An Experimental Technique on Text Normalization and its Role in Speech Synthesis

Pooja Manisha Rahate, Manoj Chandak

Abstract*: *Text Normalization is an important step in Natural Language Processing. It plays an important role in various applications such as text-to-speech systems, speech recognition, email voice activators, for visually impaired persons and also for educational purpose. Text normalization means converting the non-standard words into their standard formats (i.e. in the string format or plain text format). The normalization of text is required for words such as dates, currency symbols, abbreviations, acronyms, and numbers etc. to be pronounced or read by the system. The paper presents an overview of work done so far in the field of text normalization and speech synthesis. The work on text normalization is being done from last few decades. The paper discusses in detail the numerous techniques used by different authors for achieving the accuracy in the text normalization for several different languages all over the world.

Index Terms*: *text normalization, speech synthesizer, text-to-speech systems, SMS normalization.

I. INTRODUCTION

Natural Language Processing (NLP) is the domain which focuses on the human language. The main goal of this domain is to make computers understand the human language their words with proper semantics and perform certain tasks. Various real-world applications of NLP are Text-to-Speech (TTS) Synthesizers, Automatic Speech Recognition (ASR), and Semantic Analysis etc. The vital part of NLP is text normalization. Text normalization is the technique which converts any informal text into system or machine readable format. In this paper the word informal text will be addressed in many ways such as non-standard words, arbitrary text, un-normalized text, non-English words etc. The informal text that are used by the humans in daily life are currency amounts, numbers, abbreviations, acronyms, dates, time, email addresses etc. Humans can easily identify these informal texts as their brain is already being trained and is aware of these patterns in the text. To identify the human language with its proper semantic is known as Natural Language Understanding. For which several models or methods are used to train the systems.

Speech synthesis is the artificial process of producing the human voice or speech. The system that produces the artificial human speech is known as synthesizer. Today, due to digitalization and improvement in technology the use of speech synthesizer has increased. Synthesizers are used in companies where instead of a human a system is use to answer the customer calls and resolve their queries. The most common application of the synthesizers today is the smartphones. These devices due to synthesizers in it are capable of answering any question asked by the human. The synthesizers today can read out news for the humans, messages, emails and are also used for the remainders instead of the alarm sounds. Synthesizers are also important for non-conversational applications that speak to people who are visually impaired, for them the synthesizers are used to guide them while walking or finding the correct route, or reading any information asked by the human by searching on the internet.

Speech synthesis system performs 2 main steps: first, to convert the input text in the phonemic representation and second convert this phonemic representation in the wave form. Therefore it is also known as waveform synthesis. It is the simulation of the human speech generated by the machine. [16]

II. SPEECH SYNTHESHIS

As mentioned above, a speech synthesis system has two phases: Text Analysis and Waveform Synthesis.

A. Text Analysis:

Text analysis is the first phase of speech synthesis where the text is preprocessed to obtain the phonemic representation of the text. This phase is again divided into 3 parts: text normalization, phonetic analysis and prosodic analysis.

a) Text Normalization:

In this step the document structure is analyzed. This process works on 2 levels; sentence – level (sentence tokenization) and word – level (word tokenization). In sentence tokenization to identify the boundary of the sentence is not always an easy task. i.e. it is not necessary that the sentence boundary is always defined by the periods, it might be sometimes colon or double inverted commas etc. In word tokenization, conversion of non-standard word into their standard form is done. [16]

b) Phonetic Analysis:

In this step the pronunciation of words which are given as input from the text normalization is determined. The grapheme – to – phoneme conversion and homograph disambiguation's are handled.

c) Prosodic Analysis:

In prosodic analysis step, the rhythm, sound,

Revised Manuscript Received on December 22, 2018.

Pooja Manisha Rahate, Department of Computer Science & Engineering, Shri Ramdeobaba College of Engineering & Management, Nagpur, India.

