International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 – 5161

Available at http://inpressco.com/category/ijcet

Research Article

Text Chunker for Punjabi

Ubeeka Jain†* and Jasbir Kaur†

†R.I.E.I.T , Railmajra ,Punjab, India

Accepted 30 Sept 2015, Available online 10 Oct 2015, **Vol.5, No.5 (Oct 2015)**

Abstract

Parsing is the process of assigning a parse tree to the sentence. There are many problems related to the process of full parsing. Shallow parsing or chunking is the alternative for full parsing. In chunking the phrases of the sentences are chunked together. Chunking is more efficient and robust as it takes less time and always gives a solution. It is often deterministic as it gives only one solution to a problem. Chunkers are used in a large no. of NLP applications. Such as information extraction, named entity recognition, spell checkers, search etc . Chunkers are relatively difficult to build for Indian languages as there arise many problems during the system development. Chunkers identify the noun or verb etc chunks. Chunks are the non-overlapping regions. In this work, first standardized text chunker for Punjabi language is built and the greedy based algorithm is used for the machine learning and training of data set.

Keywords: *Natural language Processing (NLP), Part of Speech Tagge r(POS), Punjabi chunker*

1. Introduction

In NLP Computers are used to understand and manipulate text and speech to do some useful work NLP is the branch of Computer science mainly dealing with developing of systems by which computers can interact with human using natural language . NLP includes various computational and analyzing processes which enable machine to understand the language. Punjabi is an **Indo-Aryan** language. It is the 10TH most spoken language in the world and native language of about 131 million people. Most of the Punjabi speaking people live in Punjab region of Pakistan and India. It is also spoken in Himachal Pradesh, Haryana and Delhi and many countries in abroad. Punjabi is written in two different scripts called **Gurmukhi** and **Shahmukhi.**

Some of the applications for NLP are Part of Speech tagging (POS), Question Answering system, Name Entity Recognition (NER), and Multiple Word Expression (MWE) etc. which are used in machine translation.

Chunking: chunking is the process of dividing the sentence into chunks. Chunks are the non-overlapping regions in a sentence. Chunks are correlated group of words(Abney *et al*,1991).

The phrase chunker divides the sentence into noun phrases or verb phrases. These phrases are grouped together i.e. all the verbs occurring in a sentence are chunked in a single chunk and all the noun phrases are grouped in another single chunk. There also exist adjective phrases and noun adverb phrases.(Anil K Singh *et al*, 2008)

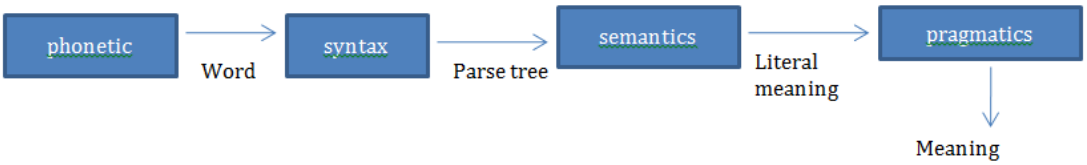
There are many levels of language analysis. These are shown in the following figure. The parsing phase lies in the syntax level of language analysis. Parsing is the process of generation of parse tree for a sentence.

Chunking is the alternative to parsing. There exists no complete grammar for any language. Ambiguity exists for many sentences. Ambiguity is the generation of more than one parse tree for one sentence. Full parsing takes a reasonable time for large amount of data. Chunking is more efficient and robust as it takes less time and always gives a solution. It is often deterministic as it gives only one solution to a problem. Context is Small and local. it can be applied to very large text resources i.e. web.(Kudo *et al* 2001)

The output of the chunker consists of series of non –overlapping regions that are also non recursive and do not contain each other. Thus the output of chunker is different from the parsing and it is easier as compared to parsing.

