HindiTech
A Blind Double Peer Reviewed Bilingual Web-Research Journal

Maximum Entropy Model For Sense Tagging

Sree Ganesh

sganeshhcu@gmail.com

Professor at SRH University, Germany.

1. Introduction:

The main thrust of Natural Language Processing (NLP) research at present is to understand the natural language texts. Language is a product of human mental activity and is in a state of continuous change. One linguistic expression can have many surface forms. Similarly, each surface form, in turn, can be interpreted as many linguistic expressions. Hence, ascertaining what is intended in a text when more than one interpretation is possible, has become a central issue in Natural Language Processing. Ambiguity has become an obstacle in language processing.

An ambiguity exists in a natural language sentence, when it has more than one interpretation. The ambiguities are of many types: (i) Lexical ambiguity – in which a word has more than one interpretation with respect to part of speech, meaning etc. The word "*bank*" is ambiguous between the financial institution and the edge of the river. Another example is the word "idle", which can occur either as a verb or as an adjective; (ii) Syntactic ambiguity – in which the alternative syntactic representations make it structurally ambiguous. Consider the sentence, *Stolen painting found by tree* which means in the following ways. (a) A tree found a stolen painting. (b) A person found a stolen painting near a tree. Consider another example sentence, "you can have peas and

beans or carrots with the set meal", which can mean (a) [peas] and [beans or carrots] (b) [peas and beans] or [carrots]; (iii) Semantic ambiguity - in which several interpretations result from the different ways in which the meaning of words in a phrase can be combined. For example, consider the sentence, "Iraqi head seeks arms" which can mean (a) Chief of Iraq wants to have weapons (b) The head of Iraqi requires some hands to be attached; (iv) Pragmatic ambiguity - results in situations where same phrase gives different meanings in the same context [She97]. Normally pragmatic ambiguities are concerned with the ironic and sarcastic situations; (v) referential ambiguity – where the referent pronouns are ambiguous. For example, consider the sentences – (a). "The men murdered the women. They are buried." (b) "The men murdered the women. They are caught. "- The pronoun "they" in the above sentences, refers to women in the context of (a) and men in the context of (b).

Hence resolution of these ambiguities is a prerequisite to understand a natural language text. The resolution of lexical and contextual ambiguities that exist in the given Telugu sentences was attempted in this thesis by developing Telugu Part-of-Speech (POS) taggers by adapting three different approaches.

1.1. POS Tagging

"POS tagging is the process of assigning a tag like noun, verb, pronoun, preposition, adverb, adjective or other lexical class marker to each word in a given sentence, considering the role or function of the word in the sentence [DeJa]."

Assigning a POS tag to each word in the sentence is not a routine task. Words can belong to different syntactic categories in different contexts. For instance, the word `books` can have two readings- in the sentence "he books tickets", the word "books" is a third person singular verb, but in the sentence "he reads books", it is a plural noun. A POS tagger should segment a word, determine its possible readings and assign the right reading in the given context.

The input to a POS tagging algorithm is a sentence and the output is a tagged sentence resolving the ambiguities at word and syntactic levels that exist in the sentence, using a set of lexical and contextual rules. Unknown words are also assigned a suitable tag.

Generally a POS tagger consists of three components. They are (i) Tokeniser (ii) Morphological Analyser or Morphological Classifier and (iii) Morphological Disambiguator.

The tokeniser is responsible for segmenting the input text into words or phrases. More advanced tokenisers attempt to recognize proper names, acronyms, phrasal constructions, etc, as single tokens, usually employing specialized dictionaries and grammar rules. For example, it is often useful to recognize phrases like “in front of ” as a single unit rather than just as sequence of words in the text.

After tokenization, the output is fed to the morphological analyser for assigning one or more number of POS tags depending on its morpho-syntactic features. It is just a lexicon lookup for non-inflected languages. Morphologically highly inflected languages need some processing to extract the morpho-syntactic features which require the use of morphological analysers to extract the information encoded in the words of the given sentence. The output of the morphological analyser gives all possible readings for each word in the sentence along with the other grammatical features of the word.

Each analysis of the word given by a morphological analyser is unambiguously mapped into a POS tag in the POS tagging framework. For example, a singular noun can be tagged as NN and a plural noun as NNP, the base form of a verb can be tagged as VB and its past form as VBD.