Manoj Chandak, HOD of Department of Computer Science & Engineering, Shri Ramdeobaba College of Engineering & Management, Nagpur, India.

intonation, tune and structure of the utterance of each phoneme is examined and applied to each phoneme to generate the phonological prosody. [16]

B. Waveform synthesis:

Waveform synthesis has three techniques: articulatory synthesis, formant synthesis and concatenative synthesis.

a) Articulatory synthesis:

Articulatory synthesis uses mechanical and acoustic model for speech generation. Even though it produces an intelligible synthesized speech but it is far away from the natural human sound. [9][16]

b) Formant synthesis:

Formant synthesis synthesizes the speech output which is created by using additive synthesis and acoustic model. Parameters that are considered for formant synthesis are fundamental frequency, voicing, noise levels that are used for creating the artificial speech. [9][16]

c) Concatenative synthesis:

Both articulatory and formant synthesis creates the artificial speeches where the concatenative synthesis generates the natural human speech. This is because the concatenative synthesis used the real-time human speech recorded samples from the database which are stored in the unit or syllable format. This unit samples are then concatenated to generate the prosody of the word that has encountered in the system. Concatenative synthesis has several units that are phone, diphone and triphone. Most of the 1960s to 1980s models use articulatory and format synthesizers for creating the human speech. [16]

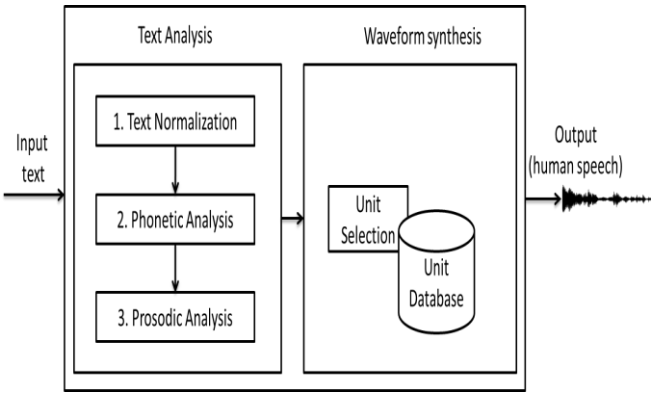

Fig. 1 Architecture for conversion of text into a speech form.

In the above architecture for waveform synthesis the concatenative synthesis is considered because the unit selection and unit database is used to generate the natural speech.

III. LITERATURE REVIEW

Text normalization has been the challenging task in speech synthesis as different languages contains different format or structure of writing text. Thus it is a difficult task to recognize the correct format for normalizing the informal text into formal text. This review may provide some insights to the new researchers and new scholars that will help them to understand different techniques and models that are used up till now for the text normalization research.

Emma Flin et al. created a system with four phases (i.e. detection, classification, division and expansion) by adopting and modifying the Sproat et.al (2001) NSW taxonomy. The system has been designed using Python Programming Language. The system is domain-modifiable by inputting the dictionary of abbreviation as per the user need. For ALPHA and NUMB (numeric) tokens, the author has used the semi-supervised label propagation algorithm, while MISC (miscellaneous) tokens are identified using rule-based. The algorithm uses 13 domain-independent features for ALPHA and 26 domain-independent features for NUMB classifier. These features look at the property of the each token and also +/- 2 surrounding tokens. The system gives much accurate performance when compared to (Sproat et.al.) voice TTS system. The accuracy achieved by the proposed system is more than 91%. [1]

Meenakshi Sharma has used the hybrid approach to translate the short messages into plain English text. The author has maintained a parallel corpus for the SMS abbreviation expansion. The system is the combination of Ruled-based Approach, Statistical Machine Translation and Direct Mapping Approach. Authors' system gives the evaluation of 89.5% accuracy when tested on various input text messages. The performance of system is tested on 3 main parameters that are accuracy, precision and F-score. The author used bi-gram to six-gram model for the propose system. [2]

