International Journal of Engineering &Technology, 8 (4) (2019) 590-595

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

An optimal method for enhancing the generation of machine code from natural language data set

Chhayarani Ram Kinkar [1] *, Yogendra Kumar Jain [2]
[1] *Electronics and communication Department, Smarat Ashok Technological Institute Civil lines, Vidisha, India*

[2] *Electronics and communication Department, Smarat Ashok Technological Institute Civil lines, Vidisha, India*
**Corresponding author E-mail: Chhayakinkar@gmail.com*

Abstract

Natural language processing is a very active area of research and development, there is not a single agreed upon a method that would satisfy everyone for the use of natural language to operate electronic devices or other practical applications. But there are some aspects used from many years in the formulation and solution of computational problem arising in natural language processing. This paper describes a model in which numerical values are assigned to word of natural language speech data set to convert the information present in natural language speech data set into an intermediate numeric form as a structured data set. The intermediated numerical values of each word will be used for generation of machine code which will be easily understand by electronic devices to draw inferences from data set. The designed model is useful for a number of practical applications and very simple to implement.

Keywords: *Numerical Value; Utility Based Objectives; Constrained Based Decomposition; Structured Data Set.*

1. Introduction

Natural language processing is a motivated range of computational techniques for analyzing and representing naturally occurring text at one or more levels of linguistic analysis for the purpose of achieving human like language processing by electronic devices or other practical applications. To fulfill this goal the system must be able to

1) Paraphrase an input text or speech.
2) Translate the text into another language.
3) Answer the question about the content of the text.
4) Draw inferences from the text.

Natural language processing has made serious inroads into accomplishing goals 1 to 3, many algorithms for processing, speech and language data are developed to fulfill these goals [1-2-3-4]. To draw inferences from the text still remains the goal of natural language processing because it needs accurate translation of natural language speech data set into equivalent proper machine code in the form of 0 and 1,so that the device can understand the exact meaning. Research has been going on from several decades to fulfill this goal. Mathematical formulations to fulfill this goal depend on the parameters of the model constructed from statistical principles and loss function that quantify misfit between observed data and prediction [5].

When a model is constructed to full fill goal of drawing an inference from natural language speech data set accuracy is very important because the data set in language processing application consist of sentences of text, segment of speech from a particular speaker, a frame of speech, words, phrase, or labeled instances of phonemes[6]. The objective function for the mathematical formulation to fulfill this goal depends on parameters of the model and data set. The general form of this objective function is

$$\max \mathrm{O}(\Lambda) := \sum_{t=1}^{T} \mathrm{Ot}(\Lambda)$$

Where Λ denotes the variables or parameter which is generally collection of real numbers, real vectors and symmetric positive semi definite matrices,

O (.) denotes the overall objective which is a continues function mapping, open set of variables to a real number,

Ot denotes the partial objective corresponding to a single item of data [1].

This separability property places speech and language processing, formulation into some framework to draw inferences from natural language speech data set for practical application [5]. Loss function that quantify misfit between observed data and predictions in this problem depends on pronunciation lexicon which depends on the composition of an acoustic model for a given word.

Construction of mathematical model and loss function for drawing inferences from text or speech as a optimal model need to be convex that is a feature that allow strong convergence guarantees. Many optimization techniques are non convex in nature due to the complexity of language model which introduces probabilities of possible values.[6]

This paper describes an optimal technique based on utility objective, continuous value variables, statistical modeling approaches, including alphabet search, alignment for model based optimization framework and helpful for drawing inferences. The model has a hierarchical structure, parameter set, and statistical in nature. In this paper the key algorithms that have been developed to relate model based optimization for specific application are developed in python. Preference is given to python among many available programming languages because it has many in built modules to access underlying system calls from within python programs. Python can be also embedded into other applications to allow scripting [7]. Program maintenance cost is also low in python [8].

2. Related work

As a background this section discuss number of approaches developed by researchers related to natural language processing by devices and the mathematical or statistical formulation of natural language data processing problems.

