Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 167 (2020) 918–931

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Computational Intelligence and Data Science (ICCIDS 2019)

Generative Model for NLP Applications based on Component Extraction

Anupam Bhardwaj[a,] *, Pooja Khanna[a], Sachin Kumar[a], Pragya[b]

[a]*Amity University, Lucknow Campus*
[b]*MVPG College, Lucknow University*

Abstract

People all around the world speak so many different languages, but a Computer System or any other Computerized Machine only understands a single language i.e. binary language (1s and 0s) This system or a process that converts human language to computer understandable language is known as Natural Language Processing (NLP), though various diversified models have suggested so far , yet the need for a generative predictive model which can optimize depending upon the nature of problem being addressed is still an area of research under work. The paper presents a Generative Model for NLP Applications based on significant components extracted from Case Studies. The generative model is a single platform for diversified areas of NLP that can address specific problems relating to read text, hear speech, interpret it, measure sentiment and determine which parts are important. This is achieved by process of elimination once the relevant components are identified. Single platform provides same model generating and reproducing optimized solutions and addressing different issues.

Keywords: Natural Language Processing; Behavior signal processing; Stroke based classification; Requirement Processing.

1. Introduction

Natural Language Processing a sub-section of Artificial Intelligence that is used for the analysis, understanding and production of language which is commonly spoken by humans to communicate with Computers and Smart Machines. Fig. 1. shows the relationship of NLP with Artificial Intelligence and Machine Learning. NLP is one of

* Corresponding author. Tel.:+91- 8171446311
E-mail address: bhardwajanupam1@gmail.com

10.1016/j.procs.2020.03.391

the methodologies of text mining used for text analysis that implements a unique kind of linguistic analysis that basically supports the machine to read. A variety of methodologies are used in NLP to decode the vagueness in the human natural language [1-3].

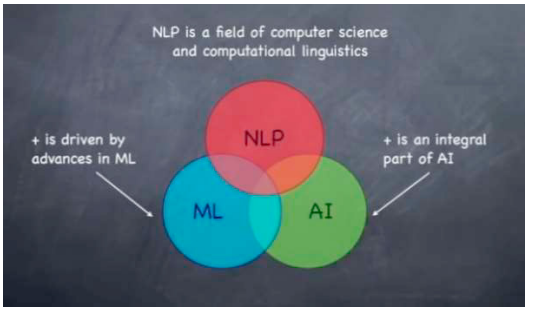

Fig. 1. Relationship of NLP with AI and ML.

Each day large volumes of data are produced all around the world, which is generally in the form of text, thus there is a need for smart system to process, study and translate this data into the suitable form. NLP can help us to implement certain tasks like Automated Speech and Text faster and more efficiently. NLP can also be used for the Automation. Nowadays people want everything to be automatic and can perform tasks through a voice command. This kind of automation is done by speech recognition for which NLP is required.

2. Components of NLP

There are two major components of NLP as depicted in Fig. 2.-

- Natural Language Understanding (NLU)
- Natural Language Generation (NLG)

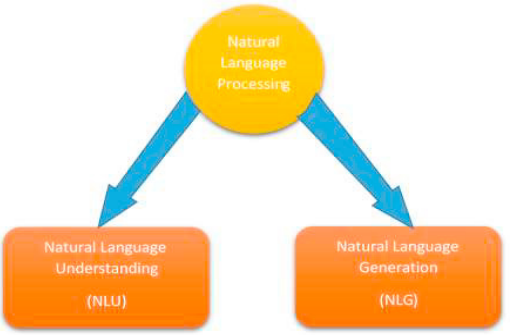

Fig. 2. Components of NLP

2.1. Natural Language Understanding

The most difficult part of NLP that a smart machine or system faces is the understanding of Natural Language. The very first step is to convert the natural language into machine understandable language i.e. binary language. It is the way in which Speech Recognition and Speech to Text systems work. It is the initial step in NLU. As soon as the data arrives in the text format, the NLU process starts with an objective of taking out the meaning from the text. Maximum Speech recognition systems works on Hidden Markov Models (HMMs). This model uses various techniques of mathematics and statistics to convert the speech into text.

HMMs works by listening the spoken data and then dividing them into small sections of generally 30 to 40 milliseconds. Then it matches these small segments of speech with the pre-recorded speech to identify the sound what was said in each segment of the speech. Then, a series of phonemes are targeted and through mathematical calculations it finds out the most similar words and sentences that was said. The next and the complex task in NLU is real understanding section [4, 5].

2.2. Natural Language generation

The artificial language obtained by the NLU step is then converted into text by the Natural Language Generator (NLG). The task of conversion of this text to audible speech is also done by NLG via text to speech. Primarily the Natural Language Processing system finds the materials that are to be converted into text. The analysis of text in

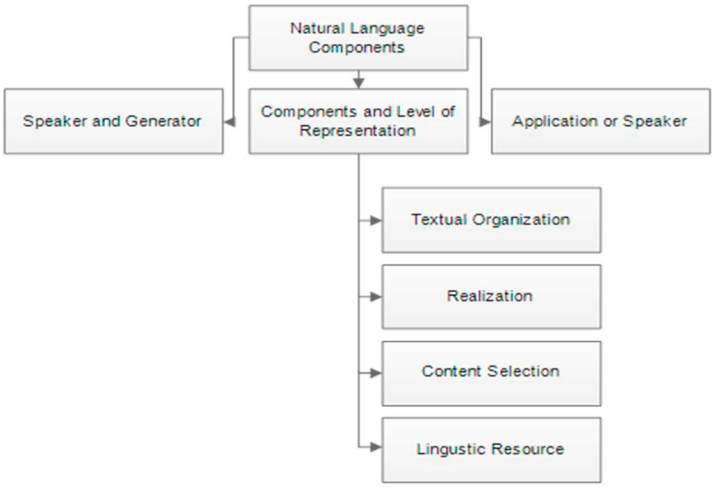

Fig. 3. Natural Language Generation components

text-to-speech system is done by using a prosody model, which detects breaks, duration, and pitch. Then, utilizing a speech data (pre-recorded voice), the system gathers all the recorded phonemes to form single coherent speech string. Various components of NLG are shown in Fig. 3.