Rest of the paper is organized as follows the section 2 describes the applications of chunker. Section 3 contains the tagset for POS tagging and chunking. Section 4 briefs about corpus development. Section 5 consists of overview of framework. Section 6 briefs about system design and implementation. Section 7 contains testing and results. Section 8 concludes the conclusion.

*Corresponding author **Ubeeka Jain** is working as Assistant Professor **and Jasbir Kaur** is a M.Tech Scholar



2. Potential Applications

Chunkers are used as a resource component for many NLP applications.

A. *Information extraction*: the chunker divides the sentence into chunks of interrelated data. Noun phrase and verb phrase are chunked and can be used in information extraction systems. IE focuses on discovering names of people and events they participate in, from a document.
B. *Question Answering system*: the complete chunk can be used as the answer of the question asked. question-answering provides the user with either just the text of the answer itself or answer-providing passages.
C. *Spell Checkers*: checks the wrongly typed words within the sentence.
D. *Named entity identification*: in this system the main aim is to identify the particular words in the document. Such as people , places and other nouns in the sentence.
E. *Search*: searching of a particular noun or verb can be done. As the sentence is chunked in pieces, search becomes an easy task and the whole chunk can be represented as the search result
F. *Machine translation*: machine translation is the process of translating one language into another language. Chunking is useful in this task as the chunks are converted into another language.

3. Tagset for Pos Tagging Aand Chunking

POS tag set used in development of this chunker is the standard tagset given by TDIL for Punjabi language. There are 35 standard tags for Punjabi (TDIL).

Table 1 Tagset for Parts of Speech Tagging

No.	Tag	Tag Description
1	N_NN	Common Noun
2	N_NNP	Proper Noun
3	N_NST	Noun loc
4	PR_PRP	Personal Pronoun
5	PR_PRF	Reflexive Pronoun
6	PR_PRL	Relative Pronoun
7	PR_PRC	Reciprocal Pronoun
8	PR_PRQ	Wh-word Pronoun
9	PR_PRI	Indefinite
10	DM_DMD	Deictic Demonstrative
11	DM_DMR	Relative Demonstrative
12	DM_DMQ	Wh-word Demonstrative
13	DM_DMI	indefinite Demonstrative
14	V_VM	Main Verb
15	V_VM_VNF	Non-finite Verb
16	V_VM_VINF	Infinitive Verb
17	V_VM_VNG	Gerund Verb
18	V_VAUX	Auxiliary Verb
19	JJ	Adjective
20	RB	Adverb
21	PSP	Postposition
22	CC_CCD	Co-ordinator
23	CC_CCS	Subordinator
24	RP_RPD	Default Particles
25	RP_INJ	Interjection Particles
26	RP_INTF	Intensifier Particles
27	RP_NEG	Negation
28	QT_QTF	General
29	QT_QTC	Cardinals
30	QT_QTO	Ordinals
31	RD_RDF	Foreign word Residuals
32	RD_SYM	Symbol Residuals
33	RD_PUNC	Punctuation
34	RD_UNK	Unknown
35	RD_ECH	Echo-words

For Chunking, mainly seven tags are used. This is based on the grammatical or the syntactical category. The chunks are represented in square brackets and the right hand side contains the head naming the chunk.

Table 2 Tagset for Chunking

No.	Chunk	Chunk Description
1	_NP	Noun chunk
2	_CCP	Conjunction chunk
3	_VGF	Verb chunk
4	_RBP	Adverb chunk
5	_JJP	Adjective chunk
6	_VGINF	Verb infinite
7	_BLK	Bulk phrase

The guidelines mentioned in tagset given by the TDIL are followed for chunking. Seven chunks are used. First is the noun phrase chunk. It is given the tag _NP and the head is noun. Examples of noun chunk are:

- [[ਅੰਤਰਰਾਸ਼ਟਰੀ\N_NN ਯੋਗ\N_NN ਦਿਵਸ\N_NN ਦੇ\PSP ਮੌਕੇ\N_NN]]_NP
- [[ਹੋਰ\QT_QTF ਮੰਤਰਾਂ\N_NN ਦੇ\PSP ਉਚਾਰਨ\N_NN ਨੂੰ\PSP]]_NP

