(2018)\P.hd thesis_VG rough till 28 feb 2018\fig and tables of thesis\Chapter 5 figure and tables\hi.mwe_trained(type3).txt - Notepad++
के      क      के      NULL    NULL    NULL    NULL    NULL    े      के      NULL    NULL    2   के    PSP
आर्थिक  आ      आर     आर्     आर्थ    आर्थि    आर्थिक  NULL    क      िक     थिक    ्थिक   6   आर्थिक JJ
एवं     ए      एव     एवं     NULL    NULL    NULL    NULL    ं      वं     एवं     NULL    3   एवं    CCD
सांस्कृतिक स    सा     सां     सांस    सांस्    सांस्क   सांस्कृ   क      िक     तिक    ृतिक   10  सांस्कृतिक  JJ
पक्षों   प      पक     पक्     पक्ष    पक्षो   पक्षों   NULL    ं      ों     षों     ्षों    6   पक्ष    NN
पर      प      पर     NULL    NULL    NULL    NULL    NULL    र      पर     NULL    NULL    2   पर    PSP
भी      भ      भी     NULL    NULL    NULL    NULL    NULL    ी      भी     NULL    NULL    2   भी    RPD
इसकी    इ      इस     इसक    इसकी   NULL    NULL    NULL    ी      की     सकी    इसकी   4   इस    DMR
स्पष्ट   स      स्     स्प     स्पष    स्पष्   स्पष्ट   NULL    ट      ्ट     ष्ट    पष्ट    6   स्पष्ट  NN
छाप     छ      छा     छाप    NULL    NULL    NULL    NULL    प      ाप     छाप    NULL    3   छाप   NN
पढ़ी     प      पढ     पढ़     पढ़ी    NULL    NULL    NULL    ी      ़ी     ढ़ी     पढ़ी    4   पढ़    VM
है      ह      है     NULL    NULL    NULL    NULL    NULL    ै      है     NULL    NULL    2   है    VAUX
।       ।      NULL   NULL    NULL    NULL    NULL    NULL    ।      NULL   NULL    NULL    1   ।     PUNC

कुषाण   क      कु     कुष     कुषा    कुषाण   NULL    NULL    ण      ाण     षाण    ुषाण   5   कुषाण  NNP
काल     क      का     काल    NULL    NULL    NULL    NULL    ल      ाल     काल    NULL    3   काल   NN
में      म      मे      में      NULL    NULL    NULL    NULL    ं      ें      में     NULL    3   में    PSP
बौद्ध    ब      बौ     बौद     बौद्    बौद्ध    NULL    NULL    ध      ्ध     द्ध     ौद्ध    5   बौद्ध   NN/MWE_C_N
धर्म     ध      धर     धर्     धर्म     NULL    NULL    NULL    म      ्म     र्म     धर्म    4   धर्म    NN/MWE_C_N
के      क      के      NULL    NULL    NULL    NULL    NULL    े      के      NULL    NULL    2   के    PSP
प्रसार   प      प्     प्र      प्रस     प्रसा    प्रसार   NULL    र      ार     सार    रसार   6   प्रसार  NN
के      क      के      NULL    NULL    NULL    NULL    NULL    े      के      NULL    NULL    2   के    PSP
साथ     स      सा     साथ    NULL    NULL    NULL    NULL    थ      ाथ     साथ    NULL    3   साथ   PSP
कला     क      कल     कला    NULL    NULL    NULL    NULL    ा      ला     कला    NULL    3   कला   NN
की      क      की     NULL    NULL    NULL    NULL    NULL    ी      की     NULL    NULL    2   की    PSP
भारतीय  भ      भा     भार     भारत    भारती   भारतीय  NULL    य      ीय     तीय    रतीय   6   भारत  JJ
शैलियों  श      शै     शैल     शैलि    शैलिय   शैलियो  शैलियों  ं      ों     यों     ियों    7   शैली   NN
मथुरा   म      मथ     मथु     मथुर    मथुरा   NULL    NULL    ा      रा     ुरा    थुरा    5   मथुरा  NNP
Normal text file    length : 87,64,009   lines : 94,577    Ln : 396   Col : 95   Sel : 135 | 2    Windows (CR LF)   UTF-8   INS
```

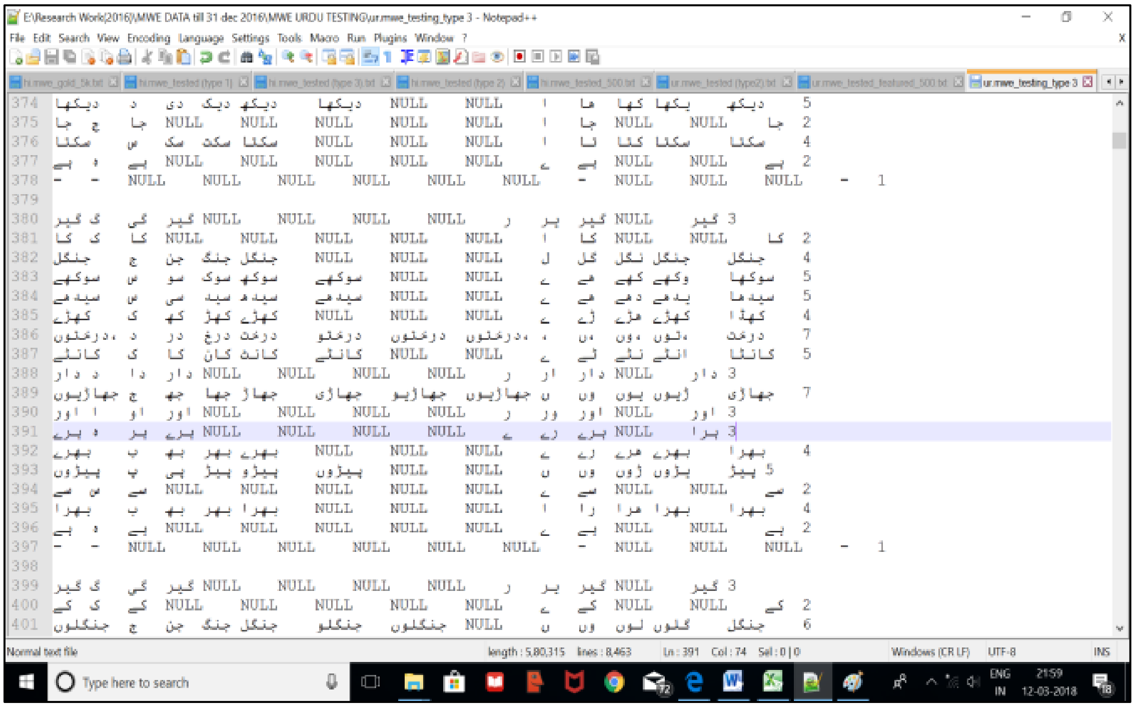
Figure 8 MWE tagged Hindi corpora sample training file (Type-III)


File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
change log
ur.mwe (type-3).txt
Page 11
Type here to search
ENG IN 00:55 06-03-2018

```
E:\Research work(2018)\P.hd thesis_VG rough till 28 feb 2018\fig and tables of thesis\Chapter 5 figure and tables\ur.mwe (type-3).txt - Notepad++
قصبہ     ق    قص    قصب    قصبہ    NULL    NULL    NULL    ہ    بہ    صبہ    قصبہ    4    قصبہ    NN
ہے       ہ    ہے    NULL    NULL    NULL    NULL    NULL    ے    ہے    NULL    NULL    2    ہے    VAUX