Victor m sanchez, Jon Antnio, address the situation in which the amount of bilingual resources available for a particular language pair is not sufficiently large to train a competitive statistical MT system, but the cost and slow development cycles of rule-based MT systems can be afforded either. They formalize a new method that uses scarce parallel corpora to automatically infer a set of shallow-transfer rules to be integrated into a rule-based MT system, thus avoiding the need for human experts to handcraft these rules[3]

Rico Sennrich Philip,Willams mathias huck, discuss various ways in which string-to-tree translation models over- or under generalize. and shows show how these problems can be addressed by choosing a suitable parser and modifying its output, by introducing linguistic constraints that enforce morphological agreement and constrain sub categorization Synchronous context-free grammars (SCFGs) can be learned from parallel texts that are annotated with target-side syntax, and can produce translations by building target-side syntactic trees from source strings. Ideally, producing syntactic trees would entail that the translation is grammatically well-formed[5].

Antonio Toral, Pavel Pecina, Josefvan genabith explores the use of linguistic information for the selection of data to train language models. They depart from the state-of-the-art method in perplexity-based data selection and extend it in order to use word-level linguistic units that is lemmas, named entity categories and part-of-speech tags instead of surface forms. They present two methods that combine the different types of linguistic knowledge as well as the surface forms

- Naïve selection of the top ranked sentences selected by each method.
- Linear interpolation of the datasets selected by the different methods [10].

Mirjam Sepesy Maučec, Gregor Donaj investigate the role of morphology in phrase-based statistical machine translation from English to highly inflectional Slovenian language. They find that Translation to inflectional language is a challenging task, because of morphological complexity.Rich morphology increases data sparsity and worsens the quality of statistical machine translation.To address this issue, they added the morphological information in terms of MSD tags, that were attached to words. MSD tag includes all morphosyntactic in formation in position-dependent attributes. Tags were attached to words by TreeTagger. Several experiments were performed using MSD tags to improve the translation results. Different configurations were tested. They show that factored translation improves modeling of short distance collocations. To capture long-distance dependencies in languages, they adds OSM models in the second set of experiments. Additional9% relative improvement was obtained. [11].

Eva Hasler, Adrià de Gispert, Felix Stahlberg, Aurelien Waite, Bill Byrne shows that Long sentences with complex syntax and long-distance dependencies pose difficulties for machine translation systems. Short sentences, on the other hand, are usually easier to translate. They show that the spaces of original and simplified translations can be effectively combined using translation lattices and compare two decoding approaches to process both inputs at different levels of integration. and demonstrate on source-annotated portions of WMT test sets and on top of strong baseline systems combining hierarchical and neural translation for two language pairs that source simplification can help to improve translation quality[13].

Marta R Costa- Jussa and Jose A R Fonollosa done a survey on hybrid machine translation (MT) motivated by the fact that hybridization techniques have become popular as they attempt to combine the best characteristics of highly advanced pure rule or corpus-based MT approaches. They found that Exiting research typically covers either simple or more complex architectures guided by either rule or corpus-based approaches. They provide a detailed overview of the modification of the standard rule-based architecture to include statistical knowledge, the introduction of rules in corpus-based approaches, and the hybridization of approaches within this category [16]

It is recognized that modern techniques and approaches are still in developing stage due to complex and evolutionary nature of natural language. The approach describe in this paper is based on basic building block of any language that is alphabets. The alphabet and their occurring sequence are used in this approach for translation into an intermediate numeric form, from which language independent machine code is generated for efficient processing and understanding of natural language by the electronic devices.

3. Proposed model

A model for drawing inferences from natural language speech data set has received a growing interest over the last decade. The different concepts and algorithms have been investigated for this purpose. The improve quality model can be built with clear knowledge of language, translation direction and nature of corpus.

This section outlines the formulation of a mathematical model for the smooth conversion of natural language speech data set into an intermediate from of numerical values. These numerical values will be used by device for drawing inferences from the speech data set. The proposed model takes real time natural language commands as a input data set and process it for conversion into an intermediated from of numerical values. Which are father converted into equivalent machine code as illustrated in figure1.