2.2.1. Speaker and Generator

For the generation of text, a speaker and a generator program concentrate the applications objective into smooth phrase relevant to the state.

2.2.2. Components and level representation

The language generation process includes the following interconnected tasks:

- Content Selection: The gathered information is compiled into a set. It is then analyzed into representational units for the removal of some part of this unit and the addition of default parts.
- Textual organization: The data is then arranged textually according to the grammar.
- Linguistic Resources: Linguistic resources are chosen to support the information's realization. These resources comes down to specific words, idioms etc. at the end.
- Realization: The chosen and codified resources are realized into actual text.

2.2.3. Application or speaker

It is for supporting the model of the condition. In this speaker only starts the process and does not take any initiation in the language generation.

3. Case Studies

Several systems have been suggested this far, for addressing NLP applications, however the need for the generative model that can address diversified NLP problems is still being worked upon. Present work suggests a model based on element retrieval from case studies, extracted from articles related to various applications of NLP, from each case study certain significant blocks have been derived, deductions from each case involved a thorough review of research articles formulated as case study. Broadly case studies can grouped under following categories: Behavioral Information Derivation through Behavioral Signal Processing, Stroke and Multidimensional Representation Method for Chinese Language, Translation of natural language queries into structured data Formulas, Interactive NLP based Document Retrieval and A Four-Dimensional Vision for Natural Language Requirement Processing, from every case study significant modules were identified, these modules were then systematically connected according to the sequence of functions to be executed to formulate a generative model [6-8].

3.1. Case Study 1: Human Behavioral Information Derivation Through Behavioral Signal Processing

Behavioral Signal processing insinuates to practices and reckoning approaches that encourages the measurement, analysis and modelling of human behavioral signals. These behavioral signals are demonstrated mutually in overt and covert multimodal signals and are also administered and utilized by the humans unambiguously and indirectly. The principal aim of BSP is to notify human estimation and judgement. Therefore, the results of BSP are termed as behavioral informatics.

In this section, case study personifying a particular style of vocal communication and a specific goal of behavioral analysis has been discussed [8-16].

Reckoning language processing instruments are used by the educators in evaluating the constituents o budding learning abilities. These tools provide pledging methods in supporting educators in evaluation of these skills and are also promising in suggesting novel computerized teaching support. Even though a lot of effort in this field has aim at some particular significant constituents for instance, mispronunciation, speech rate and emotions, but studies show that these constituents alone cannot deliver a comprehensive illustration.

The most fascinating and thought-provoking aspects of this problem are various causes such as irregularity in language and socioeconomic upbringing of the learner along with the knowledge levels and circumstances of the educator are the reason for the inconsistency outside the perceptive of learning differences. Behavioral Signal Processing provides technique to compute impartial attributes from the observable functioning of the task. These attributes are then used by the BSP for generating an analytical simulation which can commendably summaries the way a set of teachers would assess the provided data. In the above instance proficient conclusions about the behavior of the children without any kind of keen interference of the teacher are made by using Behavioral Signal Processing that is derived from the Technology Based Assessment of Language and Literacy (TBALL) scheme.

In order to acquire environmentally convincing and reliable robust verbal data, a human-computer system was used to instigate age suitable reading assignment for the children. A unique characteristic of BSP is to simulate professional processing, various human assessors were engaged to rate children for the study. These ratings were based on their general reading skill using their auditory recordings as a source for the ratings.

Depending on the student's apparent status a computerized technique which can acclimate to the learning intrusion can be facilitated through Behavioral Signal Processing [17 - 26].

From the Case study it can concluded that Behavioral Signal Processing forms a necessary component for an NLP based smart virtual assistant to respond in a more natural manner, therefore it is an essential feature of NLP applications.

3.2. Case Study 2: Stroke and Multidimensional Representation Method for Chinese Language Processing

For the past few years, alphabetical languages are the only point of interest in most of the main stream studies that are based on natural language processing. Very few studies have been focused on the ideographic languages for example the Chinese Language. Moreover, the language processing algorithms that are used for Chinese language processing usually considers that the elementary components of Chinese language are the Chinese words or the characters. Thus, these algorithms overlook the data that is hidden inside the profound manner of these characters.

Chinese encryption has some really exceptional characteristics which makes it necessary to convert Chinese characters of words into a sequence of strokes. With the help of these strokes more underlaying fundamental characteristics can be established by training a neural network for learning stroke implementation. With the help of this process individual character is denoted through a matrix made up of stroke vectors. These matrices are used to produce character implementation by the application of CNN straight onto the stroke matrices.

3.2.1. Chinese Character Embedding using Stroke Based Method

In order to produce an advance strokes illustration, a framework developed by Erik Huang was implemented for that training of the stroke vectors. To achieve the obligatory conditions of acquiring stroke vectors, the framework was minutely altered. Every document was substituted by an analogous character components table. Character components having similar semantic meaning were paired up in this table. The model stated above utilized the strokes to calculate a score for every stroke in each character, designated as $Score_c$. After the calculation of $Score_c$ the character component of the corresponding pair substitutes the previous stroke character for the calculation another score, designated as Score. Fig. 4. depicts stroke learning system on similar components.