However, no lexicon can contain all possible words. When the morphological analyser comes across a word that is not in the lexicon or in the training set, the tagger tries to guess its tag.

1.2. Applications of POS Tagging

POS tags play an important role in NLP applications. These tags give us information about the word and its neighbours and thus limit the range of meanings and help in the fields of text to speech, shallow parsing, information retrieval and extraction, word sense disambiguation, language modelling, guessing unknown words etc. A parse tree can be built based on POS tags instead of deeply parsing the entire text.

1.3. Choosing a Tag Set

POS tagging is the process to determine the POS tag for each word in the given text. The collection of such POS tags for a given language is called a tag set. Developing a POS tag set for a given language is an arduous task. The tag set should be constructed in such a way that it should give better information about the context by optimally selecting a tag from the tag set. Hence the tags in the tag set needs to be constructed so that the following characteristics are achieved.

- The tags in the tag set should discard the lexical identity. For example, all nouns should be tagged as NN and articles should be tagged as DT.
- The tags should introduce the distinction between the words which have the same syntactic/semantic structure but have different roles in different contexts. For example, the word *books* should be tagged as NN or VB depending on the context.
- The tags should introduce a perfect classification scheme to predict the morpho-syntactic feature of any unknown word occurring in the text.

1.4. Need of POS Taggers for Telugu

Telugu is an agglutinative language and has a rich morphology. The grammatical relations are expressed by means of affixes. An attempt is made to analyses small Telugu corpus using a Telugu morphological analyzer [Uma04]. It gives all possible analyses of a given word in terms of its grammatical features. Each word can have null

or one or more analyses. Thus, unknown words will have null analysis. A word with no ambiguity will have only one analysis. The word with more than one analysis is ambiguous. This ambiguity can be resolved by considering its morpho-syntactic features, which are captured by POS tags.

From the following figure (Fig 1.), it is observed that 29% of the words are identified by Telugu Morphological Analyzer that has coverage of 98%. More than 40% of the words are ambiguous and 27% of the words are unknown. The reasons for this non-identification are due to (i) the presence of proper nouns, (ii) conjoining of two or more number of words written as a single word and (iii) presence of foreign words etc.

