iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 3 **Issue:** III **Month of publication:** March 2015

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

Volume 3 Issue III, March 2015
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Chunked N-Grams for Sentence Validation

Priya Khatri[1], Garima Indolia[2]
[1,2]Computer Science Department, Maharaja Surajmal Institute of Technology, New Delhi, India

***Abstract**: This paper deals with Sentence Validation - a sub-field of Natural Language Processing. Sentence Validation refers to verification of "Natural Language" sentence on basis of its syntax and semantics. Sentence Validation is usually carried either via Statistical Means (n-grams) or by Semantic Means (by constructing some kind of Knowledge graph). We have tried to integrate the two approaches. Instead of directly using statistical methods, our sentences under-went semantic pre-processing. The results for direct statistical and semantically pre-processed approaches are then compared.*

I. INTRODUCTION

NLP is a field of Computer Science and linguistics concerned with interactions between computers and human languages. NLP is referred to as AI-complete problem. Research into modern statistical NLP algorithms require understanding of various disparate fields like linguistics, computer science, statistics, linear algebra and optimization theory. To understand NLP, we have to keep in mind that we have several types of languages today : Natural Languages such as English or Hindi, Descriptive Languages such as DNA, Chemical formulas etc, and artificial languages such as Java, Python etc. We define Natural Language as a set of all possible texts, wherein each text is composed of sequence of words from respective vocabulary. In essence, a vocabulary consists of a set of possible words allowed in that language. NLP works on several layers of language: Phonology, Morphology, Lexical, Syntactic, Semantic, Discourse, Pragmatic etc. Sentence Validation finds its applications in almost all fields of NLP - Information Retrieval, Information Extraction, Question-Answering, Visualization, Data Mining, Text Summarization, Text Categorization, Machine and Language Translation, Dialogue And Speech based Systems and many other one can think of. Statistical analysis of data is the most popular method for applications aiming at validating sentences. N-gram techniques make use of Markov Model[4]. For convenience, we restrict our study till trigrams which are preceded by bigrams. Results of this approach are compared with results of Chunked-Off Markov Model, which we developed to overcome some of limitations of standard Markov Model.
In the remaining of this paper, we first discuss about Markov Model (Section Two), to then describe Chunking Process (Section Three). The Methodology used is then explained (Section Four) followed by Results (Section 5). Lastly Future Scope is discussed (Section 6).

II. MARKOV MODEL

In probability theory, a Markov model[1] is a stochastic model used for modeling randomly-changing systems in which it is assumed that future states depend only on the present state and not on the sequence of events that preceded it (that is, it assumes the Markov property). Generally, this assumption enables reasoning and computation with the model that would otherwise be intractable. Markov Chain is the simplest Markov Model. It characterizes a mathematical system that undergoes transitions from one state to another on state space. It is usually memory-less, i.e. the chain depends only on current state and states preceding it.

$$p(D) = p(x_1) \prod_{n=2}^{N} p(x_n | x_{n-1}) \qquad (1)$$

In our model, we use "n-grams". An n-gram model depicts probabilistic model for predicting next item in sentence using (n-1) order Markov model. This model is highly successful and is in wide use today. It has two key advantages: High Scalability and Relative Simplicity.

III. CHUNKING IN SENTENCE VALIDATION

Using Markov Model, we got the statistical nature of language and can capture that well. However we realize that this model is not at all good at "understanding" language. For example, if "John eats" is valid example, then "Mary eats" should be valid too. markov Model fails to capture this.

©IJRASET 2015: All Rights are Reserved

Volume 3 Issue III, March 2015
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Therefore, we proposed to Chunk the sentences in different fragments. For the purpose of our model, we only fixed our attention on "Named Entities" since we felt that biggest pay-off will occur there. Chunking refers to labeling of multi-token sequences as one segment. There are several types of chunking, each following a different method of segmenting and grouping the text.

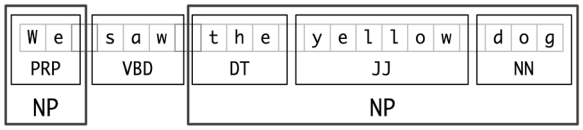


Fig. 1. Noun Phrase Chunking [2]

Chunking is often done after tagging each token with meaningful information. (Example : in Fig. 1, Noun Phrase Chunking is done after each Token is tagged with POS Tag).