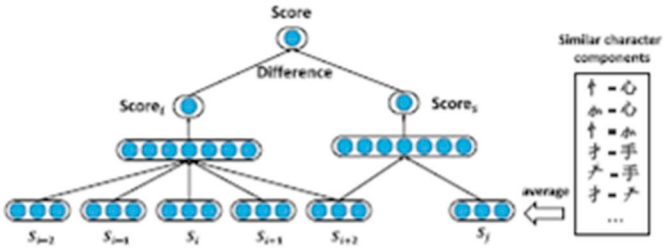

Fig. 4. Stroke embedding learning considering similar character

A smaller variation amongst the two scores, illustrates that the two components are much more similar to each other. The Stroke vectors are produced by using the back-propagation algorithm for the pretraining model.

3.2.2. Chinese Text Classification Service

After the conversion of Chinese characters into the stroke sequence and training these stroke vectors, an association between the Chinese characters and strokes is found in the form of a well-trained stroke vectors. For the application of Chinese character embedding for the classification of text on new data, this raw data is processed through the two above stated stages and then for the text classification a convolutional network made up of 6 layers and a neural network of 2 fully connected layer module is used. The complete representation of this complete process is illustrated below in Fig. 5.

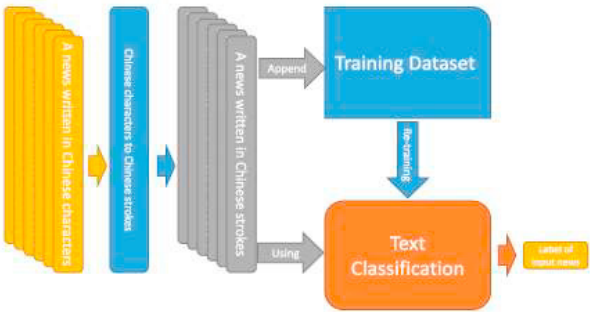

Fig. 5. Architecture of text classification process.

3.2.3. Automatic Chinese Text Summarization

During the text summarization process, it is obligatory to produce a readable text from the outputs obtained from the text classification task. The text summarization model uses RNN as the decoder for the conversion of the classified text into a readable text from the character matrix generated from the stroke vectors after text

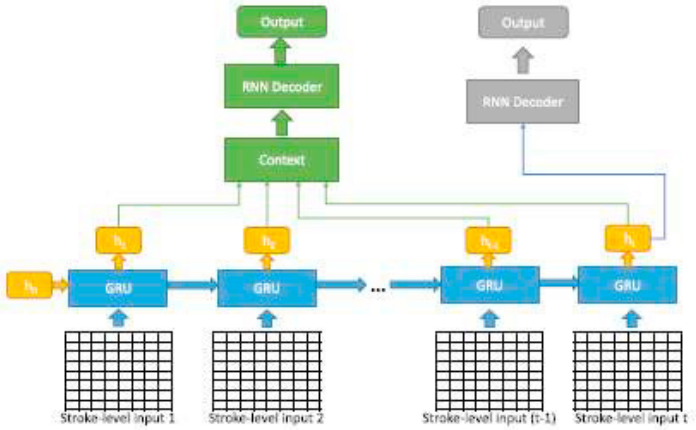

Fig. 6. Text Summarization Architecture.

classification. Fig. 6. illustrates the text summarization architecture [27-42].

The above case study suggests that this technique of stroke based Chinese language translation is of utmost importance for the current NLP based translation modules to improve their efficiency in translating ideographic languages along with the alphabetical languages, therefore stroke based module forms an essential component of NLP applications.

3.3. Case Study 3: Translation of Natural Language Queries into Structured Data Formulas

There is numerous Computer Software system around the globe with each one of them having their own diverse standards of interactions. As a result of which it leads to some the problem in their free use. To solve this problem natural language can be used for the communication with computer software systems. Use of natural language as an interface to communicate with the computer software may help people and has such advantages:

- Minimum user planning for operating a computer system.
- Effortless and quick method to put an indiscriminate question to software system.

Using regulated array of lexis and grammar for natural language user interface for software system does not steer to the profound functionality damage of the query resolving system.

3.3.1. Workflow

Natural language UI is regarded as a specialized intelligent system offering an exchange of ideas between the user and the software system inside a certain discipline. Intelligence system of natural language UI comprises the following:

- User Interface where the answer is provided by the system to the message given as the input by the user to the system.
- Natural language request to internal language of queries converter.
- Internal language components to natural language translator.

The procedure of natural language UI begins with a text input in the natural language format by typing message. Formal description creation of the text is the next task. For the analysis of following queries results of all the previous analysis are used which makes it feasible to settle the issues related with utilizing the equivalent expressions in dissimilar subject matters.

Each and every component of user interface knowledge machine of natural language can be classified into translator and analyzers. The working of translator is to translate one language to another.

3.3.2. Markov Decision Process

In this section, the discussion system is represented as a Markov Decision Process (MDP). The dialogue system consists of translation system as a part of it. MDP is explained by the expressions of a state space, an action set and strategy. All the resources by which the dialogue system interacts are represented by its state. For instance, the state contains two objects: libraries catalogue filtered by "Programming language" and libraries catalogue filtered by "license type". The overall sum of states consists of one initial state, various permutations of libraries catalogue based on the input values, the filters and a final state.

In a dialogue system the action set contains all the possible actions that are performed by it, such as communication through the user (e.g. requesting the user for an input, responding the user with some output, conformation, etc.), communicating with the external resources and internal processing.

3.3.3. Natural Language Processing

The NLP of the text includes three states:

- Morphological Analysis
- Syntactic Analysis
- Semantic Analysis

The initial phase is Morphological Analysis. For every all log relationship of linguistic classes like gender, case, declension etc. are correctly separated from the sentence.