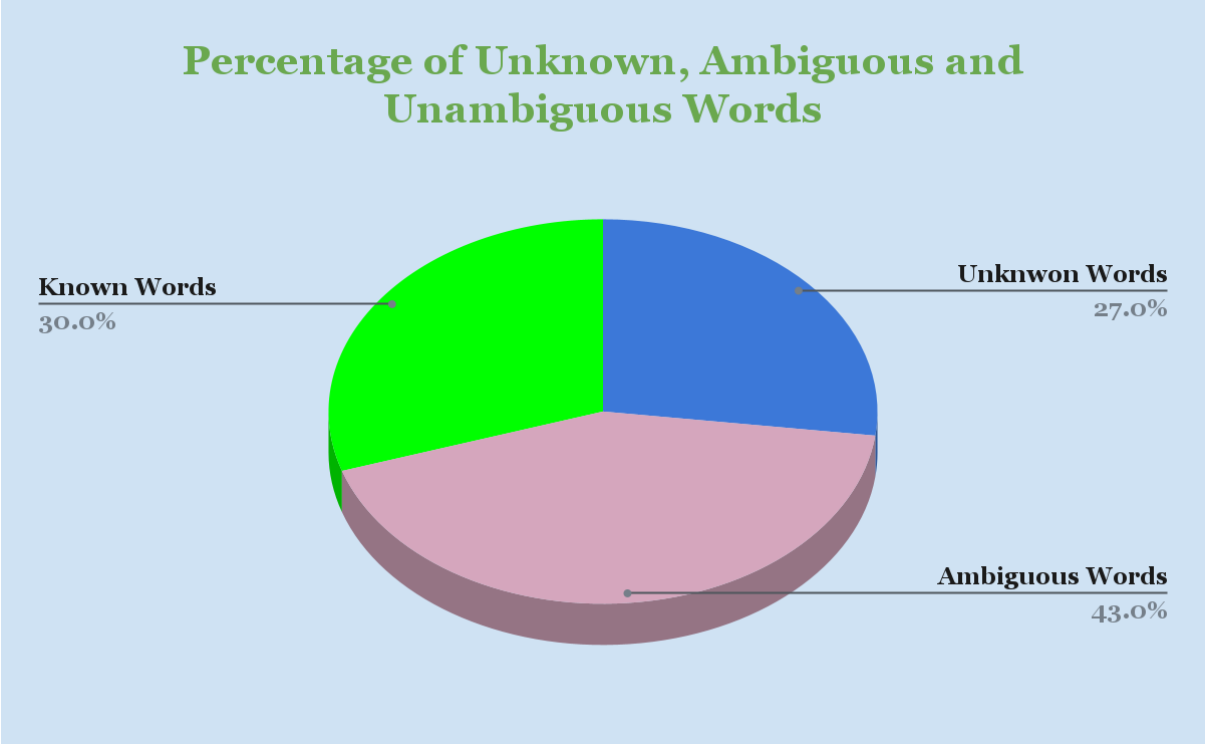

Fig. 1. Percentage of Unknown, Ambiguous and Unambiguous Words

The ambiguous and unknown words can be resolved or handled in Telugu under POS framework. For example, the noun suffix 'lo' in Telugu usually attaches with a noun and forms a locative (saptami). Sometimes it refers to the dubious role of the noun in the

sentence. For example, consider the word 'kurcIlO'. This word has two interpretations. One interpretation is "kurcIlO" (in the chair) and the other interpretation is 'kurcIlO ballalO" (either chairs or banches). If the word "kurcIlO" is succeeded by another similar noun inflection, the ambiguity can be resolved as a second interpretation. Consider another example-

kawa ceVppu
story tell
(Tell me a story).

Here the lexical item 'ceVppu' could be a noun meaning "chappl" or the imperative form of the verb "tell". Here in this sentence it is a verb which is more likely to succeed as a noun. Thus the above ambiguities can be resolved by modelling the Telugu language sentence structure.

2. Methods of POS Tagging:

There are many approaches to POS tagging. They are (i) Rule-based tagging Ex. ENGTWOL [Vou93,Vou94,Vou95] (ii) Stochastic tagging Ex. TnT tagger [Bra00] and (iii) Transformation Based tagging Ex. Brill Tagger [Bri95b] and so on. In this paper I have concentrated on the Maximum Entropy Model.

2.1. Maximum entropy model for Telugu POS tagging

An implementation of the Maximum Entropy model developed for POS tagging by Ratnaparkhi [Rat96] is used for Telugu POS tagging task. Many NLP problems like language Modelling [RaRoSa93], Machine Translation [AdStVi96], ambiguity resolution [Rat98], partial parsing [SkBr98], word Morphology [StViJo95] and for extracting context from the previous sentences [BlAnRu] are solved by using the maximum entropy model techniques. This Maximum Entropy model has also been applied to different languages like Chinese [JiXl02], Swedish [Beat], Hindi [AnKuUmSa] etc.

The following three sections briefly describe the theoretical description of Maximum Entropy modelling which was extracted from many sources.

2.2. The principle of Maximum Entropy

The principle of maximum entropy is a method for analyzing the available information in order to determine a unique epistemic probability distribution. Claude E. Shannon [WiIn, Mitch03] the originator of information theory, defined a measure of uncertainty for a probability distribution ($H(\boldsymbol{p}) = - \Sigma\, p_i \log p_i$) which he called information entropy. In his work, information entropy is determined from a given probability distribution. The principle of maximum entropy tells us that the converse is also possible: a probability distribution can be determined using the information entropy concept. It states the probability distribution that uniquely represents or encodes our state of information is the one that maximizes the uncertainty measure $H(\boldsymbol{p})$ while remaining is consistent with the information given .

Claude E. Shannon defines entropy in terms of a discrete random event x, with possible states 1. n as:

$$H(x) = \sum_{i=1}^{n} p(i) \log_2 \left(\frac{1}{p(i)}\right) = -\sum_{i=1}^{n} p(i) \log_2 p(i).$$

That is, the entropy of the event x is the sum, over all possible outcomes i of x, of the product of the probability of outcome i times the log of the probability of i (which is also called s's surprisal - the entropy of x is the expected value of its outcome's surprisal). This can also be applied to a general probability distribution, rather than a discrete-valued event.

Shannon shows that any definition of entropy satisfying his assumptions will be of the form:

$$-K\sum_{i=1}^{n} p(i)\log p(i).$$

where K is a constant (and is really just a choice of measurement units).