The conjuction chunk is tagged as _CCP. Conjunctions are the words used to join phrases, words, clauses. The example is:

- [[ਅਤੇ\CC_CCD]]_CCP
- [[ਕਿ\CC_CCS]]_CCP

Verb chunks are classified as verb chunk denoted by_ VGF and infinite verb chunk denoted by _VGINF. The examples are:

- [[ਲਿਖ\V_VM ਕੇ\V_VM_VNF]]_VGF
- [[ਉਲੰਘਣਾ\N_NN ਹੋਈ\V_VM_VF]]_VGF
- [[ਲੈਣਾ\V_VM_VNF]]_VGINF
- [[ਮੁੜਦੇ-ਮੁੜਦੇ\V_VM_VINF]]_VGINF

Adverb chunks are denoted by _RBP. These are tagged in accordance with the tagset of POS. the example is:

- [[ਪਰ\CC_CCS ਅੱਜ\RB]]_RBP
- [[ਕਰਨ\V_VM_VNF ਸ਼ੁਰੂ\RB]]_RBP

Adjective chunks are given the tag _JJP. This includes all the adjective chunks. The example is:

- [[ਉਪਰੰਤ\PSP ਮੈਡੀਕਲ\JJ]]_JJP
- [[ਖਤਰੇ\JJ ਤੋਂ\PSP ਖਾਲੀ\JJ]]_JJP

In Bulk phrase all the miscellaneous data is given the tag _BLK. The example is:

- [[ਦੇਣ\V_VM_VNF ਨਾਲ\PSP ਸੰਵਿਧਾਨ \N_NN ਦੀ\PSP]]_BLK
- [[ਕਰਵਾਇਆ\V_VM_VF।\RD_PUNC]]_BLK

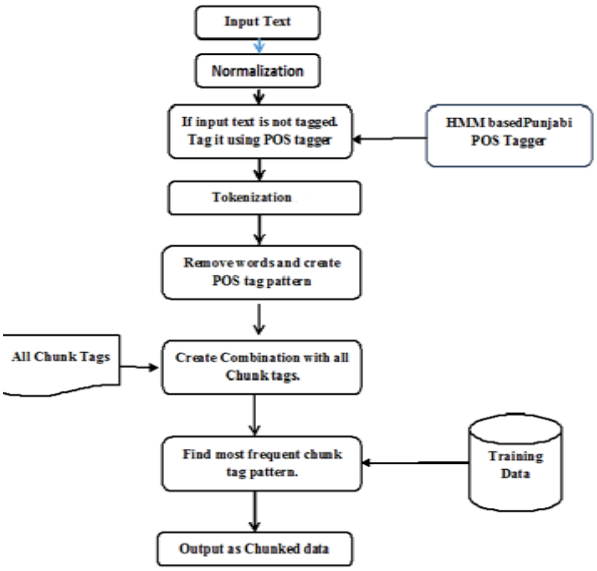
4. Corpus Development

Corpus is developed for training and testing of the system. The training data contains one thousand sentences of Punjabi which are tagged using the already developed HMM based POS tagger for Punjabi and then manually chunking of the corpus. This chunked corpus is given for training of the system using machine learning tools. The data is collected from various sources like online news, stories, newspaper articles etc. The sample of training data is as follows:

- [[ਖਤਰਾ\N_NN]]_NP [[ਟਲਣ\V_VM_VNF]]_VGF [[ਤੇ\PSP]]_BLK [[ਬੈਬੂਨ\N_NNP ਦਰੱਖਤ\N_NN ਤੋਂ\PSP]]_NP [[ਹੇਠਾਂ\RB]]_RBP [[ਉੱਤਰ\V_VM ਆਉਂਦੇ\V_VM_VF ਹਨ\V_VAUX]]_VGF [[।\RD_PUNC]]_BLK
- [[ਮਨਆਰਾ\N_NNP ਪਾਰਕ\N_NNP ਵਿਚ\PSP]]_NP [[ਇਹ\DM_DMD ਬੈਬੂਨ\N_NNP ,\RD_PUNC ਦਰੱਖਤਾਂ\N_NN]]_NP [[ਉੱਤੇ\RB]]_RBP [[ਸੌਂ\V_VM ਰਹੇ\V_VM_VF ਸਨ\V_VAUX]]_VGF [[।\RD_PUNC]]_BLK