The following phase of this process is Syntactic Analysis. The syntactic relationship among the words are generated. The main and secondary part of the sentence are separated, type of the sentence is defined and so on. The implementation of syntactic analysis is carried out phase by phase by utilizing the information attained from previous step. The lexical and syntactic rules are used for language analysis in the step.

The next phase is Semantic Analysis. It is the highly challenging phase of NLP. This analysis is centered on knowledge machine for a particular subject matter and data obtained from earlier steps. In this phase the language creation is matched with the creations saved in the memory of the system [43 - 47].

Suggested algorithm of translation of natural language queries into structured data is very helpful in retrieving the information from the database as the unstructured natural language data used by most of the NLP tools are cumbersome and have low efficiency. Structure data converter module forms essential functional block of NLP applications.

3.4. Case Study 4: Interactive NLP based Document Retrieval System

This application of NLP is used for the retrieval of documents and data from a database by processing the natural language queries. The main feature of this system is: processing of queries of user, working together with the user to rectify the inappropriate syntax queries, providing results of the given queries.

3.4.1. System Architecture

The key element of this system is query processing of natural language. This characteristic of the system permits to examine and handle natural language queries given by the user to decide what to be searched. The interactive

behavior of the system helps the user to use precise query in natural language that the program is capable of dealing with. The communication state between the user and the system is described in brief as follows:

- English query given by the user
- The structural syntax of the query is evaluated by the system.
- If the syntax of the queries is correct, then the system will start the query processing and the searched data is resulted to user.
- If the syntax of the query is incorrect the system will suggest related queries.

For the realization of the above functions, the model of the system must have three following components:

- Natural Language Query Processing: Its function is to solve the syntax and semantic representation of natural language query.
- Document Catalogue Database Retrieval: It transfers the queries in natural language to the set of database query for their implementation.
- Query Answering: The results are filtered, organized and provide the results according to the user's query.

Fig.7. below shows the System Functioning and Fig. 8. shows the System Architecture.

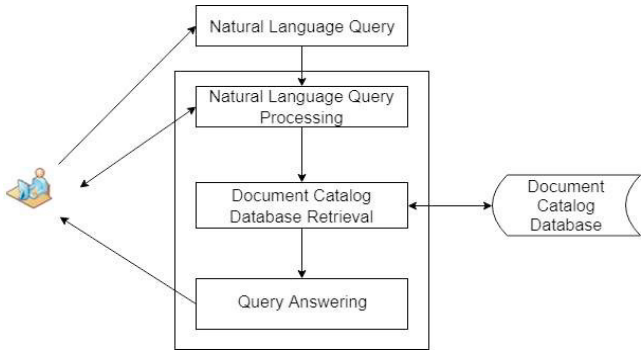

Fig. 7. System Functioning

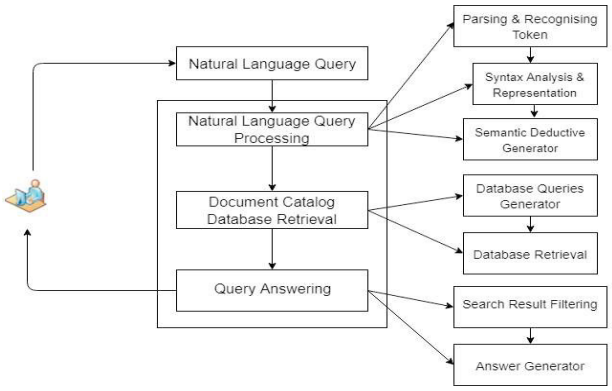

Fig. 8. System Architecture

3.4.2. Syntax Analysis

Some pre-defined rules for syntax of the queries are provided to the system. There are nearly hundred rules, these rules can comprise maximum number of rules consisting to information about e-books such as author, title, publisher…

The system evaluates and identifies the tokens. These identified tokens are then matched with pre-decided words in nouns, verbs, prepositions, conjunctions, and interrogatives lists. All rest unmatched tokens are rejected and overlooked.

3.4.3. Semantic Representation

The meaning of verb forms the base of the semantic structure of a query. The relationship of verbs with each verb involves the presence of an argument. The semantic structure comprises verb, noun and semantic relationship between the two.

The Semantic Detective Generator converts the syntax structure into a semantic structure. The essential norms are described as follows:

- Determination of verbs in syntax structure.
- Defined semantic structure that corresponds to concluded verbs are listed. The program only functions on the verbs in active and passive voices. At present the tenses of the verbs are not considered.
- Depending upon the amount and arrangement of nouns in the syntax structure, reject the semantic structures that have number of arguments that are unsuitable to the syntax structures.
- Depending on amount and arrangement of prepositions, conjunctions in syntax structure to reject the unsuitable semantic structures.
- Determining the most suitable semantic structure for a syntax structure.

3.4.4. Database Retrieval

The Database retrieval component is based on two parts:

- Database Queries Generator (GQR) Module.
- Database Retrieval (DBR) Module.

The Database Queries Generator Module generates a set of database queries from the semantic structure. Then these database queries are executed by Database Retrieval Module for the results.

3.4.5. Construction of Answer

There are two major parts of query answering component:

- Search Result Filtering (SRF) Module.
- Answer Generator (AG) Module.

To yield the result to the user by the Answer Generator module the search results from the database is filtered and organized.