Hay Mar Htun, Theingi Zin, Hla Myo Tun, used concatenation synthesis for converting the "phoneme – to – grapheme". The system is implemented using the MATLAB Programming Language. The system is divided into 2 parts: i) NLP and ii) Speech Synthesis. In NLP part, the text is first analyzed using the bi-gram model for correct POS tagger using the previous and next tokens. Then phonetic conversion is performed using dictionary based approach. The Prosody phrasing is assigned to the input text. In Speech synthesis, the author uses concatenative synthesis, in which the unit selection, phoneme based speech synthesis and domain-specific speech synthesis are used. The system works well for numbers, words and sentences using domain-specific approach, the output contains discontinues between the phoneme transitions. By using the unit selection approach, the little glitch is recognized in the output. [3]

Anand Arokia Raj et al, tried to build the TTS System for Indian Languages. The paper mostly concentrates on identifying the font-type, font-to-akshara conversion, and the pronunciation rules for Akshara and text normalization. As the Indian Languages text was stored in

ASCII format they are quiet difficult to process. The author created a corpus from various Indian Language websites for each font type. Thus the identification of font types is done using TF-IDF. Font-to-Akshara is done by building Base-Map Table and Assimilation Rules for font types. Pronunciation rules for Akshara are formed using the machine learning techniques where the contextual features and Acoustic - Phonetic and Syllabic features are used. For Normalization author used a Baseline system using a word-level decision tree. The author then compared the word level and syllable level features from which the syllable level performs significantly better than the word-level features. [4]

Richard Sproat, Navdeep Jaitly, used RNN, LSTM and RNN + FST models in their paper for text normalization. The dataset used is constructed using Wikipedia region which contains English and Russian text. The author here described 16 semiotic classes (i.e. NSW types) which should be normalized into either plain English text or plain Russian text. These 16 semiotic classes are Plain, Punctuation, Money, Measure, Time, Digit, etc. The author has described 3 experiments; first, using LSTM the system gives 82.9 % accuracy for English and 83.8% for Russian. Second, the author used RNN for which the system gives 95.0% for English and 93.5% for Russian for non-overlapped tokens. Third, the author used FST with RNN to overcome the wrong expansion for correct identifying the class. Even though the FST filter was proved valuable for correct identifying but it expands the NSW class word incorrectly, which also decreases the results for already correct and accurate normalized class performance. [5]

Shaurya Rohatgi, Maryam Zare, used the dataset provided in [5] and applied pure deep learning for text normalization process. The author did not use plain, punctuation and <eos> classes for the training. The author used Context Aware Classification model (CAC) for detecting the semiotic class and sequence to-sequence model with 2-layer LSTM for the expansion of NSW word. The author used gradient descent (Vanilla Gradient descent algorithm) without any parameters tuning as optimizer. The system achieved higher accuracy using these models, but as the input gets complex the expansion of NSW gets worse. As a baseline, the author has used regular expression using predicted class to normalize the test data which gives 98.52 % of the score on test data. [6]

Chen Li, Yang Liu, used unlabeled Twitter data for SMS normalization as the corpus for normalization. The paper describes unsupervised model which creates a lookup table of the non - standard words. This model creates the low -dimensional word embedding and their semantic similarity for identifying the non - standard words. The model identifies all possible OOV words for each plain English dictionary word. The author used discriminative re-ranking for both word-level and sentence level, where the author's system gives accuracy of an average of 75% for word-level normalization and the sentence level accuracy is 86.91%. Thus the author has used different models for re-ranking to achieve the accuracy. [7]

Dileep Kini, Sumit Gulwani, worked on the domain-specific programming language (DSL), where the user need to give the input of the informal text and the program learns the structure of the input and gives the expected output. The DSL consists of decision list, concat expression, and process expression and parse expressions as the different phases of the system. The system is known as FlashNormalize. The parse expression phase in the system uses the 5-tuple grammar which is non – recursive. The author has considered only date, telephone and numbers non – standard words for normalization purpose. The author used generic grammar for the parse expression phase so that no changes will be required in the algorithm even if the structure of NSW changes. The author worked on different languages for the normalization of non – standard words that are considered in the paper. The languages used by the author are English, Spanish, French, Polish, Russian, Chinese, German, Portuguese and Italian. [8]