اور      ا    او    اور    NULL    NULL    NULL    NULL    ر    ور    اور    NULL    3    اور    CCD
ٹورسٹ    ٹ    ٹو    ٹور    ٹورس    ٹورسٹ    NULL    NULL    ٹ    سٹ    رسٹ    ورسٹ    5    ٹورسٹ    NN
اسپاٹ    ا    اس    اسپ    اسپا    اسپاٹ    NULL    NULL    ٹ    اٹ    پاٹ    سپاٹ    5    اسپاٹ    NN
کے       ک    کے    NULL    NULL    NULL    NULL    NULL    ے    کے    NULL    NULL    2    کے    PSP
طور      ط    طو    طور    NULL    NULL    NULL    NULL    ر    ور    طور    NULL    3    طور    NN
پر       پ    پر    NULL    NULL    NULL    NULL    NULL    ر    پر    NULL    NULL    2    پر    PSP
ابھر     ا    اب    ابھ    ابھر    NULL    NULL    NULL    ر    ھر    بھر    ابھر    4    ابھر    NN
رہا      ر    رہ    رہا    NULL    NULL    NULL    NULL    ا    ہا    رہا    NULL    3    رہا    NN
ہے       ہ    ہے    NULL    NULL    NULL    NULL    NULL    ے    ہے    NULL    NULL    2    ہے    VAUX
.        .    NULL    NULL    NULL    NULL    NULL    NULL    .    NULL    NULL    NULL    1    .    PUNC

یہ       ی    یہ    NULL    NULL    NULL    NULL    NULL    ہ    یہ    NULL    NULL    2    یہ    DMD
اپنی     ا    اپ    اپن    اپنی    NULL    NULL    NULL    ی    نی    پنی    اپنی    4    اپنی    PRF
خوبصورتی    خ    خو    خوب    خوبص    خوبصو    خوبصور    خوبصورت    ی    تی    رتی    ورتی    8    خوبصورت    NN
اور      ا    او    اور    NULL    NULL    NULL    NULL    ر    ور    اور    NULL    3    اور    CCD
تاریخی   ت    تا    تار    تاری    تاریخ    تاریخی    NULL    ی    خی    یخی    ریخی    6    تاریخ    JJ/MWE_C_ADJ
اہمیت    ا    اہ    اہم    اہمی    اہمیت    NULL    NULL    ت    یت    میت    ہمیت    5    اہمیت    JJ/MWE_C_ADJ
کے       ک    کے    NULL    NULL    NULL    NULL    NULL    ے    کے    NULL    NULL    2    کے    PSP
لیے      ل    لی    لیے    NULL    NULL    NULL    NULL    ے    یے    لیے    NULL    3    لیے    PSP
جانا     ج    جا    جان    جانا    NULL    NULL    NULL    ا    نا    انا    جانا    4    جانا    VAUX
جاتا     ج    جا    جات    جاتا    NULL    NULL    NULL    ا    تا    اتا    جاتا    4    جاتا    VAUX
ہے       ہ    ہے    NULL    NULL    NULL    NULL    NULL    ے    ہے    NULL    NULL    2    ہے    VAUX
.        .    NULL    NULL    NULL    NULL    NULL    NULL    .    NULL    NULL    NULL    1    .    PUNC

یہاں     ی    یہ    یہا    یہاں    NULL    NULL    NULL    ں    اں    ہاں    یہاں    4    یہ    DMD
مزہ      م    مز    مزہ    NULL    NULL    NULL    NULL    ہ    زہ    مزہ    NULL    3    مزہ    NN
Normal text file    length : 27,00,958   lines : 37,012    Ln : 1,109   Col : 97   Sel : 145 | 2    Windows (CR LF)   UTF-8   INS
```

Figure 9 MWE tagged Urdu corpora sample training file (Type-III)

Testing file: To test the raw corpora, we have designed the test corpora similar to training corpora without MWE tags such test_type1, test_type2 and test_type3. Using model file, we can predict the MWE tags for giving three test data files. Here, we are showing snapshot of test_type3 file and output file of test data (tested file) for Urdu language only as follows in *Figures 10* and *11*.


File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
hi.mwe_gold_5k.txt
hi.mwe_tested (type 1)
hi.mwe_tested (type 3).txt
hi.mwe_tested (type 2)
hi.mwe_tested_500.txt
ur.mwe_tested (type2).txt
ur.mwe_tested_featured_500.txt
ur.mwe_testing_type 3
Type here to search
ENG
IN

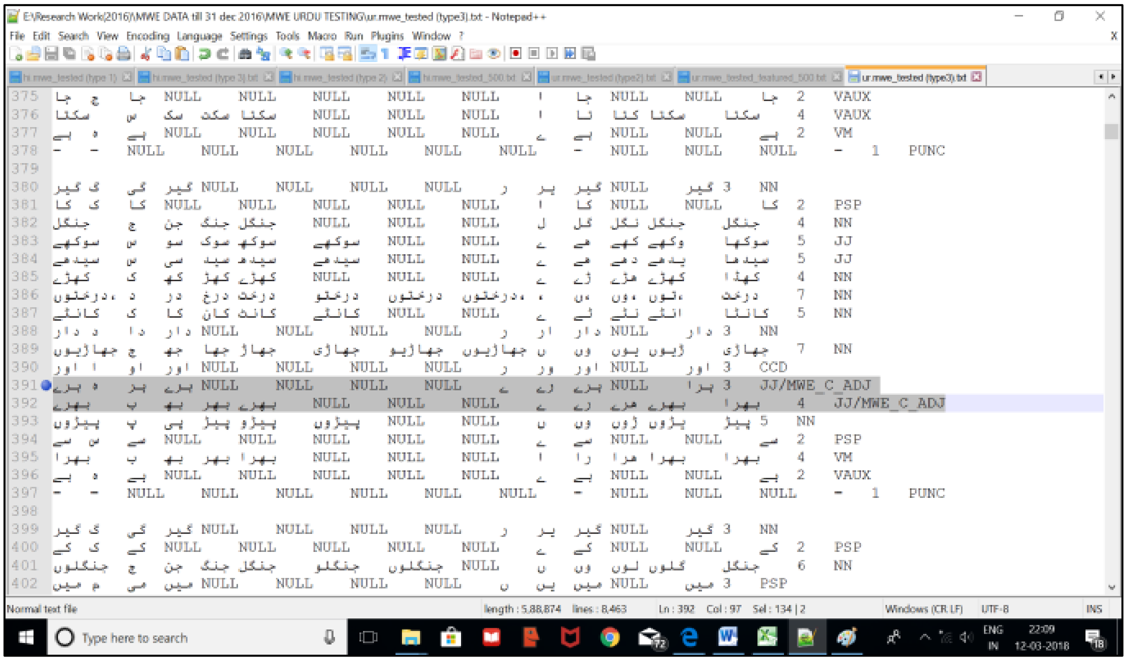
E:\Research Work(2016)\MWE DATA till 31 dec 2016\MWE URDU TESTING\ur.mwe_testing_type 3 - Notepad++

```
دیکھا	د	دی	دیک	دیکھ	دیکھا	NULL	NULL	ا	ھا	کھا	یکھا	دیکھ	5
جا	ج	جا	NULL	NULL	NULL	NULL	NULL	ا	جا	NULL	NULL	جا	2
سکتا	س	سک	سکت	سکتا	NULL	NULL	NULL	ا	تا	کتا	سکتا	سکتا	4
ہے	ہ	ہے	NULL	NULL	NULL	NULL	NULL	ے	ہے	NULL	NULL	ہے	2
-	-	NULL	NULL	NULL	NULL	NULL	NULL	-	NULL	NULL	NULL	-	1

کیر	ک	کی	کیر	NULL	NULL	NULL	NULL	ر	یر	کیر	NULL	کیر	3