Interactive Document Retrieval module is one of the most vital components of a NLP based Model as this system is responsible for the retrieval of all the necessary information required by the user and for the response generation. [48-61]

3.5. Case Study 5: A Four- Dimensional Vision for Natural Language Requirement Processing

Conditions or things that are obligatory or essential are termed as Requirements. In general, whenever there is a need to communicate any kind of requirements the most common mode of communication is Natural Language (NL). Since in most of the cases requirements are communicated in Natural Language, Natural Language Processing can be applied in Requirement Engineering processing for a wide variety of tasks. In this case study a Four-

Dimensional Framework is discussed that can provide a theoretical view of the relationship between the two fields. The four dimensions of this model are: *Discipline*, *Dynamism*, *Domain Knowledge* and *Dataset*.

3.5.1. Discipline

The theoretical conceptualization of the needs of a system that are intrinsically open to understanding are known as Requirements. The fundamental uncertainty of the natural language is used to define the requirements due to its intrinsic openness. Conversely, if the requirement process continues for a longer period of time, it become sufficiently unambiguous that it is inferred by all the participants involved in the process. For composing unambiguous requirements, the vital technique is discipline.

Various standards designed for software development, facilitates the subject of discipline writing in requirement development. But there is no language instruction to ease the arrangement for the requirements' explanation. Thus, the requirement editors exercise their personal ideas for the discipline writing which can cause ambiguity in the process. These ambiguities may lead to an undesirable modification on the product due to the contradictions among the customer's demand the developed product. The research community for requirement engineering proposes to employ NLP to provide the editors an intuition about the uncertainty of their requirements.

One of the significant parts of RE is Requirement readability. The ability of a text that makes it easier to understand is known as requirement readability. Requirement readability is linked to the complications of the terminologies and syntax of the text. There are NLP studies that are in progress on readability and text simplification.

3.5.2. Dynamism

Requirements are examined, debated, negotiated and improved, thus slowly developing an executable product during the development process of a product. For the product to implement all the extreme levels of requirements that are agreed upon by the customer, these developments trails must be recognized and regulated. Traceability is defined as the discipline that deals with the cross linking of requirements with another requirement, that may be present at different concept level or with other products of the software processes. In other words, traceability associations formulate the network controlling the characteristic dynamism of requirements software related products.

NLP has been utilized in the studies to support defining traceability associations and revise them when new requirements comes into practice. In addition, for the automatic tracing of requirements into various natural language regulations, NLP is being used to certify regulatory compliance.

Requirement categorization is also one of the important tasks related to dynamism. It helps in managing large number of requirements and makes the allotment of the requirements to specific software module. Categorizing requirements also helps them in using them in other projects as well.

3.5.3. Domain Knowledge

Requirements fits in various domains with technical or domain exclusive terminologies. There may be some cases that the developer or the requirement analyst is not familiar with the domain of the requirements and is unable to understand them, then he/she may go to the requirement editor for the help. It is very usual that the developer or the analyst asks the editor for the explanation about the requirements but sometimes the editor is unreachable by the developer or analyst and the right information is not attained.

Extracting domain specific terms and grouping them into familiar topics by the application of Natural Language Processing. Several advanced NLP methods allows the user to determine close relationships between similar terms. Requirement gathering also requires the need of domain knowledge. During this phase NLP can help by using various recommendation system to manipulate the stakeholders' domain and then leading this knowledge to suitable discussion forum. Various tactics influences the domain knowledge encrypted in Natural Language by encouraging the reutilization of internal NL requirements.

3.5.4. Dataset

There is an assumption which forms the basis of conventional NLP methodologies, that language is governed by consistencies that are catalogued as rules which are necessary to obtain significant data or knowledge from the texts. Certainly, this assumption was wrong because its machine learning methods that obtain statistical data from large number of documents and thus acquire the language rules without openly cataloguing them.

A dataset is an assortment of requirements with explanation granting semantic information that are reliant on task about them. For example, in a categorization task, each requirement is glossed with its category; for ambiguity detection, ambiguous parts within the text are marked by glosses. Unlike humans that usually performs interpretations an ML algorithm aims to guess the expected interpretations, either on the basis of a subset of the glossed data, as in the paradigm of supervised learning, or without relying on the existing annotations, as in the paradigm of unsupervised learning.

Natural Language requirements have been using Machine Learning for tasks such as requirements classification, identification of equivalent requirements, ambiguity detection, and traceability. However, a generalizable result was not obtained from most of these practices because each study focused on a limited set of requirements in a specific domain. As a matter of fact, many datasets that cover multiple domains are not publicly available, and researchers must work with a lack of resources. Generalization is a key issue because a technique might not work well in different domains, given the different terminologies and processes and the absence of a common discipline. [62 - 70]

Processing through four-dimensional vision for Natural Language forms an essential process to ensure successful implementation of NLP algorithms, therefore it forms a vital part in NLP algorithms.

4. Generative Predictive Model for NLP Applications using Case Study Approach

Analysis deducted from case studies suggests essential functional blocks, vital for NLP algorithms. The proposed work proposes a model of smart virtual assistant that can utilize the best features adopted from all the case studies analyzed for an improvement on the existing NLP models along with the ability to understand more complicated languages like the Chinese language. Fig. 9. depicts the derived model obtained from the synthesis of case studies. The first phase of proposed model will process various Natural Languages along with languages equipped with strokes, like Chinese, in a more precise manner by using the Stroke based natural language processing module.

Second phase converts the natural language data into structured data. Data is now more structured and understandable.

