TO FIND THE POS TAG OF UNKNOWN WORDS IN PUNJABI LANGUAGE

[1]Blossom Manchanda, [2]Mr. Ravishanker, [3]Sanjeev Kumar Sharma

[1]Lecturer,
B.I.S College of Engineering and Technology, Moga – 142001
[2]Lecturer, LPU, Jalandhar,
[3]Associate Professor, B.I.S College of Engineering and Technology

Abstract : *The accuracy of unknown words in the task of Part of Speech tagging is one significant area where there is still room for improvement. Because of their high information content, unknown words are also disproportionately important for how often they occur, and increase in number when experimenting with corpora from different domains. One area however, where all POS tagging methods suffer a significant decrease in accuracy, is with unknown words. These words are those that are seen for the first time in the testing phase of the tagger, having never appeared in the training data. In general, on POS tagging as well as other similar NLP tasks, accuracy on unknown words is about 10% less than words that have been seen in the training data (Brill, 1994). Unknown words also occur a significant amount of the time, comprising approximately 5% of a test corpus (Mikheev, 1997).*

Introduction to Unknown Words

Part of Speech (POS) tagging involves assigning basic grammatical classes such as verb, noun and adjective to individual words, and is a fundamental step in many Natural Language Processing (NLP) tasks. The tags it assigns are used in other processing tasks such as chunking and parsing, as well as in more complex systems for question answering and automatic summarisation. All POS taggers suffer a significant decrease in accuracy on *unknown words*, that is, words that have not been previously seen in the annotated training set. A loss of up to 10% is typical for most POS taggers e.g. Brill (1994) and Ratnaparkhi (1996)[1]. This decreased accuracy has a flow on effect for the accuracy of both following POS tags and later processes which utilize them. Unknown words also occur a significant amount of the time, ranging from 2%– 5%(Mikheev, 1997), depending on the training and test corpus. These figures are much higher for domains with large specialist vocabularies, for example biological text. We improve the performance of a Maximum Entropy POS tagger by implementing features with non-negative real values. Although Maximum Entropy is typically described assuming binary-valued features, they are not in fact required to be binary valued. The only limitations come from the optimization algorithm. For example, the Generalised Iterative Scaling (Darroch and Ratcliff, 1972) algorithm used in these experiments imposes a non-negativity constraint on feature values. Real-valued features can encapsulate contextual information extracted from around unknown word occurrences in an unannotated corpus. Using a large corpus is important because this increases the reliability of the real-values. By looking at the surrounding words, we can formulate constraints on what POS tag(s) could be assigned. This can be seen in the sentence below:

(1) pkg{i hefj dz fe g[Zo NB; a fdb B{eJhfpwkom Kbkfdzhj ?

Here, NB; a is the unknown word (word from other language) which, as competent speakers of the language, we can surmise is probably a noun or adjective. This is because it sits between two nouns, which is a position frequently assumed by words with these syntactic categories. Also, if we can find the word NB; a in other places, then we can get an even better, more reliable idea of what its correct tag should be. Since we see it so often, we know the types of words that follow it quite well. Other words that occur less frequently don't give as strong an indication of what is to follow, simply because the evidence is sparser. Our aim then, is to take this intuitive reasoning for determining the correct tag for an unknown word.

Unknown word processing

POS taggers have reached such a high degree of accuracy that there remain few areas where performance can be improved significantly. Unknown words are one of these areas, with state-of-the-art accuracy in the range 85 – 88%, which is well below the 97% accuracy achievable over all words. The prevalence of unknown words is also problematic, although somewhat dependant on the size and type of corpus being used. Rports shows that training on sections 0–18 of the Penn Treebank (Marcus et al., 1993), and test on sections 22–24. This test set then contains 2.81% (approximately 4000) unknown words. Also, when applying a POS tagger to a specialized area of text, such as technical papers, the number of unknown words and their frequency would be expected to increase dramatically, due to specific jargon terms being used. Unknown words are also more likely to carry a greater semantic significance than known words in a sentence. That is, they will often contain a larger amount of the content of the sentence than other words. This is because unknown words are unlikely to be from closed-class categories such as determiners and prepositions, but quite likely to be in open class categories such as nouns and verbs. It is these classes that generally convey most of the information in a sentence. Further, rarer words often have a more specialized meaning, and thereby classifying them incorrectly will potentially lose a lot of information. For these reasons, it is quite important that unknown words are POS tagged correctly, so that the information carried by them can be extracted properly in future stages of an NLP system. Previous work on tagging unknown words has focused on morphological features, and using common affixes to better identify the correct tag. This has been done using manually created, common English endings (Weischedel et al., 1993), with Transformation Based Learning (TBL) (Brill, 1994), and by comparing pairs of words in a lexicon for differences in their beginnings and endings (Mikheev, 1997). There is no such work has been done for Punjabi Language.

