Part-Of-Speech Tagging for Gujarati Using Conditional Random Fields

Chirag Patel and Karthik Gali
Language Technologies Research Centre
International Institute of Information Technology
Hyderabad, India
chirag_p,karthikg@students.iiit.ac.in

Abstract

This paper describes a machine learning algorithm for Gujarati Part of Speech Tagging. The machine learning part is performed using a CRF model. The features given to CRF are properly chosen keeping the linguistic aspect of Gujarati in mind. As Gujarati is currently a less privileged language in the sense of being resource poor, manually tagged data is only around 600 sentences. The tagset contains 26 different tags which is the standard Indian Language (IL) tagset. Both tagged (600 sentences) and untagged (5000 sentences) are used for learning. The algorithm has achieved an accuracy of 92% for Gujarati texts where the training corpus is of 10,000 words and the test corpus is of 5,000 words.

1 Introduction

Parts of Speech tagging is the process of tagging the words of a running text with their categories that best suits the definition of the word as well as the context of the sentence in which it is used. This process is often the first step for many NLP applications. Work in this field is usually either statistical or machine learning based, or rule based. Some of the models that use the first approach are Hidden Markov Models (HMMs), Conditional Random Fields (CRFs), Maximum Entropy Markov Models (MEMMs), etc.

The other method is the rule based approach where by we formulate rules based on the study of the linguistic aspect of the language. These rules are directly applied on the test corpus. The statistical learning based tools attack the problem mostly as a classification problem. They are not language specific and hence they fail when semantic knowledge is needed while tagging a word with more than one sense. Even for unknown words, i.e., those words which have not appeared in the training corpus, these tools go by the probabilities but are not guaranteed to give the correct tag as they lack the semantic knowledge of the language. Also, they need a large annotated corpus. But the bright side of these tools is they can tag any word (known or unknown) with a high accuracy based on the probabilities of similar tags occurring in a particular context and some features provided for learning from the training data.

On the other hand, purely rule based systems fail when the word is unknown or does not satisfy any of the rules. These systems just crash if the word is unknown. They cannot predict the plausible or likely tag. Hence an exhaustive set of rules are needed to achieve a high accuracy using this approach.

There is another class of tools which are the hybrid ones. These may perform better than plain statistical or rule based approaches. The hybrid tools first use the probabilistic features of the statistical tools and then apply the language specific rules on the results as post processing. The best approach which seems intuitive is to generalize the language specific rules and convert them into features. Then incorporate these features into the statistical tools. The problem here is the lack of control and flexibility on the statistical tools. So the perfect selection of features is what actually matters with respect to the accuracy. The more lan-

Proceedings of the IJCNLP-08 Workshop on NLP for Less Privileged Languages, pages 117–122,
Hyderabad, India, January 2008.

guage specific features that can be designed the higher accuracy can be achieved.

2 Previous Work

Different approaches have been used for part-of-speech tagging previously. Some have focused on rule based linguistically motivated part-of-speech tagging such as by Brill (Brill, 1992 and Brill, 1994). On the machine learning side, most of the previous work uses two main machine learning approaches for sequence labeling. The first approach relies on k-order generative probabilistic models of paired input sequences, for instance HMM (Frieda and McCallum, 2000) or multilevel Markov Models (Bikel et al. 1999).

CRFs bring together the best of generative and classification models. Like classification models, they can accommodate many statistically correlated features of the input, and they are trained discriminatively. And like generative models they can also tradeoff decisions at different sequence positions to obtain a globally optimal labeling. Conditional Random Fields were first used for the task of shallow parsing by Lafferty et al. (Lafferty et al., 2000), where CRFs were applied for NP chunking for English on WSJ corpus and reported a performance of 94.38%. For Hindi, CRFs were first applied to shallow parsing by Ravindran et al. (Ravindran et. al., 2006) and Himanshu et al. (Himanshu et. al., 2006) for POS tagging and chunking, where they reported a performance of 89.69% and 90.89% respectively. Lafferty also showed that CRFs beat related classification models as well as HMMs on synthetic data and on POS-tagging task.

Several POS taggers using supervised learning, both over word instances and tagging rules, report precision greater than 96% for English. For Hindi and other South Asian languages, the tagged corpora is limited and together with higher morphological complexity of these languages it poses a difficulty in achieving results as good as those achieved for English in the past.

3 Conditional Random Fields

Charles Sutton et al. (Sutton et al., 2005) formulated CRFs as follows. Let G be a factor graph over Y. Then p(y|x) is a conditional random field if for any fixed x, the distribution p(y|x) factorizes according to G. Thus, every conditional distribution p(y|x) is a CRF for some, perhaps trivial, factor graph. If F = {A} is the set of factors in G, and each factor takes the exponential family form, then the conditional distribution can be written as

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{\Psi_A \in G} \exp\left\{ \sum_{k=1}^{K(A)} \lambda_{Ak} f_{Ak}(\mathbf{y}_A, \mathbf{x}_A) \right\}.$$

X here is a random variable over data sequences to be labeled, and Y is a random variable over corresponding label sequences. All components Y_i of Y are assumed to range over a finite label alphabet Y. For example, X might range over natural language sentences and Y range over part-of-speech tagging of those sentences, with Y the set of possible part-of-speech tags. The random variables X and Y are jointly distributed, but in a discriminative framework we construct a conditional model p(Y|X) from paired observation and label sequences, and do not explicitly model the marginal p(X).