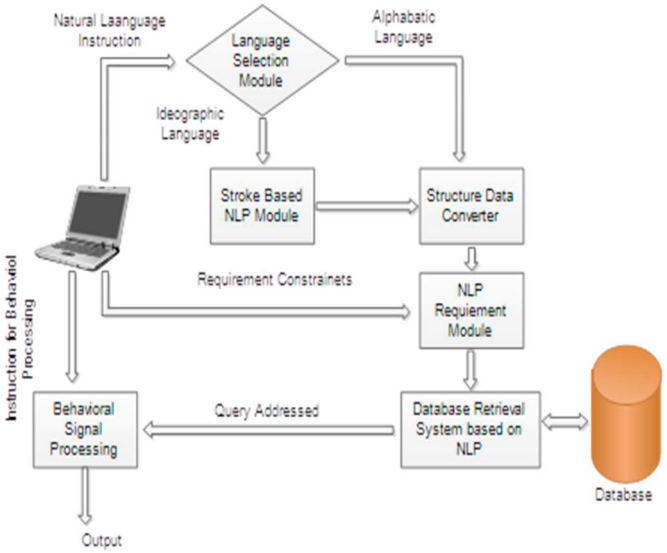

Fig. 9. Derived Model.

Next third phase involves Natural Language Requirement Processing. The structured data obtained from the previous phase will now go to a requirement processing phase which will take a decision whether the data contains all the requirement constraints provided by the user. If the data satisfies all the requirements constraints then it will be used by the text phase, which is an interactive NLP based document retrieval module for assisting information retrieval from the database.

Behavioural Signal processing forms the final module that facilitates solution generation for virtual assistant. Virtual assistant will generate results in correlation with the information obtained from human behaviour signal processing module. The BSP module will provide the ability to understand the human behaviour from the way of speaking, and based on the outputs of this, the module will modulate its voice and the nature, thus of in which queries of the user is answered.

5. Conclusion

Process of conversion of natural language into machine understandable language is known as Natural Language Processing. It looks like a simple task but in actual it is a far more complicated task to design a system that can process natural language into machine language. Since there are hundreds of languages all around the world developing a system for converting natural language into machine language requires a great amount of knowledge of grammatical rules of each language. The problem of ambiguity is also an issue that makes Natural Language Processing difficult, till date diversified solutions have been proposed addressing different issues related to language processing problems. Paper presents as generative model derived from different case studies, vital components for pre and post processing natural language have been identified and segregated to build the generative model. Model proposed provides single platform for producing optimized solutions and addressing different issues of language processing. As an extension to work the proposed model is further analyzed by developing a prototype and functional component can be modified &restructured according to need and trends.

References

[1] T. Patten and P. Jacobs. (1994) “Natural-language processing.” *IEEE Expert* **9** (**1**): 35.

[2] M. H. Amirhosseini, H. B. Kazemian, K. Ouazzane and C. Chandler. (2018) “Natural Language Processing approach to NLP Meta model automation” in *International Joint Conference on Neural Networks (IJCNN)*.

[3] Bo-June Hsu. (2008) “Generalized linear interpolation of language models.” *IEEE Workshop on Automatic Speech Recognition & Understanding (ASRU)*.

[4] Yang Li, X. Liu and L. Wang (2013) “Structured modeling based on generalized variable parameter HMMs and speaker adaptation”, in 8th International Symposium on Chinese Spoken Language Processing.

[5] W. Zhang, Bicheng Li, Dan Qu and B, Wang (2007) “Automatic Language Identification using Support Vector Machines”, in 8th international Conference on Signal Processing.

[6] Rongfeng Su, Xunying Liu and Lan Wang. (2014) “Automatic model complexity control for generalized variable parameter HMMs.” *IEEE Workshop on Automatic Speech Recognition and Understanding*.

[7] S. Raptis, S. Karabetsos, A. Chalamandaris and P. Tsiakoulis. (2015) “Towards expressive speech synthesis: Analysis and modeling of expressive speech.” *5th IEEE Conference on Cognitive Infocommunications (CogInfoCom)*.

[8] E. C. Williams, N. Gopalan, M. Rhee and S. Tellex. (2018) “Learning to Parse Natural Language to Grounded Reward Functions with Weak Supervision.” *IEEE International Conference on Robotics and Automation (ICRA)*.

[9] S. Narayanan and P. Georgiou (2013) “Behavioral Signal Processing: Deriving Human Behavioral Informatics”, in Proceedings of the IEEE **101** (**5**): 1203-1233.

[10] M. Black, J. Chang, and S. Narayanan (2008) “An empirical analysis of user uncertainty in problem-solving child-machine interactions”, in Proc. Workshop Child Comput. Interaction.

[11] S. Arunachalam, D. Gould, E. Andersen, D. Byrd, and S. Narayanan. (2001) “Politeness and frustration language in child-machine interactions.” *Politeness and frustration language in child-machine interactions*: 2675–2678.

[12] S. Yildirim, C. Lee, S. Lee, A. Potamianos, and S. Narayanan. (2005) “Detecting politeness and frustration state of a child in a conversational computer game.” *Proc. Eurospeech Conf., Lisbon, Portugal*: 2209–2212.

[13] T. Zhang, M. Hasegawa-Johnson, and S. Levinson. (2006) “Cognitive state classification in a spoken tutorial dialogue system.” *Management of Environmental Quality: Speech Commun* **48** *(**6**)*: 616–632.

[14] S. Yildirim, S. Narayanan, and A. Potamianos (2011) "Detecting emotional state of a child in a conversational computer game", in Comput. Speech Lang **25** (**1**): 29-44.

[15] K. Forbes-Riley and D. Litman (2011) "Benefits and challenges of real-time uncertainty detection and adaptation in a spoken dialogue computer tutor", in Speech Commun. **53** (**9**): 1115–1136.

[16] H. Pon-Barry and S. M. Shieber. (2011) "Recognizing uncertainty in speech." *EURASIP J. Adv. Signal Process*.