The training data format is as above. The chunk is represented in double square brackets and at the right side the tag represented the chunk is written.

5. Overview of Framework



The design of the chunker is as described in the flowchart. A sketchy idea is described below that how the input text is processed and the output is given in the form of chunked data.

For the chunking of the raw text, the input text is given to the chunker. Normalization of the text is done. In normalization unwanted chars from the input are removed and some formatting is added for further processing by the algorithm. If the input text is not tagged then POS tagging of the text is done using the already built HMM based POS tagger. The POS tagger tags the whole text into 35 standard tags. Then the tokenization of the sentences is done. The words from the tagged data are removed and the POS tag pattern is created. We concern only about the pattern of the tags for further processing. Then the combination with all the chunk tags is created. It is analyzed that which tag pattern correspond to which chunk. We have used seven tags in the system. Using the training data the most frequent chunk tag pattern is found and the input is given that chunk name.

6. System Design and Implementation

The chunking system is divided into two portions. First is training and the second is testing.

Training Process: first of all we have collected the training data. The training data is raw text collected from various sources which is first of all POS tagged. The chunks are identified and tagged in POS data. This training data is saved in a separate file. For the training process of the system machine learning approach is used. the words are removed and only the tag pattern is analyzed. The system checks the pattern and the chunk associated with it and makes a hash table for every pattern. Every tag pattern and the related chunk in the training data is saved in the directory along with the frequency of the occurrence of the pattern. The training file is saved in the memory as binary file.

Testing Process: during the testing process greedy based algorithm is used. when the POS tagged data is input to the system then the already trained system takes the POS tag pattern and checks the frequency of the pattern in the directory. After frequency analyses of the pattern in directory the most frequent chunk is found and the output as the chunked data is given. The system is implemented in Microsoft visual c#. for POS tagging of the data we have used the HMM based POS tagger already developed by Punjabi university.

Sample input and output: This section provides some sample Punjabi sentences given as the input to the system and output as chunked data is given by the system.

Input 1 :
ਅੰਤਰਰਾਸ਼ਟਰੀ ਯੋਗ ਦਿਵਸ ਦੇ ਮੌਕੇ ਉਪਰ ਰਾਜਪਥ 'ਤੇ ਆਯੋਜਿਤ ਪ੍ਰੋਗਰਾਮ ਦੌਰਾਨ 'ਓਮ' ਅਤੇ ਯੋਗ ਨਾਲ ਜੁੜੇ ਹੋਰ ਮੰਤਰਾਂ ਦੇ ਉਚਾਰਨ ਨੂੰ ਆਲ ਇੰਡੀਆ ਮੁਸਲਿਮ ਪ੍ਰਸਨਲ ਲਾਅ ਬੋਰਡ ਦੇ ਮੈਂਬਰ ਜਫ਼ਰਯਾਬ ਜਿਲਾਨੀ ਨੇ ਸੰਵਿਧਾਨ ਦੇ ਖਿਲਾਫ ਦੱਸਿਆ ਹੈ|