2.3. The Probability Model of Maximum Entropy for POS Tagging

A maximum entropy tagger learns a log linear conditional probability model from tagged text, using a maximum entropy method. The model assigns a probability for every tag t in the set T of possible tags given a word and its context h, which is usually defined as the sequence of several words and tags preceding the word. This model can be used for estimating the probability of a tag sequence $t_1..t_n$ given a sentence $w_1..w_n$:

$P(t_1..t_n \mid w_1..w_n) = \tilde{O}_{i=1}^{n} P(t_i \mid t_1..t_{i-1}, w_1..w_n) \approx \tilde{O}_{i=1}^{n} P(t_i \mid h_i)$

The principle of maximum entropy modeling used for POS tagging (assigning a maximum likelihood tag sequence to a sequence of words) is choosing the probability distribution p that has the highest entropy out of the distributions that satisfy the set of constraints. These constraints restrict the model to assign tags in accordance with the statistical data extracted from the training corpus.

The principle of maximum entropy is only useful when all the information is of a class called testable information. A piece of information is testable if it can be determined whether or not a given distribution is consistent with it.

Given testable information, the maximum entropy procedure consists of seeking the probability distribution which maximizes information entropy, subject to the constraints of the information. This constrained optimization problem is typically solved using the method of Lagrange Multipliers. Entropy maximization with no testable information takes place under a single constraint: the sum of the probabilities must be

one. The principle of maximum entropy can be seen as a generalization of the classical principle of indifference, also known as the principle of insufficient reason.

2.4. Features and Constraints

A statistical model of the process for a given training sample P(x,y) is constructed. The building blocks of this model are a set of statistics of the training sample. For example, in a training corpus, the frequency that the word *mark* is tagged to either a noun or a verb is 3/10 and 7/10 and so on. These particular statistics are independent of the context, but sometimes it depends on the conditioning information *x*. For instance, it might be noticed that, in the training sample, if the word "paper" is followed by the word "mark", then the POS of the mark is a verb with frequency 9/10.

To express the event that the word "mark" getting tagged as verb when the word "paper" is the following word, the indicator function can be introduced such as

f(x,y) = 1 if y = verb and "paper" follows "mark"

0 otherwise

The expected value of *f* with respect to the empirical distribution f(x,y) exactly may be the statistics of our interest.

2.5. Training and Testing the Tagger

Training a Maximum Entropy model is relatively easy. There is a Maximum Entropy Modeling toolkit [MxEnTk] freely available on the net. This toolkit consists of both Python and C++ modules to implement Maximum Entropy Modeling. Moreover, there is a separate language and tag set independent toolkit in Python (maxent) as a case study for building a POS tagger. This is straight way used to build POS tagger for Telugu.

The same corpus and tag set described in 6.1 are used to train the Maximum Entropy POS tagger.

2.5.1. Training the Tagger

First, a Maxent Model Instance is created and all training data sets (instances) are added to it. This is done as follows.

From maxent import MaxentModel

M = MaxentModel()

m.begin_add_event()

....

m.end_add_event()

Next a training module is called to train maxent with 100 iterations.

m.train(100,"lbfgs")

Now this training model is saved to telugu_tagger :

m.save("telugu_tagger")

This creates a binary file called telugu-tagger.

Alternatively there is a python program **postrainer.py** which takes a training corpus of a language directly with some options and extracts the language modelling of the corpus to a file (i.e., Telugu-tagger).

7.5.2. Results

A sentence from the output sample story is extracted and shown below. The first row of the table consists of input transliterated in Roman scheme as specified in the Appendix-I. The second and third rows show the corresponding English gloss and Telugu script of the given sentence respectively. The last row shows the output of Maximum Entropy tagger. The complete input story was shown in the Appendix–VIII. The entire output of Maximum Entropy tagger for the story is shown in the Appendix-XI

Telugu transliterated input	oVka vyApAri oVkasAri oVka mahanIyudu cese prasaMgAlanu vinadAniki poyAdu.
English Gloss	One tradesman once a great man given lectures (object) to listen went (Once a tradesman went to listen the lectures given by a great man)
Tagger Output	oVka/jj vyApAri/nn1 oVkasAri/nn1 oVka/jj mahanIyudu/nn1 cese/vnf prasaMgAlanu/nn1 vinadAniki/nn4 poyAdu/vf ./sym

Table 7.1. Sample Output of Maximum Entropy Tagger

2.6. Evaluation of the Maximum Entropy POS Tagger

In order to get a view of Maximum Entropy Performance on Telugu, the tagger is tested on five storied limited Telugu domains (told by Satya Sai Baba of Puttaparthi) meant for children. The output of the Maximum Entropy is manually edited and wrong taggings are marked. The accuracy percentage is calculated as follows.