[17] P. Price, J. Tepperman, M. Iseli, T. Duong, M. Black, S. Wang, C. Boscardin, M. Heritage, David Pearson, S. Narayanan, and A. Alwan. (2009) "Assessment of emerging reading skills in young native speakers and language learners." *Speech Commun.* **51** (**10**): 968–984.

[18] J. Mostow, G. Aist, C. Huang, B. Junker, R. Kennedy, H. Lan, D. T. Latimer, R. O'Connor, R. Tassone, B. Tobin, and A. Wierman. (2008) "4-month evaluation of a learner-controlled reading tutor that listens." in *The Path of Speech Technologies in Computer Assisted Language Learning: From Research Toward Practice, F. N. F. V. M. Holland, Ed. New York: Routledge*: 201–219.

[19] D. Litman and K. Forbes-Riley (2004) "Predicting student emotions in computer-human tutoring dialogues"', in in Proc. 42nd Annu. Meeting Assoc. Comput. Linguist: 351-358.

[20] J. Tepperman, S. Lee, S. S. Narayanan, and A. Alwan (2011) "A generative student model for scoring word reading skills", in IEEE Trans. Audio Speech Lang. Process. **19** (**2**)

[21] A. Kazemzadeh, H. You, M. Iseli, B. Jones, X. Cui, M. Heritage, P. Price, E. Anderson, S. Narayanan, and A. Alwan. (2005) "Tball data collection: The making of a young children's speech corpus." *Proc. 9th Eur. Conf. Speech Commun. Technol.*: 1581–1584.

[22] M. Black, J. Tepperman, and S. Narayanan. (2011) "Automatic prediction of children's reading ability for high-level literacy assessment." *IEEE Trans. Audio Speech Lang. Process.* **19** (**4**): 348–360.

[23] M. Black and S. Narayanan. (2012) "Improvements in predicting children's overall reading ability by modeling variability in evaluators' subjective judgments." *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*: 5069–5072.

[24 S. D'Mello, A. Graesser, and R. Picard (2007) "Toward an affect-sensitive autotutor"', in IEEE Intell. Syst. **22** (**4**): 53-61.

[25] A. Graesser and S. D'Mello (2011) "Theoretical perspectives on affect and deep learning", in New Perspectives Affect Learn. Technol. **3**: 11-21

[26] M. Eskenazi. (2009) "An overview of spoken language technology for education." *Speech Commun.* **51** (**10**): 832–844.

[27] H. Zhuang, C. Wang, C. Li, Yijing Li, Q. Wang, X. Zhou. (2018) "Chinese Language Processing Based on Stroke Representation and Multidimensional Representation." *IEEE Access* **6**: 41928–41941.

[28] H. Zhuang, C. Wang, C. Li, Q. Wang, and X. Zhou. (2017) "Natural language processing service based on stroke-level convolutional networks for Chinese text classification." in *Proc. IEEE Int. Conf. Web Services (ICWS)*: 404–411.

[29] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa (2011) "Natural language processing (almost) from scratch"', in J. Mach. Learn. Res. **12**: 2493-2537

[30] X. Chen, L. Xu, Z. Liu, M. Sun, and H.-B. Luan (2015) "Joint learning of character and word embeddings", in Proc. IJCAI.: 1236-1242.

[31] M. Zhang, Y. Zhang, W. Che, and T. Liu. (2013) "Chinese parsing exploiting characters." in *Proc. 51st Annu. Meeting Assoc. Comput. Linguistics* **1**: 125–134.

[32] M. Kang, T. Ng, and L. Nguyen. (2011) "Mandarin word-character hybrid-input neural network language model." *Proc. 12th Annu. Conf. Int. Speech Commun. Assoc.*: 625–628.

[33] X. Zhang, J. Zhao, and Y. LeCun. (2015) "Character-level convolutional networks for text classification." *Proc. Adv. Neural Inf. Process. Syst.*: 649–657.

[34] X. Shi, J. Zhai, X. Yang, Z. Xie, and C. Liu (2015) "Radical embedding: Delving deeper to Chinese radicals"', in Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics 7th Int. Joint Conf. Natural Lang. Process. **2**: 594-598.

[35] Y. Sun, L. Lin, N. Yang, Z. Ji, and X. Wang (214) "Radical-enhanced Chinese character embedding", in Proc. Int. Conf. Neural Inf. Process. Beijing, China: Association for Computational Linguistics: 279-286.

[36] H. Zhuang, C. Li, and X. Zhou (2018) "CCRS: Web service for Chinese character recognition." *in Proc. IEEE Int. Conf. Web Services (ICWS)*, San Francisco, CA, USA.

[37] Y. Kim, Y. Jernite, D. Sontag, and A. M. Rush (2016) "Character-aware neural language models." in *Proc. AAAI*.

[38] S. Cao et al. (2018) "Cw2vec: Learning Chinese word embeddings with stroke n-gram information." *Proc. AAAI Conf. Artif. Intell*.

[39] J. Yu, X. Jian, H. Xin, and Y. Song (2017) "Joint embeddings of Chinese words, characters, and fine-grained subcharacter components"', in Proc. Conf. Empirical Methods Natural Lang. Process.:286-291.

[40] R. Collobert and J. Weston (2008) "A unified architecture for natural language processing: Deep neural networks with multitask learning", in Proc. 25th Int. Conf. Mach. Learn.: 160-167.

[41] E. Hovy and C.-Y. Lin. (1998) "Automated text summarization and the summarist system." in Proc. Workshop Held. Baltimore, MD, USA: Association for Computational Linguistics: 197–214.

[42] D. Das and A. F. Martins. (2007) "A survey on automatic text summarization." *Literature Surv. Lang. Statist. II CourseCMU* **4**: 192–195.