CRFs define conditional probability distributions $P(\mathbf{Y}|\mathbf{X})$ of label sequences given input sequences. Lafferty et al. defines the probability of a particular label sequence Y given observation sequence X to be a normalized product of potential functions each of the form:

$\exp(\Sigma \boldsymbol{\lambda} j \boldsymbol{t} j(Yi\text{-}1, Yi, X, i) + \Sigma \boldsymbol{\mu} k \boldsymbol{s} k\,(Yi, X, i))$

where $tj(Yi\text{-}1, Yi, X, i)$ is a transition feature function of the entire observation sequence and the labels at positions i and $i\text{-}1$ in the label sequence; $sk(Yi, X, i)$ is a state feature function of the label at position I and the observation sequence; and $\boldsymbol{\lambda} j$ and μk are parameters to be estimated from training data.

$Fj(Y, X) = \Sigma\, fj\,(Yi\text{-}1, Yi, X, i)$

where each $fj\,(Yi\text{-}1, Yi, X, i)$ is either a state function $s(Yi\text{-}1, Yi, X, i)$ or a transition function $t(Yi\text{-}1, Yi, X, i)$. This allows the probability of a label sequence **Y** given an observation sequence **X** to be written as:

$P(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\lambda}) = (1/Z(\mathbf{X}))\, \exp(\Sigma \boldsymbol{\lambda} j\, Fj(\boldsymbol{Y}, \boldsymbol{X}))$

where $Z(\mathbf{X})$ is a normalization factor.

4 IL Tagset

The currently used tagset for this project and which is a standard for Indian Languages is the IL (Indian

Languages) tagset. The tagset consists of 26 tags. These have been specially designed for Indian Languages. The tagset contains the minimum tags necessary at the Parts of Speech tagging level. It copes with the phenomena of fineness versus coarseness. The tags are broadly categorized into 5 main groups, with the nouns consisting of the general nouns, space or time related nouns or proper nouns, and the verbs consisting of the main and the auxiliary verbs. Another category is of the noun and verb modifiers like adjectives, quantifiers and adverbs. Finally, there are numbers, cardinals etc.

5 Approach

Approach presented in this paper is a machine learning model. It uses supervised as well as unsupervised techniques. It uses a CRF to statistically tag the test corpus. The CRF is trained using features over a tagged and untagged data. A CRF when provided with good features gives accuracy much better than other models. The intuition here is that if we convert the linguistic rules specific to Gujarati in to features provided to CRF, then we make use of advantages of both statistical and rule based approach. But due to lack of control and flexibility not all features can be incorporated in the CRF. So after the CRF is done we do the error analysis. From the errors we formulate rules, which are general and language specific, and then convert them to new features and apply them back to CRF. This increases the accuracy.

Gujarati when viewed linguistically is a free word order language. It is partially agglutinative, in the sense maximum 4 suffixes can attach to the main root. Words in Gujarati can have more than one sense where the tags are different in different senses. For e.g. “paNa” can be a particle meaning – “also”, and also can be a connective meaning – “but”. “pUrI” can be a noun meaning – “an eatable”, can be an adjective meaning – “finished”, and can also be a verb meaning – “to fill”.

Also, in Gujarati, postpositions can be or can not be attached to the head word. For e.g. One may write “rAme” or “rAma e” literally meaning “rAma (ergative)”.

Most of all, this language can drop words from the sentences. For example:

Sent: baXA loko GaramAM gayA.
Literal: all people house + in went.
Tags: QF NN NN VM

Here, we can drop the noun (NN) “loko” and in which case the quantifier (QF) “baXA” now becomes the noun (NN).

Features used in CRF are suffixes, prefixes, numbers etc. For e.g. Words having suffix “ne”, like “grAhakone” are tagged as NN. CRF learns from the tags given to words with same suffixes in the training data. This suffix window is 4. This way the vibhakti information is explored. Similarly if words like “KAine” and “KAwo” come in the training corpus the CRF learns the preffix and tags other words with that prefix. This way the stem information is explored. Also if the token is a number then it must be QC, and if it has a number in it then it must be a NNP.

6 Experiments

Initially we just ran a rule based tagging code on the test data. This code used both machine learning and rule based features for tagging. It gave an accuracy of 86.43%. The error analysis revealed that, as the training corpus being less, the unknown words are many and also well distributed over the tags. Hence the heuristics were not effective.