Input 2/ output 1(POS tagging):
ਅੰਤਰਰਾਸ਼ਟਰੀ\N_NN ਯੋਗ\N_NN ਦਿਵਸ\N_NN ਦੇ\PSP ਮੌਕੇ\N_NN ਉਪਰ\N_NN ਰਾਜਪਥ\N_NN 'ਤੇ\PSP ਆਯੋਜਿਤ\JJ ਪ੍ਰੋਗਰਾਮ\N_NN ਦੌਰਾਨ\RB 'ਓਮ'\N_NN ਅਤੇ\CC_CCD ਯੋਗ\N_NN ਨਾਲ\PSP ਜੁੜੇ\V_VM_VF ਹੋਰ\QT_QTF ਮੰਤਰਾਂ\N_NN ਦੇ\PSP ਉਚਾਰਨ\N_NN ਨੂੰ\PSP ਆਲ\N_NN ਇੰਡੀਆ\N_NNP ਮੁਸਲਿਮ\N_NNP ਪ੍ਰਸਨਲ\N_NN ਲਾਅ\N_NN ਬੋਰਡ\N_NN ਦੇ\PSP ਮੈਂਬਰ\N_NN ਜਫ਼ਰਯਾਬ\N_NN ਜਿਲਾਨੀ\N_NN ਨੇ\PSP ਸੰਵਿਧਾਨ\N_NN ਦੇ\PSP ਖਿਲਾਫ\N_NN ਦੱਸਿਆ\V_VM_VF ਹੈ\V_VAUX ।\RD_PUNC

Final output:
[[ਅੰਤਰਰਾਸ਼ਟਰੀ\N_NN ਯੋਗ\N_NN ਦਿਵਸ\N_NN ਦੇ\PSP ਮੌਕੇ\N_NN]]_NP [[ਉਪਰ\N_NN ਰਾਜਪਥ\N_NN 'ਤੇ\PSP]]_NP [[ਆਯੋਜਿਤ\JJ ਪ੍ਰੋਗਰਾਮ\N_NN]]_NP [[ਦੌਰਾਨ\RB 'ਓਮ' \N_NN]]_NP [[ਅਤੇ\CC_CCD ਯੋਗ\N_NN ਨਾਲ\PSP]]_NP [[ਜੁੜੇ\V_VM_VF]]_VGF [[ਹੋਰ\QT_QTF ਮੰਤਰਾਂ\N_NN ਦੇ\PSP ਉਚਾਰਨ\N_NN ਨੂੰ\PSP]]_NP [[ਆਲ\N_NN ਇੰਡੀਆ\N_NNP ਮੁਸਲਿਮ\N_NNP]]_NP [[ਪ੍ਰਸਨਲ\N_NN ਲਾਅ\N_NN ਬੋਰਡ\N_NN ਦੇ\PSP ਮੈਂਬਰ\N_NN]]_NP [[ਜਫ਼ਰਯਾਬ\N_NN ਜਿਲਾਨੀ\N_NN ਨੇ\PSP ਸੰਵਿਧਾਨ\N_NN ਦੇ\PSP ਖਿਲਾਫ\N_NN]]_NP [[ਦੱਸਿਆ\V_VM_VF ਹੈ\V_VAUX ।\RD_PUNC]]_VGF

The input given to the chunker is either raw data on which we done POS tagging using HMM based Punjabi tagger or already POS tagged data is input to the system.

7. Testing and Result

After training the system with chunked data we perform the testing of the system with raw data. The various formulas used in result are as follows:

Precision: P= $\frac{\text{No. of correct answers}}{\text{No. of answers given}}$

Recall: R= $\frac{\text{No. of correct answers given by the system}}{\text{Total No. of answers}}$

F-measure: F-measure is defined as balances of Recall and Precision by using a parameter ß

F-measure= $\frac{(ß+1)RP}{(ßP+R)}$

ß is weighted as ß=1

when ß=1, F-measure is called F1-measure

F1-measure= $\frac{2RP}{P+R}$

Following results were obtained while testing the raw corpus within the system. The raw corpus used for testing was in Unicode.