[43] R. Posevkin and I. Bessmertny. (2015) "Translation of natural language queries to structured data sources." in 9th International Conference on Application of Information and Communication Technologies (AICT.

[44] I. Bessmertnyi (2014) "On constructing intellectual systems in ternary logic"', *Programming and Computer Software* **40**(**1**): 43-46.

[45] Levin E., Pieraccini R., Eckert W. (1997) "Learning dialogue strategies within the Markov decision process framework", in Automatic Speech Recognition and Understanding IEEE Proceedings: 72-79

[46] O. Eliseeva. (2009) "Natural language interface of intelligence systems." *Minsk: BGUIR*.

[47] Damljanovic, Danica, Milan Agatonovic, and Hamish Cunningham (2012) "FREyA: An interactive way of querying Linked Data using natural language." in The Semantic Web: ESWC 2011 Workshops. Springer Berlin Heidelberg.

[48] D. T. Nguyen. (2009) "Interactive document retrieval system based-on natural language query processing." in *International Conference on Machine Learning and Cybernetics*.

[49] Enrique Alfonseca, Marco De Boni, José-Luis Jara-Valencia, Suresh Manandhar (2002) "A prototype Question Answering system using syntactic and semantic information for answer retrieval"', in Proceedings of the 10th Text Retrieval Conference.

[50] Eric Brill, Susan Dumais, Michele Banko (2002) "An Analysis of the AskMSR Question-Answering System", in Proceedings of the Conference on Empirical Methods in Natural Language Processing.

[51] Boris Katz, Jimmy Lin. (2003) "Selectively Using Relations to Improve Precision in Question Answering." in Proceedings of the EACL 2003 Workshop on Natural Language.

[52] Boris Katz, Beth Levin. "Exploiting Lexical Regularities in Designing Natural Language Systems." *Proceedings of the 12th International Conference on Computational Linguistics.*

[53] Callison-Bruch, Chris. (2000) "A computer model of a grammar for English questions." *Undergraduate honours thesis, Stanford University*.

[54] Nguyen Kim Anh (2006) "Translating the logical queries into SQL queries in natural language query systems"', in Proceedings of the ICT.rda'06 in Hanoi Capital, Vietnam.

[55] Nguyen Tuan Dang, Do Thi Thanh Tuyen (2008) "E-Library Searching by Natural Language Question-Answering System", in Proceedings of the Fifth International Conference on Information Technology in Education and Training: 71-76.

[56] Nguyen Tuan Dang, Do Thi Thanh Tuyen. (2009) "e-Document Retrieval by Question Answering System." *International Conference on Communication Technology, Penang, Malaysia*

[57] Nguyen Tuan Dang, Do Thi Thanh Tuyen. (2009) "Natural Language Question Answering Model Applied to Document Retrieval System." *International Conference on Computer Science and Technology, Hongkong, China*.

[58] Nguyen Tuan Dang, Do Thi Thanh Tuyen. (2009) "Document Retrieval Based on Question Answering System." *The Second International Conference on Information and Computing Science,* Manchester, UK.

[59] Nguyen Tuan Dang, Do Thi Thanh Tuyen, Phan Tan Quoc (2009) "A Document Retrieval Model Based-on Natural Language Queries Processing"', in *The International Conference on Artificial Intelligence and Pattern Recognition*, Orlando, FL, USA.

[60] Riloff, Mann, Phillips (2006) "Reverse-Engineering Question/Answer Collections from Ordinary Text", in Advances in Open Domain Question Answering, Springer Series: Text, Speech and Language Technology **32**.

[61] Fabio Rinaldi, James Dowdall, Kaarel Kaljurand, Michael Hess. (2013) "Exploiting Paraphrases in a Question Answering System." *Proceedings of the second international workshop on Paraphrasing*.

[62] A. Ferrari, F. Dell'Orletta, A. Esuli, V. Gervasi, and S. Gnesi. (2017) "Natural Language Requirements Processing A 4D Vision." *IEEE Computer Society* **34** (**6**): 28–35.

[63] A. Casamayor, D. Godoy, and M. Campo. (2012) "Mining Textual Requirements to Assist Architectural Software Design: A State-of-the-Art Review." *Artificial Intelligence Rev.***38** (**3**): 173–191.

[64] B. Gleich, O. Creighton, and L. Kof (2010) "Ambiguity Detection: Towards a Tool Explaining Ambiguity Sources"', in Requirements Engineering: Foundation for Software Quality, LNCS 6182, Springer: 218-232.

[65] H. Yang et al. (2011) "Analysing Anaphoric Ambiguity in Natural Language Requirements", *Requirements Eng.***16 (3)**:163-189

[66] A. Ferrari and S. Gnesi. (2011) "Using Collective Intelligence to Detect Pragmatic Ambiguities." *Proc. 20th IEEE Int'l Requirements Eng. Conf. (RE 12)*: 59–80.

[67] Filippini, Massimo, and Lester C. Hunt. (2012) "US residential energy demand and energy efficiency: A stochastic demand frontier approach." *Energy Economics* **34** (**5**): 191–200.

[68] K. Collins-Thompson. (2014) "Computational Assessment of Text Readability: A Survey of Current and Future Research." *Int'l J. Applied Linguistics* **16** (**2**): 97–135.

[69] H. Sultanov and J.H. Hayes (2013) "Application of Reinforcement Learning to Requirements Engineering: Requirements Tracing"', Proc. 21st IEEE Int'l Requirements Eng. Conf. (RE 13): 52-61

[70] V. Gervasi and D. Zowghi (2014) "Supporting Traceability through Affinity Mining", Proc. 22nd IEEE Int'l Requirements Eng. Conf.:143-152