Accuracy = Number of correct tagged words / Total number of tags.

Story No.	Total Number of Sentences	Total Number of Words	No. of Words Tagged Correctly	Accuracy Percentage
Story1	20	199	186	93.46
Story2	20	146	126	86.30
Story3	25	248	209	84.27
Story4	27	233	210	90.13
Story5	28	236	193	81,78

Table 1. Performance Details of Maximum Entropy Tagger

2.6.1. Error Analysis of Maximum Entropy Tagger

The performance of Maximum Entropy is shown herein in Figure-2.

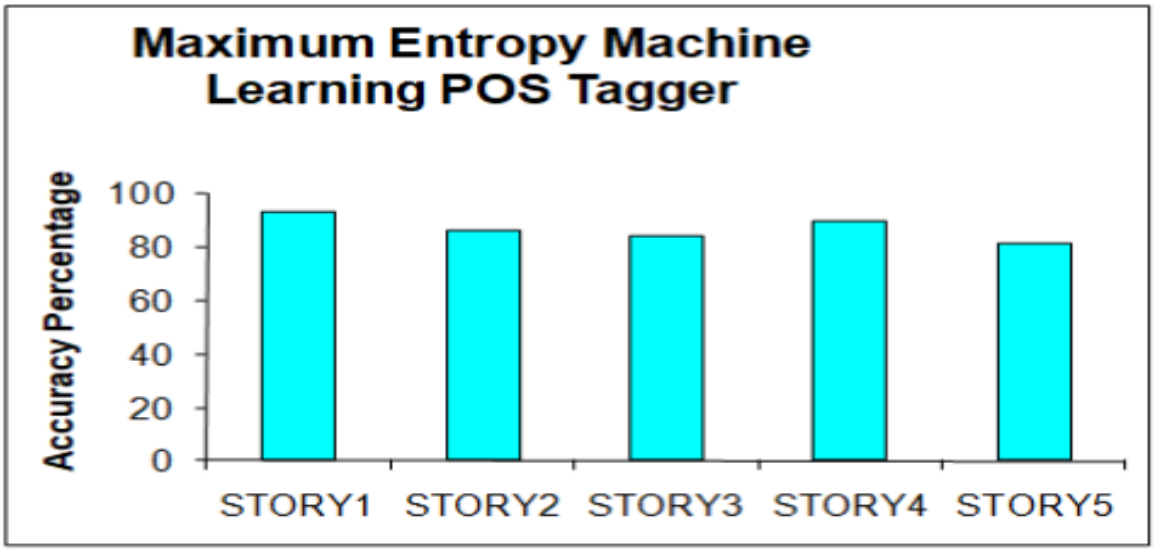

Figure 2. Performance of Maximum Entropy Telugu tagger

The Maximum Entropy is very good at assigning *karaka* roles but it is failing to recognize the verbal inflections which are deceptive in appears like '*naxi*' (looks more like a verb, if it occurs at the end, if 'naxi' does not occur in training corpus as noun inflection).

2.6.2. Comparison of Performance of Maximum Entropy Tagger When Adapted for Telugu and other languages

The performance of Telugu Maximum Entropy tagger with the other Maximum Entropy taggers is shown in Table 2.

Language	Accuracy
Telugu	87.18
English	96.60

Language	Accuracy
Swedish	91.20
Hindi	82.22

Table 2. Performance of Telugu and Other Existing Maximum Entropy Taggers

Maximum Entropy POS tagger [Rat96] when tested and adapted for English and Swedish [Beat], the accuracies of tagging are found to be 96.6% and 91.20% respectively. The accuracy of the Hindi POS tagger [AnKuUmSa] using Maximum Entropy tagger was found to be 82.22%.