For training the system, ie for in the training phase, the chunker was trained with using about 1000 sentences. Increasing the accuracy of the system can increase this further to any extent there.
1000 is total no. of sentences for testing and 750 is correct answers given by system:

P = $\frac{750}{800}$ = .93 =93%
R=$\frac{750}{1000}$= .75= 75%
F-measure= $\frac{2(.75*.93)}{.75+.93}$ = 83%

Keeping into mind the fact that this is the first standard chunker, these results are considered as good.

Comparison with existing systems: With best of our knowledge there exist no chunker available for Punjabi which has used standardized POS tagset given by TDIL. There exist chunkers for other Indian languages. We compare our system wiih the existing systems. In 1995, Ramshaw and Marcus obtained a precision of 91.8% and a recall of 92.3% for base np chunks when trained on 200000 words(A. Ramshaw,P.Marcus *et al*,1995). Zhou in 2000 used the HMM method and achieved the recall and precision of 92.25 and 91.99 respectively(Zhou *et al*,2000). Jisha P Jayan and Rajeev R R got the results for malayalam chunker- Equal : 184/200 (92.00%) Different : 16/200 (8.00%) the system gives about 92% of accuracy (Jisha *et al*) . 95.82% of the accuracy is obtained by Dhanalakshmi for tamil chunker(Dhanalakshmi *et al*, 2009). 92.63% for chunk boundary identification task and 91.70% for

the composite task of chunk labeling with a recall of 100% is obtained by Akshey singh, sushma for hindi chunker(Akshay Singh *et al*,2005). The precision and recall rates of 96.12% and 98.03% are obtained by Dipanjan Das, Monojit Choudhury for Bengali language chunker(Dipanjan *et al*).

Conclusion

This paper presents the implementation and results for the chunking system of Punjabi. This system performs the chunking of Punjabi text into seven chunks. To the best of our knowledge it is the first chunker for Punjabi language based on standardized POS tagset given by TDIL. The development of Punjabi language is in first phase. This effort will reduce the gap of development of resources. Chunker is used as an essential tool for the further development of resources. This work will definitely motivate the future researchers for development in the area of Punjabi.

References

S. Abney(1991). *Parsing by chunks* In Berwick, Abney, and Tenny, editors,*Principle-Based Parsing.* Kluwer Academic Publishers.

Anil Kumar Singh, (2008) Language Technologies Research Centre, IIIT, Hydrabad India, NLP for Less Privileged Languages: Where do we come from? Where are we going? In IJCNLP Workshop on NLP

Taku Kudo and Yuji Matsumoto(2001) Chunking with Support Vector Machines. *Proceedings of NAACL 2001* (2001) 1013-1015

Unified Parts of Speech (POS) Standard in Indian Languages - Draft Standard –Version 1.0 Department of Information Technology Ministry Communications & Information Technology Govt. of India

Lance A. Ramshaw, and Mitchell P. Marcus. (1995) Text Chunking Using Transformation-Based Learning. *Proceedings of the 3rd Workshop on Very Large Corpora* (1995) 88-94

Zhou, GuoDong, Jian Su and TongGuan Tey (2000) Hybrid Text Chunking. *Proceedings of CoNLL- 2000 and LLL-2000* (2000) 163-165.

Jisha P Jayan, Rajeev R R Parts Of Speech Tagger and Chunker for Malayalam – Statistical Approach *Computer Engineering and Intelligent Systems* ISSN 2222-1719 (Paper) ISSN 2222-2863 (Online)

Dhanalakshmi V, Anandkumar M, Shivapratap G, Soman, K P, Rajendran S (2009). Tamil POS Tagging using Linear Programming, *In International Journal of Recent Trends inEngineering*, 1(2):166-169.

Akshay Singh, Sushma Bendre(2005) HMM Based Chunker for Hindi *in the Proceedings of IJCNLP-05: The Second International Joint Conference on NaturalLanguage Processing,* 11-13 October, 2005, Jeju Island, Republic of Korea

Dipanjan Das, Monojit Choudhury, An Affinity Based Greedy Approach towards Chunking for Indian Languages Department of Computer Science and Engineering Indian Institute of Technology, Kharagpur