کا	ک	کا	NULL	NULL	NULL	NULL	NULL	ا	کا	NULL	NULL	کا	2
جنگل	ج	جن	جنگ	جنگل	NULL	NULL	NULL	ل	گل	نگل	جنگل	جنگل	4
سوکھے	س	سو	سوک	سوکھ	سوکھے	NULL	NULL	ے	ھے	کھے	وکھے	سوکھا	5
سیدھے	س	سی	سید	سیدھ	سیدھے	NULL	NULL	ے	ھے	دھے	یدھے	سیدھا	5
کھڑے	ک	کھ	کھڑ	کھڑے	NULL	NULL	NULL	ے	ڑے	ھڑے	کھڑے	کھڑا	4
درختوں،	د	در	درخ	درخت	درختو	درختوں	درختوں،	،	ں،	وں،	توں،	درخت	7
کانٹے	ک	کا	کان	کانٹ	کانٹے	NULL	NULL	ے	ٹے	نٹے	انٹے	کانٹا	5
دار	د	دا	دار	NULL	NULL	NULL	NULL	ر	ار	دار	NULL	دار	3
جھاڑیوں	ج	جھ	جھا	جھاڑ	جھاڑی	جھاڑیو	جھاڑیوں	ں	وں	یوں	ڑیوں	جھاڑی	7
اور	ا	او	اور	NULL	NULL	NULL	NULL	ر	ور	اور	NULL	اور	3
ہرے	ہ	ہر	ہرے	NULL	NULL	NULL	NULL	ے	رے	ہرے	NULL	ہرا	3
بھرے	ب	بھ	بھر	بھرے	NULL	NULL	NULL	ے	رے	ھرے	بھرے	بھرا	4
پیڑوں	پ	پی	پیڑ	پیڑو	پیڑوں	NULL	NULL	ں	وں	ڑوں	یڑوں	پیڑ	5
سے	س	سے	NULL	NULL	NULL	NULL	NULL	ے	سے	NULL	NULL	سے	2
بھرا	ب	بھ	بھر	بھرا	NULL	NULL	NULL	ا	را	ھرا	بھرا	بھرا	4
ہے	ہ	ہے	NULL	NULL	NULL	NULL	NULL	ے	ہے	NULL	NULL	ہے	2
-	-	NULL	NULL	NULL	NULL	NULL	NULL	-	NULL	NULL	NULL	-	1

کیر	ک	کی	کیر	NULL	NULL	NULL	NULL	ر	یر	کیر	NULL	کیر	3
کے	ک	کے	NULL	NULL	NULL	NULL	NULL	ے	کے	NULL	NULL	کے	2
جنگلوں	ج	جن	جنگ	جنگل	جنگلو	جنگلوں	NULL	ں	وں	لوں	گلوں	جنگل	6
```

Normal text file   length : 5,80,315   lines : 8,463   Ln : 391   Col : 74   Sel : 0 | 0   Windows (CR LF)   UTF-8   INS

21:59 12-03-2018

Figure 10 Sample MWE testing file (Type-III) of Urdu corpora for CRF++


File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
hi.mwe_tested (type 1)
hi.mwe_tested (type 3).txt
hi.mwe_tested (type 2)
hi.mwe_tested_500.txt
ur.mwe_tested (type2).txt
ur.mwe_tested_featured_500.txt
ur.mwe_tested (type3).txt
Type here to search
ENG
IN

E:\Research Work(2016)\MWE DATA till 31 dec 2016\MWE URDU TESTING\ur.mwe_tested (type3).txt - Notepad++

```
جا	ج	جا	NULL	NULL	NULL	NULL	NULL	ا	جا	NULL	NULL	جا	2	VAUX
سکتا	س	سک	سکت	سکتا	NULL	NULL	NULL	ا	تا	کتا	سکتا	سکتا	4	VAUX
ہے	ہ	ہے	NULL	NULL	NULL	NULL	NULL	ے	ہے	NULL	NULL	ہے	2	VM
-	-	NULL	NULL	NULL	NULL	NULL	NULL	-	NULL	NULL	NULL	-	1	PUNC

کیر	ک	کی	کیر	NULL	NULL	NULL	NULL	ر	یر	کیر	NULL	کیر	3	NN
کا	ک	کا	NULL	NULL	NULL	NULL	NULL	ا	کا	NULL	NULL	کا	2	PSP
جنگل	ج	جن	جنگ	جنگل	NULL	NULL	NULL	ل	گل	نگل	جنگل	جنگل	4	NN
سوکھے	س	سو	سوک	سوکھ	سوکھے	NULL	NULL	ے	ھے	کھے	وکھے	سوکھا	5	JJ
سیدھے	س	سی	سید	سیدھ	سیدھے	NULL	NULL	ے	ھے	دھے	یدھے	سیدھا	5	JJ
کھڑے	ک	کھ	کھڑ	کھڑے	NULL	NULL	NULL	ے	ڑے	ھڑے	کھڑے	کھڑا	4	NN
درختوں،	د	در	درخ	درخت	درختو	درختوں	درختوں،	،	ں،	وں،	توں،	درخت	7	NN
کانٹے	ک	کا	کان	کانٹ	کانٹے	NULL	NULL	ے	ٹے	نٹے	انٹے	کانٹا	5	NN
دار	د	دا	دار	NULL	NULL	NULL	NULL	ر	ار	دار	NULL	دار	3	NN
جھاڑیوں	ج	جھ	جھا	جھاڑ	جھاڑی	جھاڑیو	جھاڑیوں	ں	وں	یوں	ڑیوں	جھاڑی	7	NN
اور	ا	او	اور	NULL	NULL	NULL	NULL	ر	ور	اور	NULL	اور	3	CCD
ہرے	ہ	ہر	ہرے	NULL	NULL	NULL	NULL	ے	رے	ہرے	NULL	ہرا	3	JJ/MWE_C_ADJ
بھرے	ب	بھ	بھر	بھرے	NULL	NULL	NULL	ے	رے	ھرے	بھرے	بھرا	4	JJ/MWE_C_ADJ
پیڑوں	پ	پی	پیڑ	پیڑو	پیڑوں	NULL	NULL	ں	وں	ڑوں	یڑوں	پیڑ	5	NN
سے	س	سے	NULL	NULL	NULL	NULL	NULL	ے	سے	NULL	NULL	سے	2	PSP
بھرا	ب	بھ	بھر	بھرا	NULL	NULL	NULL	ا	را	ھرا	بھرا	بھرا	4	VM
ہے	ہ	ہے	NULL	NULL	NULL	NULL	NULL	ے	ہے	NULL	NULL	ہے	2	VAUX
-	-	NULL	NULL	NULL	NULL	NULL	NULL	-	NULL	NULL	NULL	-	1	PUNC

کیر	ک	کی	کیر	NULL	NULL	NULL	NULL	ر	یر	کیر	NULL	کیر	3	NN
کے	ک	کے	NULL	NULL	NULL	NULL	NULL	ے	کے	NULL	NULL	کے	2	PSP
جنگلوں	ج	جن	جنگ	جنگل	جنگلو	جنگلوں	NULL	ں	وں	لوں	گلوں	جنگل	6	NN
میں	م	می	میں	NULL	NULL	NULL	NULL	ں	یں	میں	NULL	میں	3	PSP
```

Normal text file   length : 5,88,874   lines : 8,463   Ln : 392   Col : 97   Sel : 134 | 2   Windows (CR LF)   UTF-8   INS

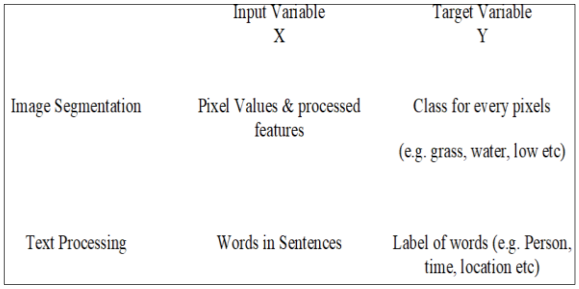
22:09 12-03-2018

Figure 11 Sample tested file (Type-III) of MWE tagged Urdu corpora given by CRF++

Both Hindi and Urdu corpora were used for training and testing. After the automatic extraction of MWE tags, assignments of MWE tags are replaced on raw tagged corpora. Then evaluation of above algorithms has been performed in the next section.