IV. OUR METHODOLOGY

The statistical approach use the N-gram technique and Markov Model building. In the standard statistical Markov N-gram Model, corpus data is fed into the database in the form of bigrams and trigrams with their respective frequencies(i.e. how many times they occur in the whole data set of sample sentences). When an input sentence is to be validated, it is tokenized into bigrams and trigrams which are then matched with database values and a cumulative probability after application of Smoothing-off technique of Kneser-Ney Smoothing which handles new words and zero count events having zero probability which may cause system crash, is calculated. [4]

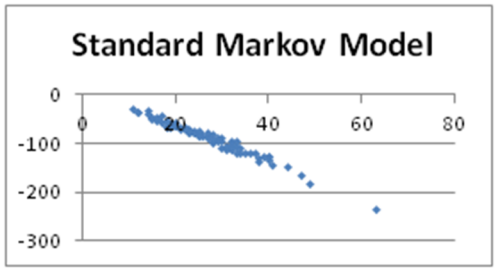


Fig. 2 Standard Markov Model Semi-log Graph [Length v/c Probability]

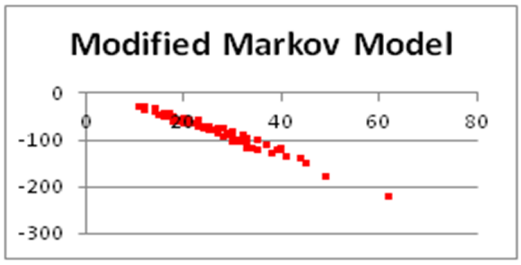


Fig. 3. Modified Markov Model Semi-log Graph [Length v/s Probability]

Chunked-Off Markov Model[5] makes use of our defined replace function implemented through pos_tag and ne_chunk functionality of NLTK. Every sentence is first tagged according to Part-Of-Speech using pos_tag. Whenever a 'NN', 'NNP' or in general 'NN*' chunk is encountered[3], it is passed to ne_chunk which replaces the named entity with its type and returns a modified sentence whose bigrams and trigrams are generated and fed into the database. The testing procedure of this approach also modifies the sentence entered by the user, calculates the probabilities of the bigrams and trigrams by matching them with database entries and finally smoothing off to yield final results.

©IJRASET 2015: All Rights are Reserved

Volume 3 Issue III, March 2015
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

V. RESULTS

All Models used the same database for testing and training purposes. From the same corpus, 85% data was used for training and rest for testing. This has two advantages - firstly we shall use the same ratio in all other approaches so that it is easier to compare them. Secondly it provides a threshold value for probability which will help us to distinguish between correct and incorrect test sentences depicting regions above and below threshold respectively. Graphs are plotted between probability(exponential, in order of 10) and length of the sentence(number of words). The results (seen in Fig. 4.) show a consistent and remarkable difference between the models - Chunked-Off Markov Model gave better responses than Standard Statistical Model for the same corpus and test data. Also, we see that chunked off responses are a lot more consistent than Standard Statistical ones. This may be due to its ability to deal with Proper Nouns. While in Pure Markov, any new proper noun will lower the probability significantly, in Chunked Markov Model, it may not be case.