The Trigram Model

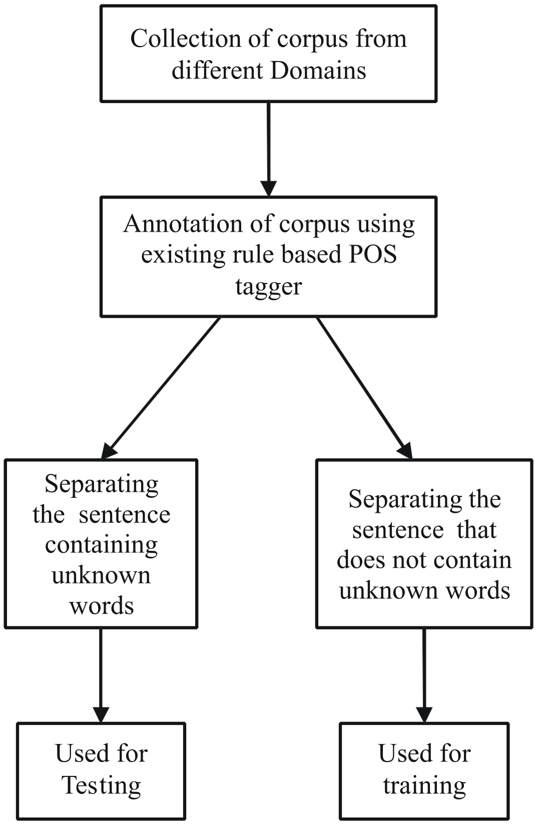
It is assumed that the unknown POS depends on the previous two POS tags, and calculate the trigram probability P(t3|t1, t2), where t3 stands for the unknown POS, and t1 and t2 stand for the two previous POS tags. The POS tags for known words are taken from the tagged training corpus. Similarly we can also calculate the POS tag of unknown word if we know the POS tag of previous and the next word of unknown words are known to us. Another similar way is using the POS tag of next two tags instead of previous two tags. So we used all these three methods in three different situations.

Case 1: If the POS tag of previous and next word to unknown is known to us, then we will calculate the trigram probability P(t3|t1, t2), where t3 stands for the unknown POS, and t1 and t2 stand for the previous and next word POS tags respectively.

Case 2: If the POS tag of previous word to unknown word is unknown which means previous word is also a unknown word, then we will calculate the trigram probability P(t3|t1, t2), where t3 stands for the unknown POS, and t1 and t2 stand for the POS tags of next two words.

Case 3: If the POS tag of next word to unknown word is unknown which means next word is also a unknown word, then we will calculate the trigram probability P(t3|t1, t2), where t3 stands for the unknown POS, and t1 and t2 stand for the POS tags of previous two words.

Now in order to calculate the trigram probability an annotated corpus was developed. This corpus was collected from different web sites by keeping in mind that all the common domains should be covered. Then this corpus was tagged by using a pre existing rule based POS tagger. This pre existing POS tagger uses 630 tags which covers almost all the word classes with their inflections. This trained POS tag was divided in to two different corpuses, one containing the sentences without any unknown word and the other containing the sentences that contain unknown words. The corpus that does not contain any unknown word was used for training the model and the other portion that contains unknown words was used for testing.


Collection of corpus from different Domains
Annotation of corpus using existing rule based POS tagger
Separating the sentence containing unknown words
Separating the sentence that does not contain unknown words
Used for Testing
Used for training

Collection of corpus

The basic need for using the statistical methods/techniques is the availability of annotated corpus. More the corpus available more will be the accuracy. Another thing that should be kept in mind is that the corpus should be accurate. So we started our work with the collection of accurate corpus. While collecting the corpus we kept the following points in our mind:-

- The corpus should be in Unicode.
- The corpus should be accurate i.e. it should have minimum no of spelling mistake.
- The corpus should not be domain specific.
- The corpus should contain as many different words as possible.

The main sources of our corpus are:-
http://punjabikhabar.blogspot.com
http://www.quamiekta.com
http://www.europediawaz.com
http://www.charhdikala.com
http://punjabitribuneonline.com
http://www.sadachannel.com
www.veerpunjab.com
www.punjabinfoline.com

Annotation of the corpus

Annotation of the corpus means giving a tag to the every individual word. The next step that we performed after the collection of corpus was to annotate the corpus. We annotate the corpus by using a tool named TAGGER. This tool is developed by us. This tool has been developed from a pre existing Rule based POS Tagger. We made some alteration in that pre existing tool and used it for the annotation of our corpus.