Suhas R. Mache et al, gives the overview of development done in the area TTS synthesizer and gives the basic idea of the different TTS systems that are currently available. The author then gives the brief description of the TTS systems that are being implemented on different regional Indian languages such as Marathi, Hindi, Tamil, Kannada, Bengali, Punjabi etc. The synthesizer strategy, segmentation process, database, prosody and performance of the Indian TTS systems with their research institute name are also described by the author in the tabular format. [9]

V. López – Ludeña et al, proposed architecture in this paper which is based on the phrase – based translation system. The architecture has 3 main modules: First, a tokenizer module for splitting the text input into a token graph which follows the rules defined by the [15]. Second, the phrase – based translation module also known as token translation uses parallel corpus for training purpose. The model used in this module is the N-gram model. Third, the post – processing module removes not converted tokens. The architecture works only on the number and abbreviations which gives significantly better results. [10]

Conghui Zhu et al, used the email data for the normalization purpose. The author has first formalized the problems that occurs in text normalization and then defined this problem at 3 different levels: paragraph level, sentence level and word level. The author used 2 steps for the unified tagging approach: pre-processing and tagging. In pre-processing, a) separates the text into paragraphs, b) determine tokens in the paragraphs and c) assign possible tags to each token. For tagging CRF model is built during the training phase with labeled data using iterative algorithm which is based on Maximum Likelihood Estimation. The author has compared his method with the baseline methods (independent and cascade) that are descried in the paper. After comparison the author's method significantly outperforms for text normalization. [11]

Chris Lin, Qian (Sarah) Mu, Yi Shao, built a 3-component TTS system with the help of Naïve Bayes classifier for token – to – token , SVM classifier for token – to – class and hand built grammar rules based on regular expressions. The author has used balance as well as unbalanced SVM with parameter tuning. The NB classifier gives 99.81% accuracy with in training and 98.97% accuracy in development. The unbalanced SVM performs well rather than balanced SVM which is shown by the author with the accuracy of

98.88%. The author also shows the confusion matric which describes that using these models also there are some tokens that are unable to classify. Thus the overall accuracy achieved by the author's system is 99.36% in development and 98.88% for testing. [12]

Slobodan Beliga, Miran Pobar and Sanda Martincic-Ipšic, described the use of classification tree for text normalization for Croatian Text. The classification tree contains the NSW as the root node which contains 3 child nodes that are number, characters and combine. The normalization algorithm implemented by the author is in Perl language. The author used the Croatian data as the corpus which contains 11K words. The algorithms for the detection and normalization were constructed by the author using the combination of programmed rules and lookup dictionary. The text of different genres has the token rate of 95% from which the authors' system correctly flecked the 80% of the tokens. [13]

Gokul P. et al, worked on TTS system for Malayalam which is one of the regional Indian language. The pre-processing in the system is done for selecting non – uniform units from the database using the memory based hierarchical model. The memory based model used by the author is Memory Prediction Framework which organises the language sentences in the hierarchical tree structure with the sentences as the root node and the syllables as the lower nodes. The authors' system makes use of (C)(C)(C)V(C)C schematized syllable structure where C are the consonants and V stands for vowels to create the synthesized speech using the concatenative synthesizer. The author used the Ziff's law for producing the hyperbolic function of the natural language utterance from the corpus. The author used the N-gram model for collocation for predicting the correct semantic for the target word. Thus the author fine-tuned the system by increasing the size of the database. [14]

Richard Sproat et al, presented a taxonomy for the non-standard words which is used in hand – tagging and in normalization model. The corpora for the system is created from four different sources that are NANTC, Classifieds, pc110 and REF which are all domain – specific corpora. The corpus is then converted into the XML based format and the NSW is tagged with the help of inter-labeller method. The author used decision tree for tag prediction. WFSTs are used in the paper for splitting and expansion of the module. Finally, the author uses N-gram model for improving the disambiguation of the non – standard words. All the methods discussed above are used for treating the NSWs. Author separately used the unsupervised learning for the expansion of the sequence of letters, acronyms and abbreviations. Thus the paper provides the performance measure for all the different domains separately. [15]