2.7. Conclusion

The Maximum Entropy is adapted and tested for Telugu. The average performance is not encouraging. This is due to the misleading suffixes of verbs. It is doing well for nouns. As this algorithm is doing extremely well for European agglutinative languages, there is a dire need to improve the performance of this algorithm for the Indian context.

Reference:

[AdStVi96] Adam Berger, Stephen A, Della Pietra, Vincent J Della Pietra. 1996. A Maximum Entropy Approach to Natural Language Processing. Computational Linguistics.

[Ank] Aniket Dalal, Kumar Nagaraj, Uma Sawant, Sandeep Shelke. Hindi Part-of-Speech Tagging and Chunking : A Maximum Entropy approach. IIT, Bombay.

[Arvi96] Arvi Hurskainen. 1996. **Disambiguation of Morphological analysis in Bantu Languages**. Proc. CoLing'96. ICCL, Copenhagen. 568-573.

[Beat] Beat Megyesi. Data Driven Methods for POS tagging and Chunking of Swidish.

[BlAnRu] E z r a Black, Andrew Finch, Ruigiang Zhang. **Applying Extra sentential Context to Maximum Entropy Based Tagging with a large semantic and syntactic Tag set.** ATR Interpreting Telecommunications laboratories, 2-2 Hikaridai Seika-cho, Soraku-gun.

[Brill95b] Brill E. 1995. Transformation-Based Error-Driven Learning and Natural Language Processing: A case study in Part of Speech Tagging. In Computational Linguistics, vol. 21, no 4, pp. 543-565.

[DeJa] Deaniel Jurafsky, James H. Martin. Speech and Language Processing: An introduction to Natural Language Processing, Computational Linguistics and Speech Recognition. Pearson Education series in Artificial Intelligence.

[JiXl02] Jian Zhao, Xlao-LongWang. 2002. **Chinese POS Tagging based on Maximum Entropy Model**. Proceedings of the First International Conference on Machine Learning and Cybernetics, Beijing.

[MxEnTk] Zhang Le. **Maximum Entropy Modeling Toolkit for Python and C++.**

[RaRoSa93] Ray Lau, Ronald Rosenfeld, Salim Roukos. 1993. Adaptive Language Modeling Using the Maximum Entropy Principle. In proceedings of the Human Language Technology Workshop, pp 108-113. ARPA.

[Rat96] RatnaParkhi A. 1996. A Maximum Entropy Model for Part-of-Speech Tagging. Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-96) Philadelphia,PA, USA.

[Rat98] RatnaParkhi A. 1998. **A Maximum Entropy Models for Natural Language ambiguity Resolution.** Ph.D. thesis., University of Pennsylvania, Philadelphia, PA, USA.

[She97] Shemtov H. 1997. **Ambiguity Management in Natural Language Generation**. PhD Thesis, Stanford University, U.S.A.

[SkBr98] Skut W, Brunts T. 1998. **A Maximum Entropy Partial parser for unrestricted text.** In proceedings of the 6th Workshop on very large corpus (VLC-980, PP 143-151, Montreal, Canada, ACL.

[StViJo95] Steven Della Pietra, Vincent Della Pietra, John Lafferly. 1995. Inducing Features of Random Fields. Technical Report CMU-CS95-144, School of Computer Science, Carnegie-Mellon University.

[Uma04] Umamaheshwara Rao G. 2004. A Telugu Morphological Analyser. Paper presented at the National Seminar on Language Technology Tools & Implementation of Telugu; Vol – I.Phonology and Moprhology.CALTS, University of Hyderabad, Hyderabad.

[Vou93] Voutilainen Atro, Pasi Tapanainen. 1993. **Ambiguity resolution in a reductionistic parser**. Proceedings of the Sixth Conference of the European Chapter of the Association for Computational Linguistics, pp 394-403, Utrecht, the Netherlands, April. Association for Computational Linguistics.

[Vou94] Voutilainen Atro. 1994. Designing a Parsing Grammar. Publications no. 22, Department of General Linguistics, University of Helsinki, Finland.

[Vou95] Voutilainen Atro. 1995. Morphological disambiguation. Karlsson et al. chapter 6, pp 165-284.

Citation: Ganesh, Sree (2010). Maximum Entropy Model For Sense Tagging, HindiTech: A Blind Double Peer Reviewed Bilingual Web-Research Journal, 1 (2), 9-23. URL: https://hinditech.in/maximum-entropy-model-for-sense-tagging/