Screening/Filtering of annotated corpus

As the annotated corpus contains many words having ambiguous tags i.e. the words having more than one tag, so we filtered the sentence that contains ambiguous words. In this way we divided the annotated corpus in two parts, one containing the sentences that have ambiguous words and the other that does not contain any sentence having ambiguous word. After this first filtering we applied another type of filtering. From the annotated corpus that does not contain any ambiguous word we separate the sentence that does not contain unknown words.

Creating Triplets with frequency

From the corpus that does not contain any unknown word we created the triplets of pos tags. After creating triplets we calculate their frequency of occurrence in the corpus.

Triplet	Frequency
NNFSD_VBP_VBMAXSS3XBNO	1
VBP_VBMAXSS3XBNO_PTUKE	1
VBMAXSS3XBNO_PTUKE_PNPMPGDF	2
PTUKE_PNPMPGDF_NNMSO	3
PNPMPGDF_NNMSO_PPIBSD	44
NNMSO_PPIBSD_NNMSD	119
PPIBSD_NNMSD_VBMAMSXXPINIA	9
NNMSD_VBMAMSXXPINIA_CJC	21
VBMAMSXXPINIA_CJC_AVU	8
AJIFSD NNFSD VBMAFSXXPTNIA	70
NNFSD VBMAFSXXPTNIA CJC	40
VBMAFSXXPTNIA CJC NNFSD	5
CJC NNFSD VBMAXSS3XBNO	3
CJC_AVU_PTUE	35

Results and Discussion

We divide the testing corpus in to four parts of equal length. These four equal parts contains different no of unknown words. We get the following results:

Total words in the corpus	No of unknown words	Correctly tagged Unknown words	Incorrectly tagged Unknown words	Not tagged
12430	547	392	92	63
12450	345	254	30	61
12444	355	225	25	105
12465	456	329	73	54

The reason for not tagging the unknown word is absence of the triplet with that combination. Most of the untagged unknown words are of similar type. The incorrect tags of unknown words can be further reduced by selecting two highest frequency triplets satisfying the condition. Suppose we have a word **ft bZB** in following sentence:

fJ; fcañw d/nz ft Z ft bZB B{fg; sb Bkba ; {N eoB dhpi kJ/i fj o d/e/wkfonki Kdkj ?

(fJ; _PNDBSO fcañw_NNFSO d/_PPIDAMSO nz_NNMSO ft Z_PPIBSD ft bZB_Unknown B{PPUNU fg; sb_Unknown Bkb_AVU ; {N_NNFSD eoB_NNMSO dh_PPIDAFSO pi kJ/_PPU i fj o_NNMSO d/_PPIDAMSO e/_PPIMPD wkfonk_VBMAMSXXPTNIA i Kdk_VBOPMSXXXINDA j ?VBAXBST1 . _Sentence)

In above sentence ft bZB is unknown word.

When we search for the triplet

PPIBSD _Unknown _PPUNU

We get many combinations with different frequencies but the two highest frequencies are

PPIBSD _NNMSO_PPUNU	54
PPIBSD _NNFSO_PPUNU	48

So instead of replacing Unknown with NNMSO (with highest frequency 54) we prefer replace Unknown with NNMSO/NNFSO. Further the POS tagger will resolve this ambiguity.

References:

[1] Adwait Ratnaparkhi, (1996) "A Maximum Entropy Model for Parts-of-Speech Tagging", 1-7.

[2] Chooi-Ling GOH, Masayuki ASAHARA, Yuji MATSUMOTO, (2004) "Pruning False Unknown Words to Improve Chinese Word Segmentation"139-148.

[3] Huihsin Tseng, Daniel Jurafsky, Christopher, (2005) "Morphologcal features help POS tagging of unknown words across language varieties", 1-8.

[4] Ilya Segalovich, (2006) "A fast morphological algorithm with unknown word guessing induced by a dictionary for a web search" 3-4

[5] Shulamit Umansky-Pensin, Roi Reihart, Ari Rappoport, (2010) "A Multi Domain Web Based Algorithm for POS Tagging of Unknown Words", 1-9.

[6] Tetsuji Nakagawa, Taku Kudoh, Yuji Matsumoto, (2001) "Unknown Word Guessing And Part-of-Speech Tagging using Support Vector Machines", 1-7.

[7] Tetsuji Nakagawa, Yuji Matsumoto, (2006) "Guessing Parts of Speech of Unknown Words using Global Information", 1-8.

[8] Xiaofei Lu, (2005), "Hybrid Methods for POS Guessing Of Chinese Unknown Words", 1-6.