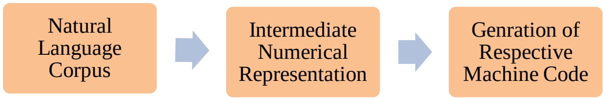
Natural Language Corpus
Intermediate Numerical Representation
Genration of Respective Machine Code

Fig. 1: Structure of Proposed Model.

The main aim in the design of model is to design an optimal model based on utility objective therefore in this model conversion of natural language speech data set is based upon the structural properties of grammar of natural language. A grammar consist of finite, large set of sentence formulation rule which are linear in nature. To build, improve quality model linguistic and morph- syntactic information of a word is analyzed in a strictly sequential manner.

3.1. Intermediate numeric representations

The proposed model deals with the computational nature of word therefore conversion into an intermediated numerical form is completed in two different stages as illustrated in figure2.

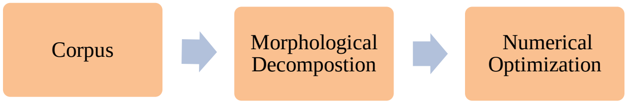

Fig. 2: Stages for Conversion Into Numerical Value

The first stage deals with morphological decomposition of compound word that is converted of receiving word into root word by removing the involved affixes and suffixes within the word boundary. The designed algorithm identifies a specific alphabet sequence, and categorizes it accordingly to respective morphemes.

The information regarding the separation of affixes and suffixes is convey to next stage in numeric form and the generated root word is transfer to next stage in its pure form.

The second stage deals with only relationship and mapping of root word with the equivalent numerical value. In proposed model already tagged corpus is considered and key algorithm to realize the objective functions are developed in python.

3.1.1. Morphological decomposition

The morphological information attached with word is not purely concatenative spelling rules make it complicated. A simple algorithm is not enough to split off affixes and suffixes. To get more accurate result an adhoc algorithm based on iterative line linear search is designed .The optimized formulation for the algorithm is describe in equation(1). Design model is based on utility objective therefore objective function for morphological decomposition have four fixed numerical values R_{11},R_{12},R_{21},R_{22},corresponding to affixes/suffixes attached with root word. The adhoc nature of algorithm separate affixes/suffixes simultaneously and it is describe as follows.

$$\Delta(\upsilon) := \{ \emptyset(\check{S}) \in \sum_{i,j=1}^{i,j=2} \mathrm{Ri, j} \} \quad (1)$$

Where

$\Delta(\upsilon)$ is a continuous function mapping $\emptyset(\check{S})$ in to a numerical value Ri,j

$\emptyset(\check{S})$ is specific affix/suffix sequence attached with root word as a fast/last alphabets, and $\emptyset(\check{S})$ is define as

$\emptyset(\check{S}) = \sum_{i=0}^{3} \hat{a}\mathrm{i}$

Where

$\hat{a}_0$, $\hat{a}_1$, $\hat{a}_2$, $\hat{a}_3$ are four allowable constrains of$\emptyset(\check{S})$, and are array of fix alphabets sequence of variable length.

ñ denotes total length of word with affix/suffix, ń denotes length of root word.

$\Delta(\upsilon)$ is convex in nature, and incorporate bounds on numerical value as a mapping of $\emptyset(\check{S})$,therefore search for specific alphabet sequence start from last alphabet to ñ-3 alphabets to speed up the mapping of values. This generates a sequence of iterates {σk}k= ñ-1, ….. ñ-3. where σk is line search parameter restricted to interval ñ-1 to ñ-3 . Then matching function decides respective numerical value for true value of linear line search parameter and array function for proper mapping. Adhoc nature of algorithm work in same manner for affixes. The algorithm for proposed mapping function is work in following manner.

```
{
Read the word and store it in a array
Calculate the length of array
ñ= array length
for the interval ñ-1 to ñ-3
Scan the alphabet and store them in σk
σk={ }
if
{
σk ϵ $\emptyset(\check{S})$
case 1
σk = $a_o$
σk := R11
case 2
σk = a1
σk := R12
case 3
σk = a2
σk := R21
case 4
σk = a3
σk := R22
else
σk := 00
}
if
{
σk := value of Rij
remove alphabets in σk from the word and store root word in an array
ń= root word length
else
ń= ñ
}
Map calculated value of σk with morphological function Δ(υ)
}
```

In the above optimized formulation only four cases of suffix/affixes related to noun, verb, tenses are considered. All special cases of plural nouns, verbs, adjective nouns are not considered because of utility based objective. The designed algorithm also takes care that if it is removing any affixes/suffix from received word, then information regarding it is given to the next stage for correct processing natural language speech data set.