B. Conditional random field (CRF++) Model:

CRF is a sequential data labelling approach based on statistical modelling. It's a graphical model with no instructions for selecting the data sequence. CRF++ [41] is an example of a single exponential model or tool. In CRF++, we can use the previous tag, current tag, and future tag window frames at the same time. As a result, the probability of a certain tag can be estimated simultaneously with respect to previous and future tags. In general, CRF++ can be considered as a set of factors (features), which represent the relation between a number of variables. Suppose, we have input variable and target variable for Image segmentation and text processing as the format in *Figure 12*:



	Input Variable X	Target Variable Y
Image Segmentation	Pixel Values & processed features	Class for every pixels (e.g. grass, water, low etc)
Text Processing	Words in Sentences	Label of words (e.g. Person, time, location etc)

Figure 12 Sample Dataset for CRF++ model

Through the *Figure 12*, we can see that text can have two variables such as words and their labels. To present the relation between words and tags, we require any model. In our research work, we are dealing with CRF++ for text processing. Therefore, we are taking an example of POS tagging to understand the working of conditional random field model which is shown in *Figure 13.*

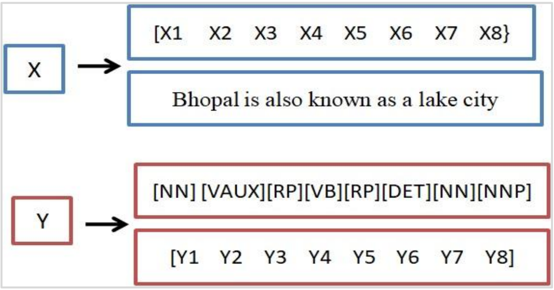

X
[X1 X2 X3 X4 X5 X6 X7 X8}
Bhopal is also known as a lake city
Y
[NN] [VAUX][RP][VB][RP][DET][NN][NNP]
[Y1 Y2 Y3 Y4 Y5 Y6 Y7 Y8]

Figure 13 Example of POS tagging for CRF++ model

In *Figure 13*, if input variable (X) are words and want to predict the class of word as a target variable. Then, one way to predict the class of word is to calculate the probability distribution in CRF++ model as follows:

Model the conditional distribution = P(Y | X)
To predict the sequence, compute:
Y^{*} =arg max P (Y | X)

$\downarrow$

Y^{*} must be able to compute it efficiently.

Feature functions for CRF:
Feature functions can be calculated by using neighbour information as like the following *Figure 14.*

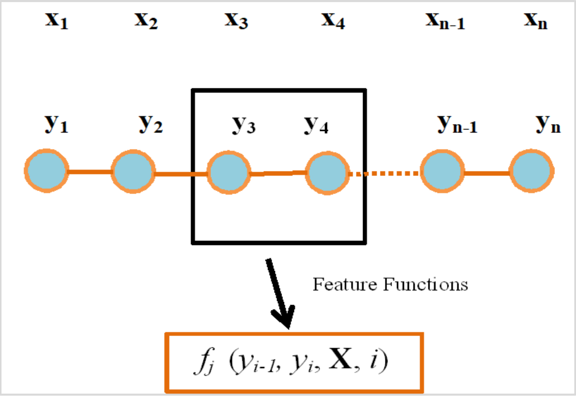

x_1 x_2 x_3 x_4 x_{n-1} x_n
y_1 y_2 y_3 y_4 y_{n-1} y_n
Feature Functions
$f_j\ (y_{i-1}, y_i, \mathbf{X}, i)$

Figure 14 Calculating feature functions in CRF model

In *Figure14*, label sequence model is a normalized product of feature functions. These features such as x & y are presented in Equation 1 and Equation 2 as follows.

$$P\ (y \mid x, \lambda) = \frac{1}{Z(x)} \exp \sum_{i=1}^{n} \sum \lambda_j f_j(y_{i-1}, y_i, \mathbf{x}, i) \qquad (1)$$

Where

$$Z\ (x) = \sum_{y \in Y} \sum_{i=1}^{n} \sum \lambda_j f_j(y_{i-1}, y_i, \mathbf{x}, i) \qquad (2)$$

The CRF++ model represents as a log-linear on the feature functions. By above formulae, we can estimate the expected level of target values.

CRF++, on the other hand, can train a significantly larger number of label distributions for two reasons:

- CRF++ allows you to define a vast number of features: Unlike HMMs, which are forced to be local (due to binary transition and emission transition functions that force each word to depend only on the current label and each label to depend only on the previous label), CRF++ can use a wide range of global characteristics.
- For learning, CRF++ has an arbitrary weight. As a result, we should understand how to learn feature weights using CRF++. One technique is to employ the gradient descent method.

If we've trained our CRF++ model for POS tagging, and then a fresh sentence is presented as an input, how can we label it? Then, the naive approach is to calculate p(l|s) for each possible labelling l, then choose the label with the highest probability. There are km possible labels for a tagset of size 'k' and a phrase of length 'm' because there are km possible labels for a tagset of size 'k' and a sentence of length 'm'. This method would need checking for an infinite number of labels.

Finally, realising that CRF++ has an optimal value enables us to utilise a dynamic programming technique to discover the most accurate and optimal label or tag. In our research work, we used the MWE tagged corpora to train the machine using CRF++ model for automatic extraction of MWEs. For training purpose, we employed 55k sentence of MWE tagged corpora.

3.4Phase-4(Results & Performance Evaluation)

The list of MWE tags were produced as an output by the help of proposed system. The list generated of MWE tags contains compound words of noun, verb, adjective and adverb as a collocation. Some collocations were not valid then replacing of incorrect tags has been performed on MWE tag list. Finally list of valid MWEs were created and according to this list, the MWEs on the tagged raw corpora were assigned. Also, the set of unique MWEs were designed. Numbers of generated MWEs are shown in following *Table 5*.

Table 5 No. of Generated MWEs by proposed algorithm and some are collected manually

S. No.	**Bigrams/MWE**	**Hindi language**		**Urdu language**	
		Our corpus	**Database (Unique Entity/ Non-Repetitive data)**	**Our Corpus**	**Database (Unique Entity/ Non-Repetitive data)**
1	Total No. of MWE_C_N	1,55,051	646	1,71,093	610
2	Total No. of MWE_C_V	705	88	788	77
3	Total No. of MWE_C_ADJ	37,583	511	36,533	628
4	Total No. of MWE_C_ADV	1,135	419	1,164	339
5	Total No. of MWE_RP	1322	42	1,465	36
6	Total No. of MWE_ECH	22	13	26	9
7	Total No. of MWE_A	1385	59	848	45
8	Total No. of MWE_IP	3394	3394	2807	2807
9	Total No. of MWE_S	490	96	NA	NA

Human evaluation:

The performance of the proposed system is evaluated by asking a human expert to identify MWEs from a test corpus of 500 sentences [42]. Same sentences were given to the multiword expression extraction algorithm. After this the precision, recall and f-measure were calculated by using the Equation 3, 4 and 5.

$$\text{Precision} = \frac{\#\text{Matches}}{\#\text{MachinegeneratedOutput}} \quad (3)$$

$$\text{Recall} = \frac{\#\text{Matches}}{\#\text{ActualtotalOutput}} \quad (4)$$

$$\text{F} - \text{Measure} = \frac{2\times\text{P}\times\text{R}}{\text{P}+\text{R}} \quad (5)$$

Here, precision was calculated by the total no. of output matches between human and machine divided by the total no. of MWE identified by machine. Recall was computed by the total no. of output matches between human and machine divided by the total no. of MWEs identified by the human expert and finally, f-measure was calculated by a combination of precision and recall. The accuracy of system can be measured on the basis of these following evaluation parameters. The accuracy of system is shown in below *Table 6* and *Table 7* for Hindi and Urdu respectively.

