International Journal of Engineering Research in Computer Science and Engineering (IJERCSE) Vol 3, Issue 3, March 2016

Post Tagging Using Neural Network

[1] Shivani Kundra, [2] Ridhi Kapoor
[1]Student, M.Tech CSE, [2] Assistant Professor
[1][2] Department of Computer Science, DAV University
Jalandhar, Punjab, India
[1] shivanikundra670@gmail.com [2]Ridikapoor89@gmail.co

Abstract: **A cost-effective method for Part-Of-Speech (POS) tagging of a Thai corpus using neural networks is proposed. Computer experiments show that this method has a success rate of over 80% for tagging text of untrained data, and an error rate below 8%. These results are much better than those obtained by conventional table lookup methods. Some experiments comparing original and various modified back-propagation algorithms for training the neural network tagger are also conducted. Results of these experiments show that the learning algorithm with DBDB adaptation rule at a semi-batch update mode is the best one for tagging text in terms of convergence rate and computaional complexity.**

I. INTRODUCTION

Language models were initially developed for speech recognition and machine translation problems. They are used in other natural processing applications like information retrieval, parsing and part of speech tagging. In this research, we proposed a language model for predicting part of speech tag based on the context of the word, using neural network. Part of speech tagging is a task of assigning the appropriate part of speech or lexical category to each word in a natural lan-guage sentence. It is an initial step in Natural Language Processing(NLP) and is useful for most NLP applications and has a diverse application domain such as Speech Recognition, Speech Synthesis, Grammar checking, Phrase chunker, ma-chine translation etc. The Primary task of the tagger is to tag words in the text with proper part of speech tag. But often, part of speech becomes ambiguous and for a word, multiple tags may exists. i.e. The mapping between the words to the tags in one-to-many. Part of Speech can done using linguistic rules, stochastic models ar a combination of both. In the rule based approach, a knowledge base of rule is developed by linguistic to de ne precisely how and where to assign the various part of speech tags. This approach has already been used to develop the POS taggers for di erent languages, we found that machine learning approaches have been implemented for POS tagging for corpus rich foreign languages, such as Second Order Markov models Based taggers, Neural Networks based taggers for English, for Portuguese, etc

In case of Indian Languages particularly Punjabi, we found very limited pa-pers and resources. Punjabi is not a corpus rich language like English. For English and other foreign languages machine learning based approaches have been applied successfully for POS tagging. In case of Indian languages specially Punjabi, a rules based tagger and an HMM based tagger have been proposed. So, we stud-ied various research papers for foreign and Indian languages mainly considering Neural Networks and is presented here. The successful implementation of part of speech tagger using neural networks for foreign languages and Indian languages had motivated us to use neural networks for developing a part of speech tagger for Punjabi.

1.1 *NATURAL LANGUAGE PROCESSING*

Natural language processing (NLP) is a eld of computer science, articial intelligence, and computational linguistics concerned with the interactions between computers andhuman (natural) languages. As such, NLP is related to the area of humancomputer interaction. Many challenges in NLP involve natural language understanding, that is, enabling computers to derive meaning from human or natural language input, and others involve natural language generation.

The goal of the Natural Language Processing (NLP) group is to design and build software that will analyze, understand, and generate languages that humans use naturally, so that eventually you will be able to address your computer as though you were addressing another person.

This goal is not easy to reach. "Understanding" language means, among other things, knowing what concepts a word or phrase stands for and knowing how to link those concepts together in a meaningful way. It's ironic that natural language, the symbol system that is easiest for humans to learn and use, is hardest for a computer to master. Long after machines have proven capable of inverting large matrices with speed and grace, they still fail to master the basics of our spoken and written languages.

The challenges we face stem from the highly ambiguous nature of natural lan-guage. As an English speaker you e ortlessly understand a sentence like "Flying planes can be dangerous". Yet this sentence presents di culties to a software program that lacks both your knowledge of the world and your experience with linguistic structures. Is the more plausible interpretation that the pilot is at risk, or that the danger is to people on the ground? Should "can" be analyzed as a verb or as a noun? Which of the many possible meanings of "plane" is relevant? Depending on context, "plane" could refer to, among other things, an airplane, a geometric object, or a woodworking tool. How much and what sort of context needs to be brought to bear on these questions in order to adequately disambiguate the sentence?