3.1.2. Numerical optimisation

Natural language consists of a finite set of alphabets. The set of word is infinite, but set of meaning full word is finite. The size of this set is too large. In a similar way the set of sentences is infinite, but the set of meaning full sentence is finite with large size. Utility based objective function reduces the size of this large set with a limited combination of only those words which are related to practical application of electronic devices. In this case we can define

$$\mathbb{C}= \sum_{ń=1}^{Ň} F_{ń}^{J}$$

Where

$\mathbb{C}$ de notes corpus of natural language speech data set containing sentence related with operation of device, which is a finite set of Ň elements

$F_1^J, F_2^J, \dots \dots \dots . F_{ń}^J$ are the individual sentences.

J is the number of word in a sentence.

The goal of design model is automatic translation of $F_1^J = F_1 + F_2 + \cdots \dots . . F_J$ in to $E_1^J = E_1 + E_2 + \cdots \dots . . E_J$

Where

F_1^J de notes sentence of natural language containing word F_1, F_2……F_J

E_1^J is the equivalent numerical translation of F_1^J

E_1, E_2…E_J are the numerical value of F_1, F_2……F_J respectively.

The conversion is done in word order sequence, one to one correspondence between word and its numerical value. Each word of F_1^J is scan and total length of each word is calculated. The calculated length of respective word is stored. This conversion is explained below in figure 3.

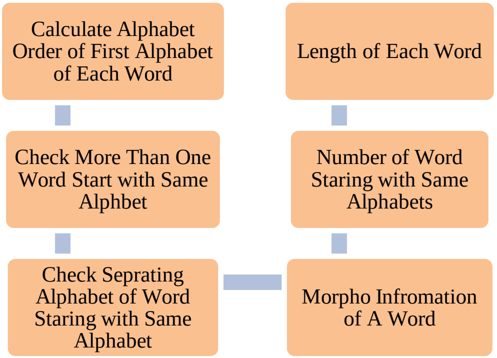
Calculate Alphabet Order of First Alphabet of Each Word
Length of Each Word
Check More Than One Word Start with Same Alphbet
Number of Word Staring with Same Alphabets
Check Seprating Alphabet of Word Staring with Same Alphabet
Morpho Infromation of A Word

Fig. 3: Numerical Value Assignment.

For each word of F_1^J a equivalent numerical value is calculated as follows.

$$E_1^J = \propto_1 \propto_2 \propto_3 \propto_4 \propto_5 \propto_6$$

$\propto_1$ is related with alphabetical order of first alphabet of a word in $\mathrm{F}_1^{\mathrm{J}}$. To calculate this value alphabet at 0th location of a word is scan by algorithm.. The allowable range for assigning numerical value to $\propto_1$ is two digits.

$\propto_2$ is related with word starting with same alphabets in $\mathrm{F}_1^{\mathrm{J}}$,ideally its value is zero, but if $\mathrm{F}_1^{\mathrm{J}}$ contain two or more word starting with same alphabets then its value become non zero. Its value indicates up to which location alphabets in both the words are same. To assign the value of $\propto_2$ the design algorithm scan a word from 0th location to ń/2 location. Allowable range for $\propto_2$ is single digit.

$\propto_3$ is related with that alphabet, which separate the words starting with same alphabets .Ideally, this value is zero, but assigns a value if $\propto_2$ assign a non zero value. Its value indicates alphabetical order value of that alphabets from which both the words are differentiating from each other. The allowable range of digits for $\propto_3$ is depend on the value of $\propto_2$, for example for two similar word a 4 digit value is assign for $\propto_3$, for 3 similar alphabets a 6 digit value is assign for $\propto_3$and so on .The relationship between value of $\propto_2$ and $\propto_3$ is given as 2’n where ’n is number of similar alphabets. The allowable range for is $\propto_3$ variable digits.