Table 6 Human evaluation of complete system for Hindi

Parameters		Training File (Type-I)	Training File (Type-II)	Training File (Type-III)
No. of Sentences as an Input		500	500	500
Machine generated MWE		294	316	343
Human generated MWE	Human Expert(H1)	402	402	402
	Human Expert(H2)	391	391	391
	Human Expert (H3)	407	407	407
Matched MWEs		198	244	278
H1	Precision	67.34	77.21	81.04
	Recall	49.25	60.69	69.15
	F-measure	56.89	67.96	74.61
H2	Precision	67.34	77.21	81.04
	Recall	50.63	62.40	71.09
	F-measure	57.80	69.01	75.73
H3	Precision	67.34	77.21	81.04
	Recall	48.64	59.95	68.30
	F-measure	56.48	67.49	74.12

Table 7 Human evaluation of complete system for Urdu

Parameters		Training File (Type-I)	Training File (Type-II)	Training File (Type-III)
No. of Sentences as an Input		500	500	500
Machine generated MWE		294	316	343
Human generated MWE	Human Expert(H1)	411	411	411
	Human Expert(H2)	397	397	397
	Human Expert (H3)	403	403	403
Matched MWEs		202	251	267
H1	Precision	68.70	79.43	77.84
	Recall	49.14	61.07	64.96
	F-measure	57.29	69.05	70.81
H2	Precision	68.70	79.43	77.84
	Recall	50.88	63.22	67.25
	F-measure	58.46	70.40	72.65
H3	Precision	68.70	79.43	77.84
	Recall	50.12	62.28	66.25
	F-measure	57.95	69.81	71.57

Automatic evaluation:

In this research work, automatic evaluation is done for the purpose of checking the performance of the system. For automatic evaluation, a confusion matrix is created. Through the confusion matrix, we obtained the precision, recall, f-measure and accuracy of each and every MWE tags. For development of confusion matrix, two files, one of gold data and another of test data with 1,000 sentences were considered and compared by calculating the exact matched and unmatched tags. Gold data means correctly tagged data by human expert which is used for mapping with machine generated data for evaluation purpose. The *Table 8* and *Table 9* and *Figure 15* and *Figure 16* illustrates the precision, recall, f-measure and accuracy of the MWE tags for Hindi and Urdu language consecutively.

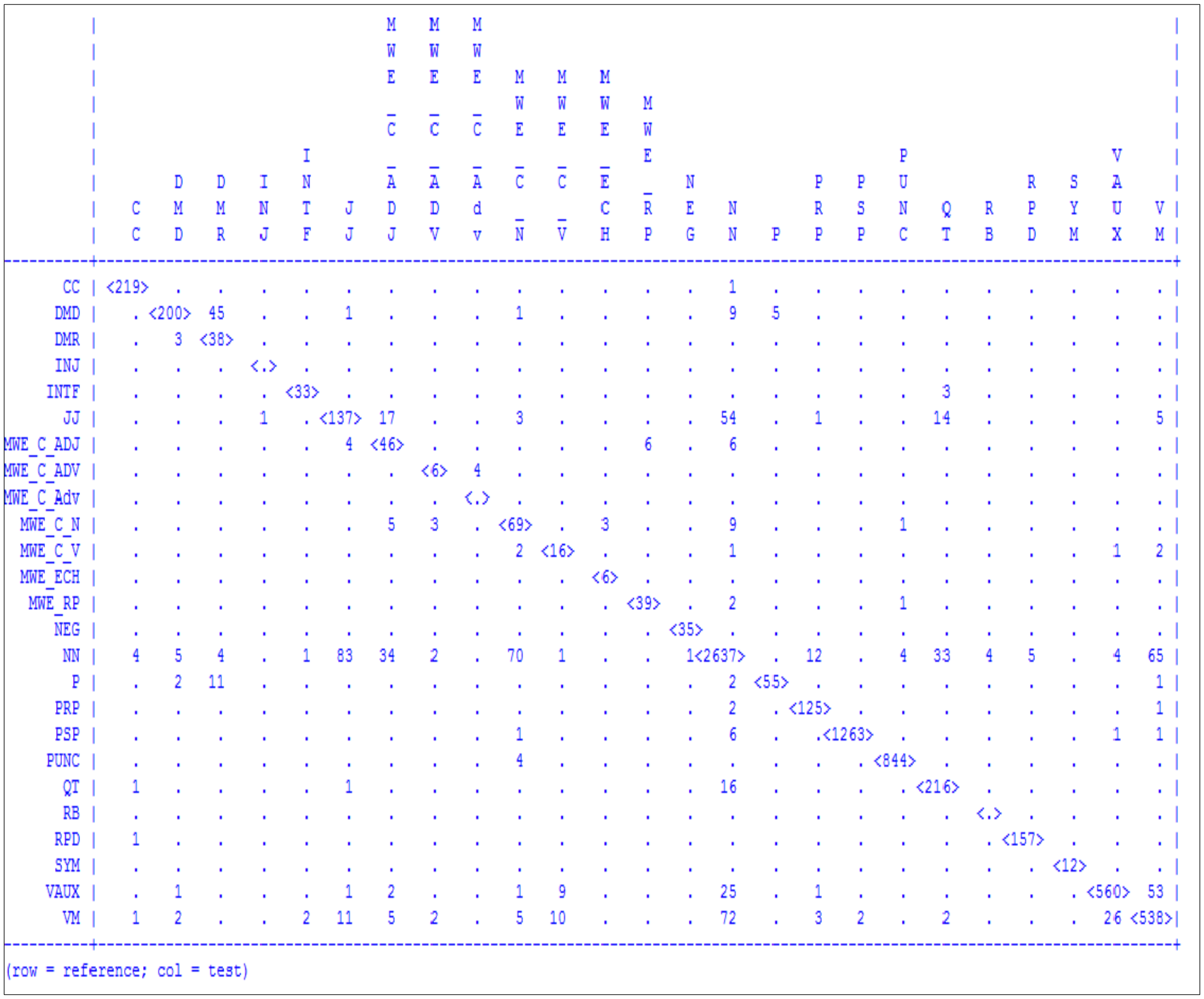
Here we are designing confusion matrix for tested file of type-III only, because we have seen that type-III tested file produces maximum accuracy score in human evaluation method. Therefore, automatic evaluation is also performed for only type-III tested data.

