Towards Verb Frame Extraction: Clustering Verb Arguments

Team: Vidyadhar Rao, Chandrakanth M, Mentor: Abhilash I

Introduction:

Words are the basic units of any language which are associated / mapped with the real objects. A collection of words is a sentence that carries some meaning with it. Verbs are the most grammatical category in a language. Actions, activities, and states are denoted with the help of verbs. Verb requires various participants which are specified by the arguments. Verbs play a major role in interpreting the sentence meaning. Verb frames are building blocks of linguistic computational models. So, the study of verb structure will provide a knowledge base for Natural Language Processing.

Verb Frame:

The relation of Verb with the other components of a sentence like mandatory arguments are specified by word order, presence of case markers on the arguments, karaka etc. These relations reflect the semantics of the verb i.e., provide a good understanding of interpreting the sense of the verb being used in a sentence. Other information like tense, aspect, modality, gender, number, person etc., also allow the verb to take specific variations in a language. A verb Frame is represented as tabular form showing

- Karaka relations
- Vibhakti (post position taken by the argument)
- Lexical category of the arguments.

  An example of Hindi verb frame for xEnA is shown below.

arc-label	vibhakti	lextype
k1	0	n
k2	ko	n

- Addition information like Necessity of the argument i.e., number of mandatory arguments also gives an idea of usage of the verb in a language.

Problem:

Task is “Clustering of Verb Arguments” in order to estimate the number of mandatory arguments of a verb and vibhakti of these arguments. The underlying premise for using clustering approach to the problem is for a given verb, arguments share similar properties. For example all arguments that stand in K1 relation take a post position ‘ne’. The clustering is done on the arguments of a verb extracted from a corpus of huge size.

Approach:

Approach to the problem involves the following tasks

- Extraction of simple sentences
    - The problem of conflicts between two verbs for the same argument in a sentence does not arise in simple sentences.
    - Shallow parser is used to extract sentences with single verb (VM).
- Extraction of Potential Argument list
    - All the NP chunk heads from the shallow parser output are considered for our evaluation.
- Clustering of Potential Arguments for a given verb.

Experimental Setup:

Amar ujala corpus is a huge corpus of 324MB has been used in our experiments. Monosemous verbs are simpler to handle and reduces the complexity of our problem. So, these verbs are only considered in our experiments. These verbs are extracted from Hindi Word Net.

Extracting simple sentences:

Shallow parser is used to do this task but it is be *very slow*. The shallow parser took ~20 hrs to run on 1200 files. Average number of sentences in a file is

25. The corpus has around 50K files but it took *40 days* to run on 32K files. 8800 simple sentences with monosemous verbs are extracted.

- Top 7 monosemous verbs extracted are
  - Beja 1429
  - KarIxa 253
  - mara 245
  - harA 205
  - liKA 172
  - beca 159
  - Gera 105
- We have considered verbs with at least 100 simple sentences.

Extraction of Potential Argument list along with their Features:

All the NP chunk heads from the shallow parser output are considered for our evaluation. The following features are used for the arguments based on the fact that we have a very large corpus,

- Distance from verb
  - Mandatory arguments tend to appear closer to verb. Only the Chunk distance is used as of now.
- Frequency of the argument
  - Mandatory arguments appear more often with the verb.
- Post position, gender, number, person

A combination of the above features is used to cluster the arguments for a verb.

Clustering the Arguments:

We have used weka clustering tool to cluster the obtained instances of arguments for each verb. The clustering technique used is Expectation-Maximization (EM) Clustering. Its ability to deal with missing data and observe unidentified variables is used for data clustering in machine learning. In Natural Language Processing, two prominent instances of the algorithm are Baum-Welch Algorithm and inside-outside algorithm for unsupervised induction of probabilistic context-free grammars.

Given the instances of the arguments for a verb the clustering tool outputs clusters of arguments. The arguments having similar features or properties tend to fall in the same cluster. So, the number of clusters formed would give an estimate of the number of (mandatory) arguments the verb would take on in a language. The obtained clusters give only an estimate of the number of arguments to the verb and do not specify the type of relation the verb holds with a particular cluster. In order to evaluate the relations we need to verify whether similar arguments are falling into the same cluster. As of now the weka tool does not provide the information about an instance of argument to a particular cluster. So, we are looking of other tools which might give us this information to evaluate the relation of clusters with the arguments. The experimental results are shown in File "Results.tar".

Guided Clustering using Linguistic Cues:

1. We need to eliminate non-arguments in our data like Genitives and others. For example in the sentence "Mohan gave Ram`s pen to Sita." The NP "Ram" is an argument to "Pen" and not to the main verb of the sentence "gave". But we haven`t eliminated such cases in our experiments.
2. We also need to identify the arguments directly using Derivational features/cues, etc.
    a. Using a richer set of semantic features like Derivational morphological features.
    b. For example consider the agentive derivations shown below.

Suffix	Example
-ar	कुम्हार सुनार लुहार
-ek	लेकक, मारेक

c. Using the suffix information we can directly identify the arguments for such agentive derivation. More over these semantic features are easy to compute.

3. For an elaboration on more such cues can be refer to Chapter#5, Hindi, Yamuna Kachru.

Conclusions:

Clustering using rationale features makes explicit the regularities in the usage of verbs with the arguments. These methods also give frequency of usage. These methods are helpful for unlabelled dependency parsing. But to make it useful for dependency parsing an evaluation on relation of arguments of a cluster with the verb is yet to be done from our experiments.