Then we ran a CRF tool on the test data. We found it giving an accuracy of 89.90%. Then during the error analysis we observed that the features were not up to the mark. Then we selected particular features which were generalization of rule based, used in the previous code, and more specific to Gujarati. This increased the accuracy to 91.74%. Then after adding more heuristics the accuracy was in fact reducing. Heuristics like converting all NNPs to NNs, removing some tags as options while tagging the unknown words like CC,QW,PRP etc. as these in a language are very limited and are expected that they must have came once in the training corpus. We also tried tagging the word on the basis of possible tags between the two surrounding words. But that too reduced the accuracy. Also heuristics like previous and current word vibhakti combination failed.

Training data	Test data	Results (%)
11185	5895	91.74

Table-1. POS Tagging Results and Data Size

7 Error Analysis

Here the above table confirms that the errors have occurred across all the tags. This is mainly due to lack of training data. The numbers of unknown words in the corpus were around 40%. The CRF while using the features and the probabilities to tag a particular unknown word made mistakes due to the flexible nature of the language. For e.g. the maximum errors occurred because of tagging an adjective by a noun. An example:

motA`QFC BAganA`QF viSeRa`NN SEkSaNika`JJ jaruriyAwo`NN GarAvawA`VM bAlYako`NN sAmAnya`JJ skUlamAM`NN jaSe`VM .`SYM

Actual Tag	Assigned Tag	Counts
JJ	NN	58
NNP	NN	35
NN	JJ	26
NN	VM	22
NNC	NN	21
PSP	NN	19
VM	VAUX	19
NNPC	NN	18
NNC	JJ	17
NST	NN	14
VM	NN	13

Table-2. Errors Made by the Tagger.

In the above example the word "viSeRa`NN" is wrongly tagged. This being an adjective is tagged as NN, firstly because it is an unknown word. Also in this language adjectives may or may not occur before the nouns. Hence the probability of this unknown word to be a NN or a JJ is equal or will depend on the number of instances of both in the training corpus. Further more there is more probability of it being tagged as a noun as the next word is an adjective. There are very less instances where two adjectives come together in the training corpus. Again the chances of it being a noun increase as the QF mostly precede nouns instead of adjectives. Here we also have a QF before the unknown word. The same reason also is responsible for the third class of errors – NN being wrongly tagged as JJ. These errors can only be corrected if the word is some how known. Again the next class of errors is the Named Entity Recognition problem which is an open problem in itself.

8 Conclusion

We have trained a CRF on Gujarati which gives an accuracy of around 92%. From the experiments we observed that if the language specific rules can be formulated in to features for CRF then the accuracy can be reached to very high extents. The CRF learns from both tagged that is 600 sentences and also untagged data, which is 5,000 sentences.

From the errors we conclude that as the training data increases, the less number of unknown words will be encountered in the test corpus, which will increase the accuracy. We can also use some machine readable resources like dictionaries, morphs etc. when ever they are built.

9 Intuition

We noticed that on a less amount of training data also we have a good accuracy. The reason we felt intuitive was Gujarati uses the best part of the vibhakti feature linguistically. It, being more agglutinative than Hindi has more word forms, hence more word coverage, and being some less agglutinative than Telugu, has less ambiguity and also is practical to hard code the vibhaktis, uses the best part of advantages of the vibhakti feature in POS tagging. Based only on the hard coded vibhakti information we could tag around 1500 unknown words out of 5000.

10 Future work

We are looking forward to manually tag more training data in the future. We will also be trying to build language resources for Gujarati that will help in the Tagger. By increasing the amount of training data we expect an appreciable increase in the accuracy.

References

Himanshu Agarwal and Anirudh Mani. 2006. Part of Speech Tagging and Chunk-ing with Conditional Random Fields. *In the Proceedings of NWAI workshop.*

Pranjal Awasthi, Delip Rao, Balaraman Ravindran. 2006. Part Of Speech Tagging and Chunking with HMM and CRF. *Proceedings of the NLPAI contest workshop during NWAI '06, SIGAI Mumbai.*

Karthik Kumar G, Sudheer K, Avinesh PVS. 2006. Comparative study of various Machine Learning methods For Telugu Part of Speech tagging. *Pro-*

ceedings of the NLPAI contest workshop during NWAI '06, SIGAI Mumbai.

John Lafferty, Andrew McCallum and Fernando Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segment-ing and Labeling Sequence Data. In *proceedings of ICML'01*.

Avinesh PVS and Karthik G. 2007. Part-Of-Speech Tagging and Chunking Using Conditional Random Fields and Transformation Based Learning. *Proceedings of the SPSAL workshop during IJCAI'07.*

Fei Sha and Fernando Pereira. 2003. Shallow Parsing with Conditional Random Fields. In the *Proceedings of HLT-NAACL.*

Charles Sutton. 2007. An Introduction to Conditional Random Fields for Relational Learning. *In proceedings of ICML'07.*

CRF++: Yet Another Toolkit.

http://chasen.org/~taku/software/CRF++