Evaluation of Hindi MWE tags:

Table 8 Evaluation of Hindi MWE tags using confusion matrix

MWE Tags	Precision	Recall	F-measure	Accuracy
MWE_C_N	0.9718	0.862	0.9139	0.9353
MWE_C_V	1.0	0.888	0.9411	0.9898
MWE_C_ADJ	0.9019	0.884	0.8932	0.9444
MWE_C_ADV	0.7692	1.0	0.8695	0.9851
MWE_C_S	1.0	1.0	1.0	1.0
MWE_RP	0.8666	1.0	0.9285	0.9696
MWE_ECH	0.6666	1.0	0.80	0.9850
MWE_A	1.0	1.0	1.0	1.0

MWE Tags	Precision	Recall	F-measure	Accuracy
MWE_IP	1.0	1.0	1.0	1.0
Overall System	0.9073	0.907	0.9073	0.9682



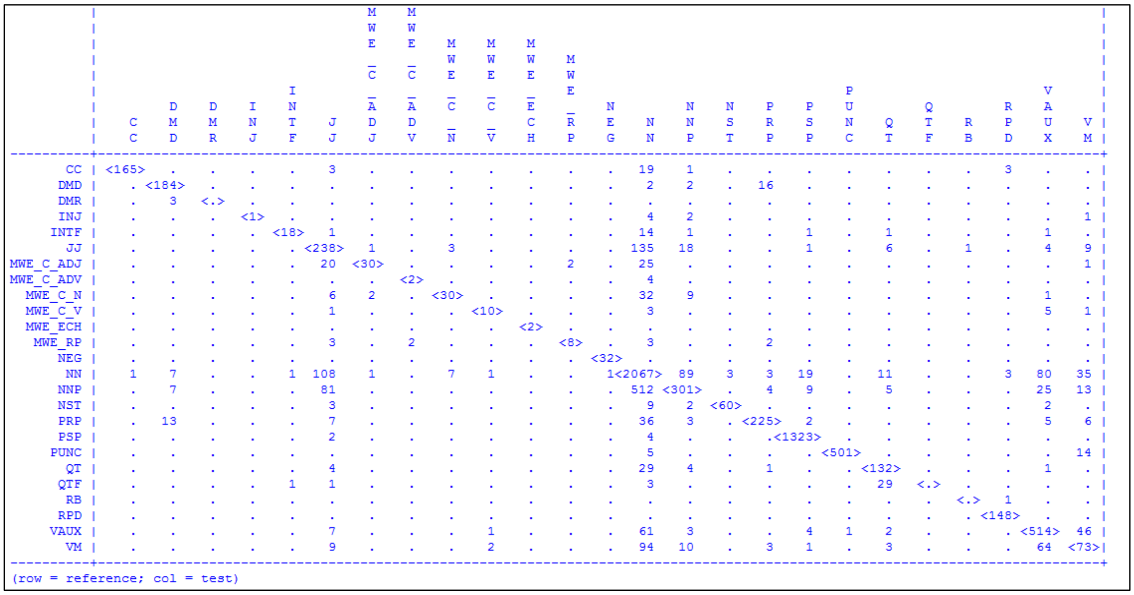
```
          |                                  M    M    M                                                                                 |
          |                                  W    W    W                                                                                 |
          |                                  E    E    E    M    M    M                                                                  |
          |                                  _    _    _    W    W    W    M                                                             |
          |                                  C    C    C    E    E    E    W                                                             |
          |                        I         _    _    _    _    _    _    E                             P                        V      |
          |        D    D    I    N         A    A    A    C    C    E    _    N              P    P    U              R    S    A      |
          |   C    M    M    N    T    J    D    D    d    _    _    C    R    E    N         R    S    N    Q    R    P    Y    U    V |
          |   C    D    R    J    F    J    J    V    v    N    V    H    P    G    N    P    P    P    C    T    B    D    M    X    M |
----------+------------------------------------------------------------------------------------------------------------------------------+
       CC | <219>   .    .    .    .    .    .    .    .    .    .    .    .    .    1    .    .    .    .    .    .    .    .    .    . |
      DMD |    . <200>  45    .    .    1    .    .    .    1    .    .    .    .    9    5    .    .    .    .    .    .    .    .    . |
      DMR |    .    3  <38>   .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    . |
      INJ |    .    .    .   <.>   .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    . |
     INTF |    .    .    .    .  <33>   .    .    .    .    .    .    .    .    .    .    .    .    .    .    3    .    .    .    .    . |
       JJ |    .    .    .    1    . <137>  17    .    .    3    .    .    .    .   54    .    1    .    .   14    .    .    .    .    5 |
MWE_C_ADJ |    .    .    .    .    .    4  <46>   .    .    .    .    .    6    .    6    .    .    .    .    .    .    .    .    .    . |
MWE_C_ADV |    .    .    .    .    .    .    .   <6>   4    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    . |
MWE_C_Adv |    .    .    .    .    .    .    .    .   <.>   .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    . |
  MWE_C_N |    .    .    .    .    .    .    5    3    .  <69>   .    3    .    .    9    .    .    .    1    .    .    .    .    .    . |
  MWE_C_V |    .    .    .    .    .    .    .    .    .    2  <16>   .    .    .    1    .    .    .    .    .    .    .    .    1    2 |
  MWE_ECH |    .    .    .    .    .    .    .    .    .    .    .   <6>   .    .    .    .    .    .    .    .    .    .    .    .    . |
   MWE_RP |    .    .    .    .    .    .    .    .    .    .    .    .  <39>   .    2    .    .    .    1    .    .    .    .    .    . |
      NEG |    .    .    .    .    .    .    .    .    .    .    .    .    .  <35>   .    .    .    .    .    .    .    .    .    .    . |
       NN |    4    5    4    .    1   83   34    2    .   70    1    .    .    1<2637>   .   12    .    4   33    4    5    .    4   65 |
        P |    .    2   11    .    .    .    .    .    .    .    .    .    .    .    2  <55>   .    .    .    .    .    .    .    .    1 |
      PRP |    .    .    .    .    .    .    .    .    .    .    .    .    .    .    2    . <125>   .    .    .    .    .    .    .    1 |
      PSP |    .    .    .    .    .    .    .    .    .    1    .    .    .    .    6    .    .<1263>   .    .    .    .    .    1    1 |
     PUNC |    .    .    .    .    .    .    .    .    .    4    .    .    .    .    .    .    .    . <844>   .    .    .    .    .    . |
       QT |    1    .    .    .    .    1    .    .    .    .    .    .    .    .   16    .    .    .    . <216>   .    .    .    .    . |
       RB |    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .   <.>   .    .    .    . |
      RPD |    1    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    . <157>   .    .    . |
      SYM |    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .  <12>   .    . |
     VAUX |    .    1    .    .    .    1    2    .    .    1    9    .    .    .   25    .    1    .    .    .    .    .    . <560>  53 |
       VM |    1    2    .    .    2   11    5    2    .    5   10    .    .    .   72    .    3    2    .    2    .    .    .   26 <538>|
----------+------------------------------------------------------------------------------------------------------------------------------+
(row = reference; col = test)
```

Figure 15 Confusion Matrix of Hindi MWEs evaluation

Evaluation of Urdu MWE tags:

Table 9 Evaluation of Urdu MWE tags using confusion matrix

MWE Tags	Precision	Recall	F-measure	Accuracy
MWE_C_N	1.0	0.789	0.913	0.914
MWE_C_V	1.0	1.0	0.941	1.0
MWE_C_ADJ	0.93	0.93	0.8932	0.954
MWE_C_ADV	0.50	1.0	0.869	0.977
MWE_RP	0.80	0.8	0.928	0.954
MWE_ECH	1.0	1.0	0.8	1.0
MWE_A	1.0	1.0	1.0	1.0
MWE_IP	1.0	1.0	1.0	1.0
Overall System	0.9318	0.872	0.901	0.9662



```
          |                                 M    M                                                                                     |
          |                                 W    W                                                                                     |
          |                                 E    E    M    M    M                                                                      |
          |                                 _    _    W    W    W    M                                                                 |
          |                                 C    C    E    E    E    W                                                                 |
          |                        I        _    _    _    _    _    E                                  P                        V      |
          |        D    D    I    N         A    A    C    C    E    _    N         N    N    P    P    U         Q         R    A      |
          |   C    M    M    N    T    J    D    D    _    _    C    R    E    N    N    S    R    S    N    Q    T    R    P    U    V |
          |   C    D    R    J    F    J    J    V    N    V    H    P    G    N    P    T    P    P    C    T    F    B    D    X    M |
----------+-----------------------------------------------------------------------------------------------------------------------------+
       CC |<165>   .    .    .    .    3    .    .    .    .    .    .    .   19    1    .    .    .    .    .    .    .    3    .    . |
      DMD |   . <184>   .    .    .    .    .    .    .    .    .    .    .    2    2    .   16    .    .    .    .    .    .    .    . |
      DMR |   .    3   <.>   .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    . |
      INJ |   .    .    .   <1>   .    .    .    .    .    .    .    .    .    4    2    .    .    .    .    .    .    .    .    .    1 |
     INTF |   .    .    .    .  <18>   1    .    .    .    .    .    .    .   14    1    .    .    1    .    1    .    .    .    1    . |
       JJ |   .    .    .    .    . <238>   1    .    3    .    .    .    .  135   18    .    .    1    .    6    .    1    .    4    9 |
MWE_C_ADJ |   .    .    .    .    .   20  <30>   .    .    .    .    2    .   25    .    .    .    .    .    .    .    .    .    .    1 |
MWE_C_ADV |   .    .    .    .    .    .    .   <2>   .    .    .    .    .    4    .    .    .    .    .    .    .    .    .    .    . |
  MWE_C_N |   .    .    .    .    .    6    2    .  <30>   .    .    .    .   32    9    .    .    .    .    .    .    .    .    1    . |