$\propto_4$ is used for morphological information attached with the word. If received word is in its purest form then the value of $\propto_4$ is 00, otherwise its value is decided by morphological decomposition mapping function $\Delta(\upsilon)$. Allowable range for is $\propto_4$ two digits.

$\propto_5$, is related with same words of $\mathrm{F}_1^{\mathrm{J}}$, ideally value assign to $\propto_5$ is zero, but if $\mathrm{F}_1^{\mathrm{J}}$ has two or three word starting with same alphabets then its value indicates how many word in $\mathrm{F}_1^{\mathrm{J}}$ are starting with same alphabet. The allowable range for $\propto_5$ is one digit.

$\propto_6$, is related with word length. Each word of $\mathrm{F}_1^{\mathrm{J}}$ is scan and total length of each word is calculated. The calculated length of respective word is stored in $\propto_6$.

Any word in natural language does not have all 6 $\propto$ equivalent values. This numerical value is converted in to equivalent machine code to draw inference from natural language speech data set.

4. Result and discussion

The second stage of this proposed model converts natural language speech data set into an equivalent numerical value according to objective function required for intermediate representation of natural language which is define as

$$\max \mathrm{O}(\wedge) := \sum_{\mathrm{t}=1}^{\mathrm{T}} \mathrm{Ot}(\wedge)$$

Where

$\wedge$ is collection of real numbers, and symmetric positive semi definite matrix.

This matrix generated by proposed model is shown in figure4.

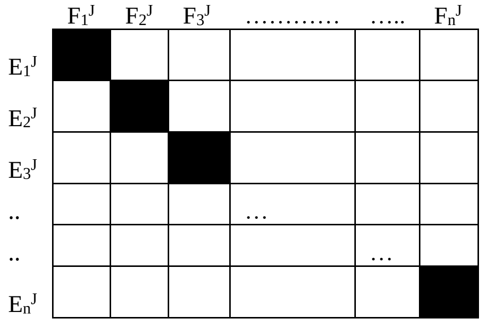

Fig. 5: Individual Item Numeric Value.

O(.) is a continues mapping function, mapping sentences in to a real number, and

Ot is the partial objective corresponding to a single item of data are shown in figure 5

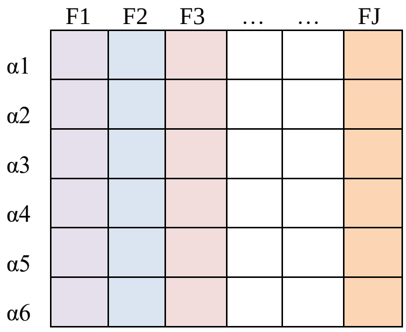

Fig. 5: Individual Item Numeric Value.

These matrixes show that each alphabet,word,sentence of natural language speech data set assign a unique value.

Number of languages exists in the word for communication among human beings. Human mind is a powerful machine and able to understand more than one language for communication. In concern with electronic device, it is difficult to inbuilt more than one language compiler for generation of equivalent machine code. A common methodology for intermediated representation for translation into equivalent machine code is necessary The approach described in this paper is based on basic building block of any language that is alphabets which is a set of non empty symbols ,which are written in specific order. By assigning numerical value to these symbols in a specific order a sequence of numerical number is generated which will be use for the generation of equivalent machine code The propose approach is very simple to implement and gives better translation in less time. The numerical value plays crucial role in generation of high quality, fluent machine code.

5. Conclusion

Language recognition or language understanding is an important area of natural language processing where optimization techniques and methods play an important role. If these optimization techniques are equipped with strong theoretical properties which serve well for a unified treatment for range practical applications then the task become an easy task. Many translation software are designed for processing of natural language for by electronic device, but none of them provide a perfect and dynamic solution because of evolutionary nature of language. Looking at the high level of accuracy this numeric approach is much better because it uses the idea of alphabetical which are written in a specific order in any natural language. Numerical translation also solves the problem of linguistic irregularities, ambiguities, lack of grammar knowledge in devices. To achieve high accuracy in generation of machine code from natural language speech data set. A need of standardization model with high accuracy is fulfill by this model.