We address these problems using a mix of knowledge-engineered and statisti-cal/ machine-learning techniques to disambiguate and respond to natural language input. Our work has implications for applications like text critiquing, information retrieval, question answering, summarization, gaming, and translation. The gram-mar checkers in O ce for English, French, German, and Spanish are outgrowths of our research; Encarta uses our technology to retrieve answers to user questions; Intellishrink uses natural language technology to compress cellphone messages; Microsoft Product Support uses our machine translation software to translate the Microsoft Knowledge Base into other languages. As our work evolves, we expect it to enable any area where human users can bene t by communicating with their computers in a natural way. Most NLP applications such as information extraction, machine translation, sentiment analysis and question answering, require both syntactic and semantic analysis at various levels. Traditionally, NLP research has focused on developing algorithms that are either language-speci c and/or perform well only on closed-domain text. At Google, we work on solving these problems in multiple languages at web-scale by leveraging the massive amounts of unlabeled data on the Web. We support a number of Google products such as web search and search advertising.

1.2 PART OF SPEECH

A part of speech is a category of words (or, more generally, of lexical items) which have similar grammatical properties. Words that are assigned to the same part of speech generally display similar behavior in terms of syntaxthey play similar roles within the grammatical structure of sentencesand sometimes in terms of mor-phology, in that they undergo in ection for similar properties. Commonly listed English parts of speech are noun, verb, adjective, adverb, pronoun, preposition, conjunction, interjection, and sometimes article or determiner.

A part of speech particularly in more modern classi cations, which often make more precise distinctions than the traditional scheme does may also be called a word class,lexical class, or lexical category, although the term lexical category refers in some contexts to a particular type of syntactic category, and may thus exclude parts of speech that are considered to be functional, such as pronouns. The term form class is also used, although this has various con icting de nitions. Word classes may be classi ed as open or closed: open classes (like nouns, verbs and adjectives) acquire new members constantly, while closed classes (such as pronouns and conjunctions) acquire new members infrequently, if at all.

Almost all languages have the word classes noun and verb, but beyond these there are signi cant variations in di erent languages. For example, Japanese has as many as three classes of adjectives where English has one; Chinese, Korean and Japanese have a class of nominal classi ers; many languages lack a distinction between adjectives and adverbs, or between adjectives and verbs (see stative verbs). This variation in the number of categories and their identifying properties means that analysis needs to be done for each individual language. Nevertheless, the labels for each category are assigned on the basis of universal criteria.

1.3 ARTIFICIAL NEURAL NETWORKS

In machine learning and cognitive science, arti cial neural networks (ANNs) are a family of statistical learning models inspired by biological neural networks (the central nervous systems of animals, in particular the brain) and are used to estimate or approximate functions that can depend on a large number of inputs and are generally unknown. Arti cial neural networks are generally presented as systems of interconnected "neurons" which send messages to each other. The connections have numeric weights that can be tuned based on experience, making neural nets adaptive to inputs and capable of learning.

For example, a neural network for handwriting recognition is de ned by a set of input neurons which may be activated by the pixels of an input image. After being weighted and transformed by a function (determined by the network's designer), the activations of these neurons are then passed on to other neurons. This process is repeated until nally, an output neuron is activated. This determines which character was read.

Like other machine learning methods - systems that learn from data - neural networks have been used to solve a wide variety of tasks that are hard to solve using ordinary rule-based programming, including computer vision and speech recognition.

An Arti cial Neural Network (ANN) is an

information processing paradigm that is inspired by the way biological nervous systems, such as the brain, pro-cess information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly intercon-nected processing elements (neurones) working in unison to solve speci c problems. ANNs, like people, learn by example. An ANN is con gured for a speci c applica-tion, such as pattern recognition or data classi cation, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurones. This is true of ANNs as well. Neural networks (NN) often referred as arti cial neural networks (ANN) to distinguish them from biological neural networks, are modeled after the workings of the human brain. The neural network is actually an information processing system that consists of a graph representing the processing system as well as various algorithms that ac-cess that graph. As with the human brain, the neural network consists of many connected processing elements. The neural network, then, is structured as a di-rected graph with many nodes and arcs between them. The nodes in the graph are like individual neurons, while the arcs are their interconnections. Each of these processing elements functions independently from the others and uses only local data to direct its processing. This feature facilitates the use of neural networks in a distributed and/or parallel environment. The neural network approach, like decision trees, requires that a graphical structure be built to represent the model and then that the structure be applied to the data. The neural network can be viewed as a directed graph exist in an input layer, while the output nodes exist in output layer. The hidden nodes exist over one or more hidden layers. To perform the data mining task, a tuple is input through the input nodes and the output node determines what the prediction is. Unlike decision trees, which have only one input node, the neural network has one input node for each attribute value to be examined to solve data mining function. Unlike decision trees, after a tuple is processed, the neural net-work may be changed to improve future performance. Although the structure of the graph does not change, the labeling of the edges may change.