  MWE_C_V |   .    .    .    .    .    1    .    .    .  <10>   .    .    .    3    .    .    .    .    .    .    .    .    .    5    1 |
  MWE_ECH |   .    .    .    .    .    .    .    .    .    .   <2>   .    .    .    .    .    .    .    .    .    .    .    .    .    . |
   MWE_RP |   .    .    .    .    .    3    .    2    .    .    .   <8>   .    3    .    .    2    .    .    .    .    .    .    .    . |
      NEG |   .    .    .    .    .    .    .    .    .    .    .    .  <32>   .    .    .    .    .    .    .    .    .    .    .    . |
       NN |   1    7    .    .    1  108    1    .    7    1    .    .    1<2067>  89    3    3   19    .   11    .    .    3   80   35 |
      NNP |   .    7    .    .    .   81    .    .    .    .    .    .    .  512 <301>   .    4    9    .    5    .    .    .   25   13 |
      NST |   .    .    .    .    .    3    .    .    .    .    .    .    .    9    2  <60>   .    .    .    .    .    .    .    2    . |
      PRP |   .   13    .    .    .    7    .    .    .    .    .    .    .   36    3    . <225>   2    .    .    .    .    .    5    6 |
      PSP |   .    .    .    .    .    2    .    .    .    .    .    .    .    4    .    .    .<1323>   .    .    .    .    .    .    . |
     PUNC |   .    .    .    .    .    .    .    .    .    .    .    .    .    5    .    .    .    . <501>   .    .    .    .    .   14 |
       QT |   .    .    .    .    .    4    .    .    .    .    .    .    .   29    4    .    1    .    . <132>   .    .    .    1    . |
      QTF |   .    .    .    .    1    1    .    .    .    .    .    .    .    3    .    .    .    .    .   29   <.>   .    .    .    . |
       RB |   .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .   <.>   1    .    . |
      RPD |   .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    . <148>   .    . |
     VAUX |   .    .    .    .    .    7    .    .    .    1    .    .    .   61    3    .    .    4    1    2    .    .    . <514>  46 |
       VM |   .    .    .    .    .    9    .    .    .    2    .    .    .   94   10    .    3    1    .    3    .    .    .   64  <73>|
----------+-----------------------------------------------------------------------------------------------------------------------------+
(row = reference; col = test)
```

Figure 16 Confusion matrix of Urdu MWEs evaluation

4.Discussions with comparative analysis

Type-3 training file is compared to Type-2 and Type-1 training files in terms of Hindi and Urdu language. *Table 10* and *Table 11* shows the result of a comparative analysis.

According to *Table 10*, when compared to type-2 and type-1 training files, the most accurate training file is type-3, which contains 13 features. There are fewer features in type-2 and type-1 training files here than in type-3 training files. For the machine learning approach, it is better to have more features associated with the training file, as shown in below *Table 10.*

The CRF++ machine learning model has three different types of training files, and we've done the same for the Urdu language. Type-3 training files produce better accuracy, precision, recall, and f-measure parameters when also used with Urdu corpora. We can conclude that a training file with a large number of features is the most accurate, based on the below *Table 11.* A complete list of abbreviations is shown in *Appendix I.*

Table 10 Comparative analysis among 3 types of training files in Hindi

Evaluation Parameters	Type-1	Type-2	Type-3
Precision	0.6967	0.7431	0.9073
Recall	0.7235	0.7661	0.907
F-measure	0.7026	0.7541	0.9073
Accuracy	0.7334	0.8991	**0.9682**

Table 11 Comparative analysis among 3 types of training files in Urdu

Evaluation Parameters	Type-1	Type-2	Type-3
Precision	0.6551	0.7681	0.9318
Recall	0.6432	0.7497	0.872
F-measure	0.6532	0.7615	0.901
Accuracy	0.6973	0.7829	**0.9662**

There are some limitations to our proposed approach such as-

a) Approach is language specific; if we want to apply it to another language, we must identify the MWEs in that language in order to prepare a training file.

b) b) CRF++'s training file adheres to a specific format.

c) Every one of the training files is used to train the machine learning algorithm.

5.Conclusion and future work

In this research, we have discussed the identification and extraction of Hindi and Urdu language MWEs. It was difficult to extract MWE because of its unique composition and behaviour. Automated, semi-automatic, and manual techniques are used to identify MWE candidates. The vast majority of idiomatic expressions are compiled by hand. We used pre-processing to choose the only valid MWE after collecting all forms of multiword. These valid MWE tags were used to annotate the Hindi and Urdu corpora. MWE tags were used instead of POS tags (compound words or multiword) in the annotation. Annotated corpora can be used to improve MT by correctly predicting the combination of words.

This paper proposes a method for identifying and extracting MWEs from Indian languages like Hindi and Urdu. Pattern-based rules are used to identify the MWE in a corpus. MWE tagsets are made up of a list of all the MWEs that have been found. Later, new MWE tags were added to the previously-labeled datasets, which resulted in fresh datasets with new MWE tags. A training file for corpora in both languages was created using some features, and the CRF++ model was nourished this file as input to train the machine, which was then tested and evaluated. Comparing gold and test data files yielded a confusion matrix of 1,000 sentences each from the Hindi and Urdu languages for testing purposes. The system's 96.82 % accuracy in Hindi and 96.62 % accuracy in Urdu were calculated. The size of the training corpus can be expanded in the future in order to improve the efficiency and reliability of this system for automatically identifying and extracting MWEs. New tags like MWE_NE can be added in the future to expand on this work.

Acknowledgment

None.

Conflicts of interest

The authors have no conflicts of interest to declare.

Author’s contribution statement

Vaishali Gupta: Paper writing, designing of framework for MWE identification, experimental setup, comparative analysis. **Nisheeth Joshi:** Problem identification, concept designing, support to design framework, editing & review.

References

[1] De CHM, Ramisch C, Das GVNM, Villavicencio A. Alignment-based extraction of multiword expressions. Language Resources and Evaluation. 2010; 44(1):59-77.

[2] Constant M, Eryiğit G, Monti J, Van DPL, Ramisch C, Rosner M, et al. Multiword expression processing: a survey. Computational Linguistics. 2017; 43(4):837-92.

[3] Baldwin T, Kim SN. Multiword expressions. Handbook of Natural Language Processing. 2010; 2:267-92.

[4] Sag IA, Baldwin T, Bond F, Copestake A, Flickinger D. Multiword expressions: a pain in the neck for NLP. In international conference on intelligent text processing and computational linguistics 2002 (pp. 1-15). Springer, Berlin, Heidelberg.

[5] Nandi M, Ramasree R. Rule based extraction of multi-word expressions for elementary sanskrit texts. International Journal of Advanced Research in Computer Science. 2013; 3(11):661-7.

[6] Kumar S, Behera P, Jha GN. A classification-based approach to the identification of multiword expressions (MWEs) in magahi applying SVM. Procedia Computer Science. 2017; 112:594-603.

[7] Boroş T, Pipa S, Mititelu VB, Tufiş D. A data-driven approach to verbal multiword expression detection. PARSEME shared task system description paper. In proceedings of the 13th workshop on multiword expressions 2017 (pp. 121-6).

[8] Sinha RM. Stepwise mining of multi-word expressions in Hindi. In proceedings of the workshop on multiword expressions: from parsing and generation to the real world 2011 (pp. 110-5).

[9] Agrawal S, Sanyal R, Sanyal S. Hybrid method for automatic extraction of multiword expressions. International Journal of Engineering & Technology. 2018; 7(2.6):33-8.

[10] Majumder G, Pakray P, Khiangte Z, Gelbukh A. Multiword expressions (MWE) for Mizo language: literature survey. In international conference on intelligent text processing and computational linguistics 2016 (pp. 623-35). Springer, Cham.

[11] Singh D, Bhingardive S, Bhattacharyya P. Multiword expressions dataset for Indian languages. In proceedings of the tenth international conference on language resources and evaluation (LREC'16) 2016 (pp. 2331-5).