References

[1] T.Sainath,B.Kingsbury,H.Soltau,andB.Ramabhadran,"Optimization techniques to improve training speech of deep belief networks for large speech tasks,"IEEETrans.Audio,Speech,Lang.Process.,Spec.Iss. Nov. 2013. vol. 21 pp 2331-2342. https://doi.org/10.1109/TASL.2013.2284378.

[2] Stephen J. Wright, Dimitri Kanevsky, Li Deng , Xiaodong He "Optimization Algorithms and Applications for Speech and Language Processing" IEEE Trans. Audio, Speech, Lang. Process., Nov. 2013, vol. 21, no. 11, pp. 2231–2242. https://doi.org/10.1109/TASL.2013.2283777.

[3] Victor m sanchez , jon antnio" A generalised alignment template formalism and its application to the inference of shallow-transfer machine translation rules from scarce bilingual corpora" computer speech and language,july 2015,vol 32,issue.1 pp 46-90. https://doi.org/10.1016/j.csl.2014.10.003.

[4] V.Hautamäki,K.A.Lee,T.Kinnunen,B.Ma,andH.Li,"Optimizing the performance of spoken language recognition with discriminative training," IEEE Trans. Audio, Speech, Lang. Process., Aug. 2013, vol. 21, no. 8, pp. 1622–1631.

[5] Rico Sennrich Philip,Willams mathias huck" A tree does not make a well-formed sentence: Improving syntactic string-to-tree statistical machine translation with more linguistic knowledge" computer speech and language ,july 2015 vol 32,issue.1pp 27-45. https://doi.org/10.1016/j.csl.2014.09.002.

[6] Li Deng, Xiao Li, "Machine Learning Paradigms for Speech Recognition: An Overview" IEEE Trans. Audio, Speech, Lang. Process., May. 2013, vol. 21, no. 5, pp. 1060–1089. https://doi.org/10.1109/TASL.2013.2244083.

[7] Steven Bird, Ewan Klein, Edward loper "Natural Language Processing in Python" OREILLY.

[8] Magnus Lie Hetdmol "Python Algorithm" APRESS.

[9] Marta R Costa- Jussa and jose A R Fonollosa" Latest trends in hybrid machine translation and its applications" computer speech and language ,july 2015, vol 32,issue.1pp 3-10. https://doi.org/10.1016/j.csl.2014.11.001.

[10] Antonio Toral, Pavel Pecina, Josefvan genabith "Linguistically-augmented perplexity-based data selection for language models" computer speech and language ,july 2015.,vol 32,issue.1pp 11-26. https://doi.org/10.1016/j.csl.2014.10.002.

[11] Mirjam Sepesy Maučec, Gregor Donaj" Morphology In Statistical Machine Translation From English To Highly Inflectional Language" Journal of information technology and control 2018,vol 47 no.1 pp 63-74. https://doi.org/10.5755/j01.itc.47.1.17887.

[12] Tetsuo Sawaragi, Sosuke Iwai, Osamu Katai ,"A Human-Friendly Interface System for Decision Support Based on Self-Organized Multi-Layered Knowledge Structures" Toward Interactive and Intelligent Decision Support Systems, Volume 286 of the series Lecture Notes in Economics and Mathematical Systems pp 30-39. https://doi.org/10.1007/978-3-642-46609-0_4.

[13] Eva Hasler, Adrià de Gispert, Felix Stahlberg, AurelienWaite, Bill Byrne" Source sentence simplification for statistical machine translation" computer speech and language, sept.2017, vol 45,pp 221-235. https://doi.org/10.1016/j.csl.2016.12.001.

[14] Wright, S.J. Kanevsky, D. ; Li Deng ; Xiaodong He" Optimization Algorithms and Applications for Speech and Language Processing "Audio, Speech, and Language Processing, IEEE Transaction 2013 ,Volume:21,Issue: 8. https://doi.org/10.1109/TASL.2013.2283777.

[15] Felipe Sánchez-Martínez."Choosing the best machine translation system to translate a sentence by using only source-language information". In Proceedings of the 15th Annual Conference of the European Associtation for Machine Translation, May 2011.