In addition to solving complex problems, neural networks can learn from prior applications. That is, if a poor solution to the problem is made, the network is modi ed to produce a better solution to this problem the next time. A major drawback to the use of neural networks is the fact that they are di cult to explain to end users. Also, unlike decision trees, neural networks usually work only with numeric data. We rst must determine structure of the graph. Since there are two important attributes , we assume that there are two input nodes. Since we are to classify into three classes, we use three output nodes. The number of hidden layers in the neural network is not easy to determine. In most cases, one or two is enough. The human brain can be described as a biological neural networkan interconnected web of neurons transmitting elaborate patterns of electrical signals. Dendrites receive input signals and, based on those inputs, re an output signal via an axon. Or something like that. How the human brain actually works is an elaborate and complex mystery, one that we certainly are not going to attempt to tackle in rigorous detail in this chapter. The good news is that developing engaging animated systems with code does not require scienti c rigor or accuracy, as weve learned throughout this book. We can simply be inspired by the idea of brain function.

In this chapter, well begin with a conceptual overview of the properties and features of neural networks and build the simplest possible example of one (a network that consists of a single neuron). Afterwards, well examine strategies for creating a Brain object that can be inserted into our Vehicle class and used to determine steering. Finally, well also look at techniques for visualizing and animating a network of neurons.

1.3.1 Multilayer Perceptron

A multilayer perceptron (MLP) is a feedforward arti cial neural network model that maps sets of input data onto a set of appropriate outputs. A MLP consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. Except for the input nodes, each node is a neuron (or processing element) with a nonlinear activation function.

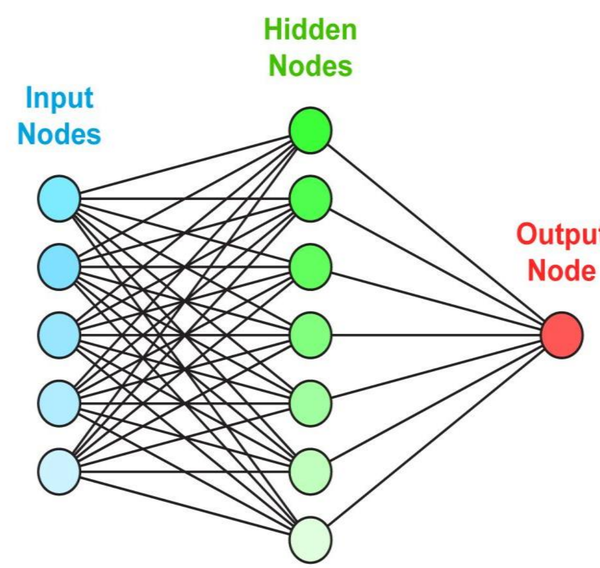

Figure 1.1: [2]Multi-Layer Neural Network

MLP utilizes a supervised learning technique called back propagation for train-ing the network. MLP is a modi cation of the standard linear perceptron and can distinguish data that are not linearly separable.

1.3.2 Activation Function

In computational networks, the activation function of a node de nes the output of that node given an input or set of inputs. A standard computer chip circuit can be seen as a digital network of activation functions that can be "ON" (1) or "OFF" (0), depending on input. This is similar to the behavior of the linear perceptron in neural networks. However, it is the nonlinear activation function that allows such networks to compute nontrivial problems using only a small number of nodes.

There are a number of common activation functions in use with neural net-works. This is not an exhaustive list.

1.4 APPLICATIONS OF PART-OF-SPEECH TAGGING

The POS tagger can be used as a preprocessor. Text indexing and retrieval uses POS information. Speech processing uses POS tags to decide the pronunciation. POS tagger is used for making tagged corpora.

Kamaljot Singh(11301077)