IV. CONCLUSION

Text normalization is the important part of any speech and language processing application. Many papers had been published from the last few decades for the improvement of text normalization. Many new techniques and models are also developed for different languages to be processed for the same. But still their remains some flaws in every system for fully classify each NSW correctly. For speech synthesis the most commonly used synthesis for converting the text to speech is the concatenative synthesis due to its natural and intelligible speech. After so many efforts also there remains a lot of scope for the improvement in the speech and language processing domain especially for the text normalization phase.

REFERENCES

1. Emma Flin et al, "A Text Normalisation System for Non-Standard EnglishWords",Proceedings of the 3rd Workshop on Noisy User-generated Text, pages 107–115, Copenhagen, Denmark, September 7, 2017.
2. Meenakshi Sharma,"Text Normalization Using Hybrid Approach", International Journal of Computer Science and Mobile Computing, Vol.4 Issue.1, January- 2015, pg. 544-554.
3. Hay Mar Htun, Theingi Zin, Hla Myo Tun, "Text To Speech Conversion Using Different Speech Synthesis", International Journal Of Scientific & Technology Research Volume 4, Issue 07, July 2015.
4. Anand Arokia Raj et al, "Text Processing for Text-to-Speech Systems in Indian Languages", 6th ISCA Workshop on Speech Synthesis, Bonn, Germany, August 22-24, 2007.
5. Richard Sproat, Navdeep Jaitly, "RNN Approaches to Text Normalization: A Challange", arXiv preprint arXiv:1611.00068, Jan 2017.
6. Shaurya Rohatgi, Maryam Zare, "DeepNorm - A Deep Learning approach to Text Normalization", arXiv preprint arXiv:1712.06994, Dec 2017.
7. Chen Li, Yang Liu, "Improving Text Normalization via Unsupervised Model and Discriminative Reranking", Proceedings of the ACL 2014 Student Research Workshop, pages 86–93, 2014.
8. Dileep Kini, Sumit Gulwani, "FlashNormalize: Programming by Examples for Text Normalization", Proceeding IJCAI'15 Proceedings of the 24th International Conference on Artificial Intelligence, Pages 776-783, 2015.
9. Suhas R. Mache et al, "Review on Text-To-Speech Synthesizer", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 4, Issue 8, pg. 54-59, August 2015.
10. Lopez Ludeña, V., San Segundo, R., Montero, J. M., Barra Chicote, R., & Lorenzo, J. (2012). "Architecture for text normalization using statistical machine translation techniques." In IberSPEECH 2012 (pp. 112 – 122). Madrid, Spain, 2012.
11. Conghui Zhu et al, "A Unified Tagging Approach to Text Normalization", Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pg. 688–695, Prague, Czech Republic, 2007.
12. Chris Lin, Qian (Sarah) Mu, Yi Shao, "iTalk A 3-Component System for Text-to-Speech Synthesis", unpublished.
13. Slobodan Beliga, Miran Pobar and Sanda Martincic-Ipšic, "Normalization of Non-Standard Words in Croatian Texts", arXiv preprint arXiv:1503.08167v2, 30 Mar 2015.
14. Gokul P., Neethu Thomas, Crisil Thomas and Dr. Deepa P. Gopinath, "Text Normalization and Unit Selection for a Memory Based Non Uniform Unit Selection TTS in Malayalam", Proceedings of the 12th International Conference on Natural Language Processing, pg. 172-177, Trivandrum, India, 2015.
15. Richard Sproat et al, "Normalization of non-standard words", Computer Speech and Language (2001) 15, pg. 287–333.
16. Daniel Jurafsky, James Martin, Speech and Language Processing, Pearson India, 2nd Edition.