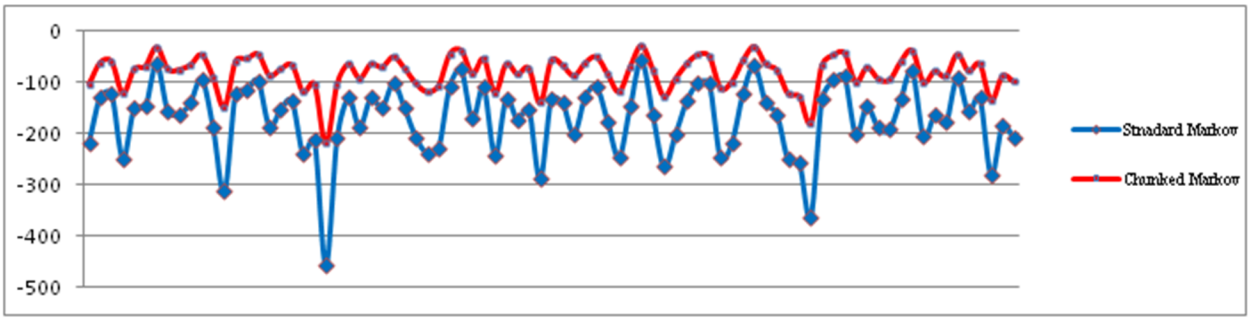


Fig. 4. Comparison between Standard Markov Model and Chunked off Markov Model

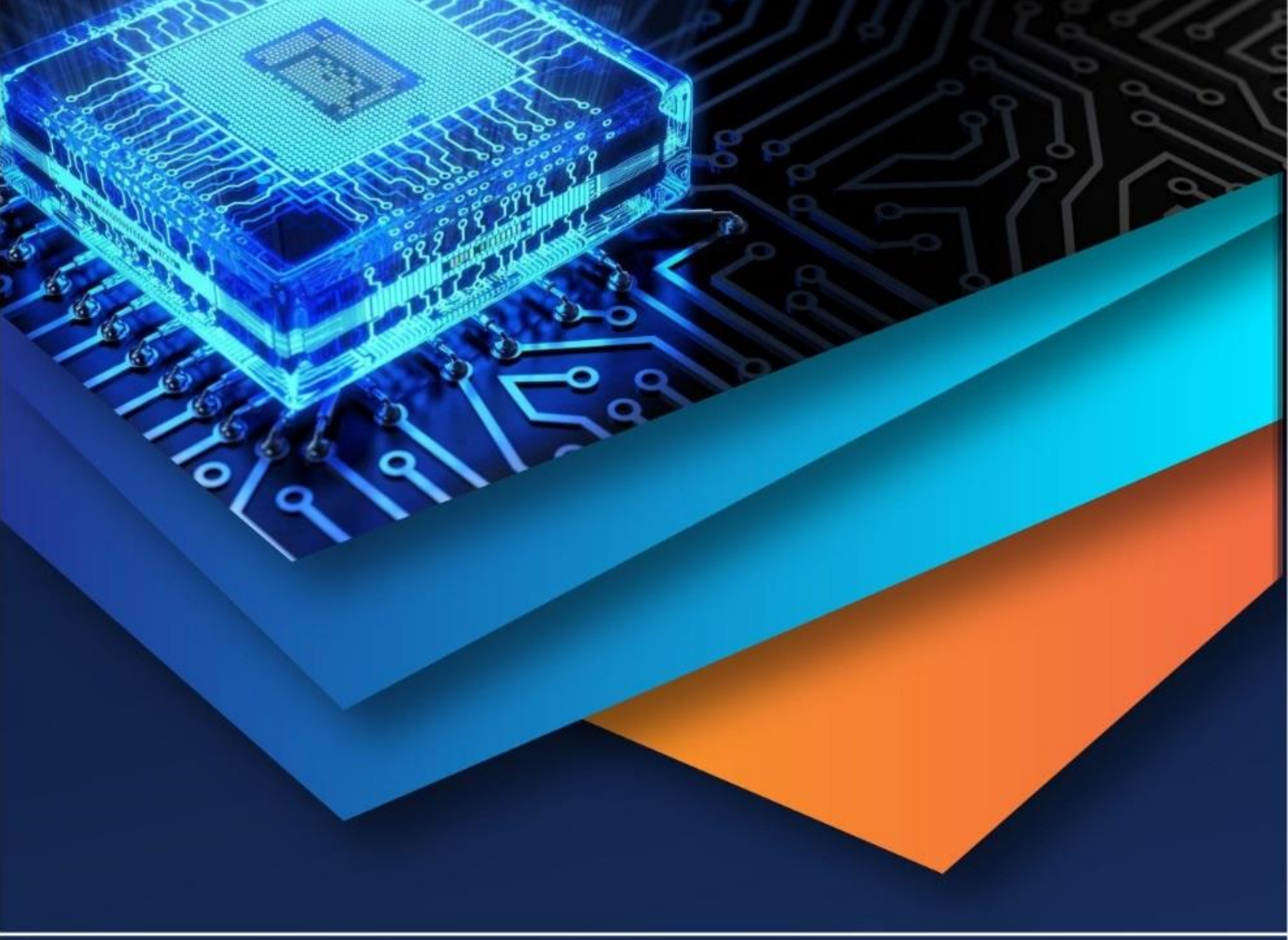
VI. CHALLENGES AND FUTURE OF SENTENCE VALIDATION

NLP is often referred to as AI-complete problem, since the various challenges associated with this field are so numerous and complex, that solving them is equivalent to solving the AI field. Language Modeling is the first step into this arena and any progress made here affects all other sub-fields of NLP. With fully fledged Semantic graph which can explain all relations and entities still remain pipe-dream, modification in n-grams to make use of semantic progress is viable field. In future, we envision, n-gram tokens moving beyond just lexical units and having different dimensions. A good Language Model even might make multiple tokenization efforts and assign probabilistic values to them. This effort is incorporating semantic research in statistical efforts. We fundamentally believe that progress in other direction, i.e. making use of Statistical means to develop Semantic Relations is extremely important. While it may be difficult for humans to develop complete Entity-Relation chart and Entities Ontology or computer which humans intuitively understand, we may try to make computer develop some rules based on statistical parsing.

REFERENCES

[1] "*SCOPE 34 - Practitioner's Handbook on the Modeling of Dynamic Change in Ecosystems, Chapter 6, Markov Models and Related Procedures.*" Retrieved from http://www.scopenvironment.org [Sep 2014]
[2] *"NLTK Documentation*" Retrieved from http://www.nltk.org/book [Aug 2014]
[3] *"Python Docs: Regular Expressions"* Retrieved from https://docs.python.org [Oct 2014]
[4] Chen, Stanley F., and Joshua Goodman. "An empirical study of smoothing techniques for language modeling." *Computer Speech & Language* 13.4 [1999]
[5] Rosenfield, Roni. "*Two decades of statistical language modeling: Where do we go from here?*." [Oct 2000]

©IJRASET 2015: All Rights are Reserved



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)