[12] Dandapat S, Mitra P, Sarkar S. Statistical investigation of Bengali noun-verb (NV) collocations as multi-word-expressions. Proceedings of Modeling and Shallow Parsing of Indian Languages (MSPIL). 2006:230-3.

[13] Attia M, Toral A, Tounsi L, Pecina P, Van GJ. Automatic extraction of Arabic multiword expressions. In proceedings of the 2010 workshop on multiword expressions: from theory to applications 2010 (pp. 19-27).

[14] Kulkarni N, Finlayson M. JMWE: a java toolkit for detecting multi-word expressions. In proceedings of the workshop on multiword expressions: from parsing and generation to the real world 2011 (pp. 122-4).

[15] Chakraborty T, Das D, Bandyopadhyay S. Identifying bengali multiword expressions using semantic

clustering. Lingvisticæ Investigationes. 2014; 37(1):106-28.

[16] Daoud D, Al-kouz A, Daoud M. Time-sensitive Arabic multiword expressions extraction from social networks. International Journal of Speech Technology. 2016; 19(2):249-58.

[17] Singh A, Jamwal SS. Identification, extraction and translation of multiword expressions. International Journal of Advanced Research in Computer Science and Software Engineering. 2016; 6(7):445-9.

[18] Joon R, Singhal A. Role of lexical and syntactic fixedness in acquisition of hindi MWEs. In international conference on advances in computing and data sciences 2019 (pp. 155-63). Springer, Singapore.

[19] Qasmi NH, Zia HB, Athar A, Raza AA. SimplifyUR: unsupervised lexical text simplification for Urdu. In proceedings of the 12th language resources and evaluation conference 2020 (pp. 3484-9).

[20] Han L, Jones GJ, Smeaton AF. MultiMWE: building a multi-lingual multi-word expression (MWE) parallel corpora. arXiv preprint arXiv:2005.10583. 2020.

[21] Fleischhauer J. Predicative multi-word expressions in persian. In proceedings of the 34th Pacific Asia conference on language, information and computation 2020 (pp. 552-61).

[22] Goyal KD, Goyal V. Development of hybrid algorithm for automatic extraction of multiword expressions from monolingual and parallel corpus of English and Punjabi. In proceedings of the 17th international conference on natural language processing (ICON): system demonstrations 2020 (pp. 4-6).

[23] Ramisch C, Savary A, Guillaume B, Waszczuk J, Candito M, Vaidya A, et al. Edition 1.2 of the PARSEME shared task on semi-supervised identification of verbal multiword expressions. In proceedings of the joint workshop on multiword expressions and electronic lexicons 2020 (pp. 107-18).

[24] Marszałek-kowalewska K. Discovery of multiword expressions with loanwords and their equivalents in the persian language. In proceedings of the international conference on recent advances in natural language processing 2021 (pp. 918-28).

[25] Tan KS, Lim TM, Tan CW. A study on multiword expression features in emotion detection of code-mixed twitter data. In international conference on artificial intelligence in engineering and technology (IICAIET) 2021 (pp. 1-5). IEEE.

[26] Han L, Jones GJ, Smeaton AF, Bolzoni P. Chinese character decomposition for neural MT with multi-word expressions. arXiv preprint arXiv:2104.04497. 2021.

[27] Jamwal SS, Gupta P, Sen VS. Multiword expression extraction using supervised ML for dogri language. In mobile radio communications and 5G networks 2022 (pp. 365-77). Springer, Singapore.

[28] Iwatsuki K, Boudin F, Aizawa A. Extraction and evaluation of formulaic expressions used in scholarly papers. Expert Systems with Applications. 2022.

[29] Muraki EJ, Abdalla S, Brysbaert M, Pexman PM. Concreteness ratings for 62 thousand English multiword expressions. Concreteness Ratings for Multiword Expressions. 2022.

[30] Nunsanga MV, Pakray P, Lalngaihtuaha M, Singh LLK. Stochastic based part of speech tagging in mizo language: unigram and bigram hidden markov model. In edge analytics 2022 (pp. 711-22). Springer, Singapore.

[31] Khan W, Daud A, Khan K, Nasir JA, Basheri M, Aljohani N, et al. Part of speech tagging in Urdu: comparison of machine and deep learning approaches. IEEE Access. 2019; 7:38918-36.

[32] Kaur J, Saini JR. A study of text classification natural language processing algorithms for Indian languages. VNSGU Journal of Science and Technology. 2015; 4(1):162-7.

[33] Gayen V, Sarkar K. A machine learning approach for the identification of bengali noun-noun compound multiword expressions. arXiv preprint arXiv:1401.6567. 2014.

[34] Sing S, Jha GN. English multi-word expressions (MWE): a tagset for health domain. In international conference on advances in computing, communications and informatics (ICACCI) 2018 (pp. 1812-7). IEEE.

[35] Venkatapathy S, Joshi A. Measuring the relative compositionality of verb-noun (VN) collocations by integrating features. In proceedings of human language technology conference and conference on empirical methods in natural language processing 2005 (pp. 899-906).

[36] Diab MT, Krishna M. Unsupervised classification of verb noun multi-word expression tokens. In international conference on intelligent text processing and computational linguistics 2009 (pp. 98-110). Springer, Berlin, Heidelberg.

[37] Bharati A, Sangal R, Mishra D, Venkatapathy S, Reddy TP. Handling multi-word expressions without explicit linguistic rules in an MT system. In international conference on text, speech and dialogue 2004 (pp. 31-40). Springer, Berlin, Heidelberg.

[38] Hu D. An introductory survey on attention mechanisms in NLP problems. In proceedings of SAI intelligent systems conference 2019 (pp. 432-48). Springer, Cham.

[39] Khan SA, Anwar W, Bajwa UI. Challenges in developing a rule based urdu stemmer. In proceedings of the 2nd workshop on south southeast asian natural language processing 2011 (pp. 46-51).

[40] Kansal R, Goyal V, Lehal GS. Rule based Urdu stemmer. In proceedings of COLING 2012: demonstration papers 2012 (pp. 267-76).

[41] Lafferty J, Mccallum A, Pereira FC. Conditional random fields: probabilistic models for segmenting and labeling sequence data. 2001.

[42] Shahnawaz, Mishra RB. Statistical machine translation system for English to Urdu. International Journal of Advanced Intelligence Paradigms. 2013; 5(3):182-203.

Vaishali Gupta completed her Ph.D in Computer Science & Engineering from Banasthali Vidyapith, Rajasthan, India. She is working as an Assistant Professor, IES, IPS Academy, Indore, MP. She got a young scientist award in computer science & engineering in the year of 2017 from MPCST, Bhopal, MP. She has an interest in language processing specifically for Indian Languages. She has developed various NLP tools for Hindi and Urdu language and also published more than 15 research papers. Her current research areas are Natural Language Processing, Information retrieval, Machine Learning, Deep Learning and Artificial Intelligence.
Email: Vaishali.gupta77@gmail.com

Nisheeth Joshi works as an Associate Professor at Banasthali University. He has published various research papers in reputed journal, conferences and book series. His areas of interest include Computational Linguistics, Natural Language Processing, and Artificial Intelligence. Besides this, he is also very actively involved in the development of MT engines for English to Indian languages. He is one of the experts empanelled with the TDIL program, Department of Information Technology, Goverment of India, a premier.
Email: Nisheeth.joshi@rediffmail.com

Appendix I

S. No.	Abbreviation	Description
1	API	Application Programming Interface
2	AM	Association Measures
3	CRF	Conditional Random Field
4	HMM	Hidden Markov Model
5	LLR	Log Likelihood Ratio
6	MT	Machine Translation
7	MWE	Multiword Expression
8	MWEs	Multiword Expressions
9	NLP	Natural Language Processing
10	PMI	Pointwise Mutual Information
11	POS	Part of Speech