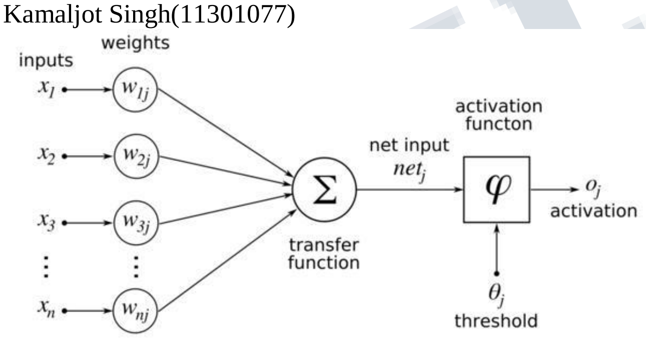

Figure 1.2: Activation Function

II. LITERATURE REVIEW

In the last decade, a great deal of scienti c researchers and studies have been performed on prediction, Brief descriptions of the papers are given below:trigramThis paper presents a novel approach of part of speech tagging using neural network for Punjabi language. To the best of our knowledge neural network never used for part of speech tags for Punjabi language. Language model was ini-tially developed for speech recognition and machine translation problem. They are used in another natural application like information retrieval, parsing and part of speech tagging. In this paper we proposed a language model for predicting part of speech tag based on context of word, neural network. Part of speech tagging is task of assigning appropriate part of speech or a lexical category to each word in a natural language sentence. It is the intial step in natural language processing and is useful for NLP applications and has diverse application domain such as speech reconition, speech synthesis , grammer checker,phrase chunker, machine transla-tion. The primary task of tagger is to tag words in the text with proper part of speech tag. Part of speech tagging can be done using linguistic rule, stochastic model or combination of both. I n this paper multilayer neural preceptron neural network tagger with xed length has been proposed for tagging of Punjabi text. The learning algorithm used for the proposed tagger is error back propagation learning algorithm. The tagged corpus was divided into training and testing data with randomize function. A feature vector was generated from training data by taking neighboring context for the current word. Trigram model has been used for generating this feature vector for every word in the training data .In this paper a multi layer preceptron neural network based part of speech tagger with xed context length is represented .Supervised learning approach with error back prop-agation algorithm is used for learning purposes. A feature vector is constructing considering the window size 2 for current word that is the previous two words along with their tags and next two words along with their tags. A feature vector is generated from dictionary prepared from training data. The proposed model gives better results as compa6red to existing taggers proposed for Punjabi. The tagger can handle unknown and ambiguous words very e ciently with higher accuracy. Considering the limited resources available for Indian languages particularly Pun-jabi the machine learning based approaches like neural network support vector machine could be a promising solution for part of speech tagging problem for Indian languages.

III. PRESENT WORK

3.1 ARCHITECTURE OF POS TAGGER

1. Tokenization: The given text is divided into tokens so that they can be used for further analysis. The tokens may be words, punctuation marks, and utterance boundaries.

2. Ambiguity look-up: This is to use lexicon and a guessor for unknown words. While lexicon provides list of word forms and their likely parts of speech, guessors analyze unknown tokens. Compiler or interpreter, lexicon and guessor make what is known as lexical analyzer.

3. Ambiguity Resolution: This is also called disambiguation. Disambigua-tion is based on information about word such as the probability of the word. For example, power is more likely used as noun than as verb. Disambiguation is also based on contextual information or word/tag sequences. For exam-ple, the model might prefer noun analyses over verb

analyses if the preceding word is a preposition or article. Disambiguation is the most di cult problem in tagging.

3.2 TAG SET

3.2.1 Evaluation Metrics

Results for part of speech tagging are obtained using confusion matrix (?). The confusion matrix is a speci c table layout that allows visualization of the performance of an algorithm, typically a supervised learning. Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. The name stems from the fact that it makes it easy to see if the system is confusing two classes (i.e. commonly mislabeling one as another).

The matrix contains 4 elds and number of di erent metrics are evaluated from them. The elds are True Positive(T_P), True Negative(T_N) also known as Type-II error, False Positive(F_P) also known as Type-I error and False Negative(F_N). The metrics that derive from confusion include:

3.2.1.1 Precision

Precision is the fraction of the documents retrieved that are relevant to the user's information need. It is also called positive predictive value(PPV):

$$\text{Precision} = \frac{T_P}{T_P + F_P} \quad (3.1)$$

3.2.1.2 Recall

Recall is the fraction of the documents that are relevant to the query that are successfully retrieved.

$$\text{Recall} = \frac{T_P}{T_P + F_N} \quad (3.2)$$

3.2.1.3 Accuracy

The quality or state of being correct or precise.

$$\text{Accuracy} = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \quad (3.3)$$

3.2.1.4 F-Measure

F-measure is a measure of a test's accuracy. It considers both the precision p and the recall r of the test to compute the score.

$$F_1 = 2\frac{P:R}{P + R} \quad (3.4)$$

Sr. No.	POS Main Category	POS Sub Category	POS Tag
1	Noun	Common Noun	NN
2	Noun	Proper Noun	NNP
3	Noun	Compound Noun	NNC
4	Noun	Compound Propoer Noun	NNPC
5	Pronoun	All categories	PRP
6	Adjective	All categories	JJ
7	Cardinal	-	QC
8	Ordinal	-	QO
9	Verb	Main Verb	VBM
10	Verb	First Person	FP
11	Verb	Second Person	SP