MY PERSONAL ASSISTANT - AUTOMATING MEETING AND CONFERENCE NOTE TAKING BY THE IMPLEMENTATION OF NATURAL LANGUAGE PROCESSING, WEB SPEECH API AND TWILLIO

Project report submitted in partial fulfilment of the requirement for the degree of Bachelor of Technology

in

Computer Science and Engineering

By

Akshat Mathur (141207)

Under the supervision of

Shubhi Gupta

Hashedin Technologies Pvt Limited

to

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology Waknaghat, Solan-173234, Himachal Pradesh

CERTIFICATE

Candidate's Declaration

I hereby declare that the work presented in this report entitled **"My Personal Assistant - Automating Meeting and Conference Note taking by the implementation of Natural language Processing, Web Speech API and Twillio"** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering/Information Technology** submitted in the department of Computer Science & Engineering and Information Technology**,** Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out over a period from August 2017 to December 2017 under the supervision of **Shubhi Gupta**(Technical Associate).

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Akshat Mathur, 141207

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

Shubhi Gupta
Technical Associate,
Hashedin Technologies Pvt Ltd
Dated:

ACKNOWLEDGEMENT

We owe my profound gratitude to my project supervisor **Shubhi Gupta**(Technical Associate), who took keen interest and guided me all along in the project work titled – "**My Personal Assistant - Automating Meeting and Conference Note taking by the implementation of Natural language Processing, Web Speech API and Twillio**". I take this opportunity to express deep regards to my guide for her exemplary guidance, monitoring and constant encouragement throughout the course of this project. The in-time facilities provided by Hashedin Technologies throughout the project development are also equally acknowledgeable. At the end I would like to express my sincere thanks to all my friends and others who helped me directly or indirectly during this project work. The project development helped me in research and we got to know a lot of new things in the domain.

I am really thankful to her.

Akshat Mathur, 141207

CONTENTS

LIST OF ABBREVIATIONS

MYPA :- My Personal Assistant

NLP :- Natural Language Processing

LIST OF FIGURES

ABSTRACT

This report encapsulate the documentation for the project MYPA(Pronounced as My PA) and its use-cases. MYPA is an in-meeting virtual assistant, Cloud based Django application especially build for organizations wanting to make meetings more productive. This application syncs with our calendar and fetches your meetings events. While in meeting with just a click of a button it starts recording the meeting and gives you live Speech to text conversation, and after the meeting ends it is smart enough to give you smart notes of the meeting which you can share through email or download as text file. This application is implemented in python's Django framework with the integration of multiple technologies and APIs such as google Web Speech API for live Speech to text conversion, NLP has been implemented using NLTK which is a very highly efficient Natural Language Toolkit in python, Twillio APIs for calling in the meeting and recording voice. This project also utilizes the cloud technology as it is deployed over Heroku using Docker. MYPA obsolete the hand-written meeting notes and makes meeting and conferences more productive

CHAPTER 1 INTRODUCTION

1.1 General

The whole idea of MYPA revolves around the a technological advancement of going paperless. Since the last decade we have been hustling to build various techs to save world ecosystem balance in which saving plantation/greenery plays a major role.

According to the research making papers is a major reason for deforestation.

We have already taken may steps to reduce these numbers by building and integrating latest technology into our lives and going paperless, We have kindle for reading which had created a significant market and reduced paper usage without effecting the revenue for the publication, the major benefit of incorporating the paperless technology is

#Automation :- With paperless technologies such as presented in this paper we can automate things which cannot be done with a pen and a paper which eventually saves a lot of time on a cumulative basis, There are many day to day activities which can be made automatic by technologies like this, that might be creating a To-Do list, calendar entries, daily shopping list, taking lecture notes and what not.

Enormous organizations are working day and night just to make themselves a 100% eco-friendly and contributes towards this ecosystem and technologies like this defiantly is a step forward.

1.2) MY PERSONAL ASSISTANT

'MyPA - My Personal Assistant' is an in-meeting virtual assistant. The main reason or the 'WHY' this project was required can be easily understood with the following points.

- Taking handwritten notes is not preferred by most attendees. If a person attends a lot of meetings, noting down all the important points in each of the meeting is tiresome.
- Failing to attend meetings. Even though the attendee has accepted an invitation, last minutes issue may force him to opt-out of the meeting.
- Individual notes may lead to confusion. If each person takes individual notes according to his/her own understanding, the general motive of the meeting is somewhat lost.

So, to solve all the of the issues mentioned above and many more, MyPA comes to the rescue.

The initial idea was to make MyPA as a bot, which gets invited in your meetings, and holla! Rest all is taken care of. The bot goes and attend the meeting with you / in place of you as your personal assistant, records all the conversation and provides you both the text of the meeting and the "smart notes" . It then provides the user the option to mail the notes to all the attendees of the meeting ,so that all of them are on the same page.

Having one month of time, it was a challenging task to incorporate all the requirements. The team came out with a product that proved that such a product is possible and helpful for the end-user as it transfers a part of load from the user to the virtual assistant. The initial idea of making it a fully automatic process was partly achieved in those 30 days, and the following paragraphs, explain the product that was built. (The explanation takes a SCRUM meeting as an example).

The user signs up on MyPA website, after which he provides his google calendar access, which is required for the web-app to keep track of all the meeting, The dashboard screen shows multiple cards, one for one calendar event, on click of which he is taken to the homepage, where on a click of a button, the app starts listening to all the remote meeting conversation. It then make use of Web Speech API to do speech-to-text translation. The translated text is then sent via AJAX calls for the Natural Language Processing wherein with the help of certain Algorithms , the smart note is prepared and displayed live to the user . At the end of the meeting, on a click of a button the notes get mailed to other attendees of the meeting as well.

But that does only half of the job. The other half , i.e the automatic dial-in of the virtual assistant to the meeting is achieved with the help of Twilio calls. The app scans the description to find the meeting ID and using it, makes call at the meeting time into the meeting and records all the conversation. So, if the user is unable to attend a meeting, he gets to listen it later, and by logging in to the app, he can also get the smart text.

So, the above paragraphs explained what the project turned out to be in a month.Now, let us explain you the future scope to make it fully automatic and more user-friendly. Using Google Cloud APIs, the app will automatically start recording ,translating and provide the smart note, without the user having to log in to the app. The user can then check the "smart note" at any point of time and email it by just going to the website.

1.3) PROBLEM STATEMENT

Taking handwritten notes is not something that is preferred by all people during meetings. Some of them tends to remember it all while some others are too lazy for it. Furthermore, many a times not everyone succeeds in attending a meeting. Moreover, individual notes if taken, may lead to confusion as each individual make notes according to his/her own understanding, which in some cases may not be in sync with the theme of the meeting.

1.4) Screen-Shots

Figure1 Login Page

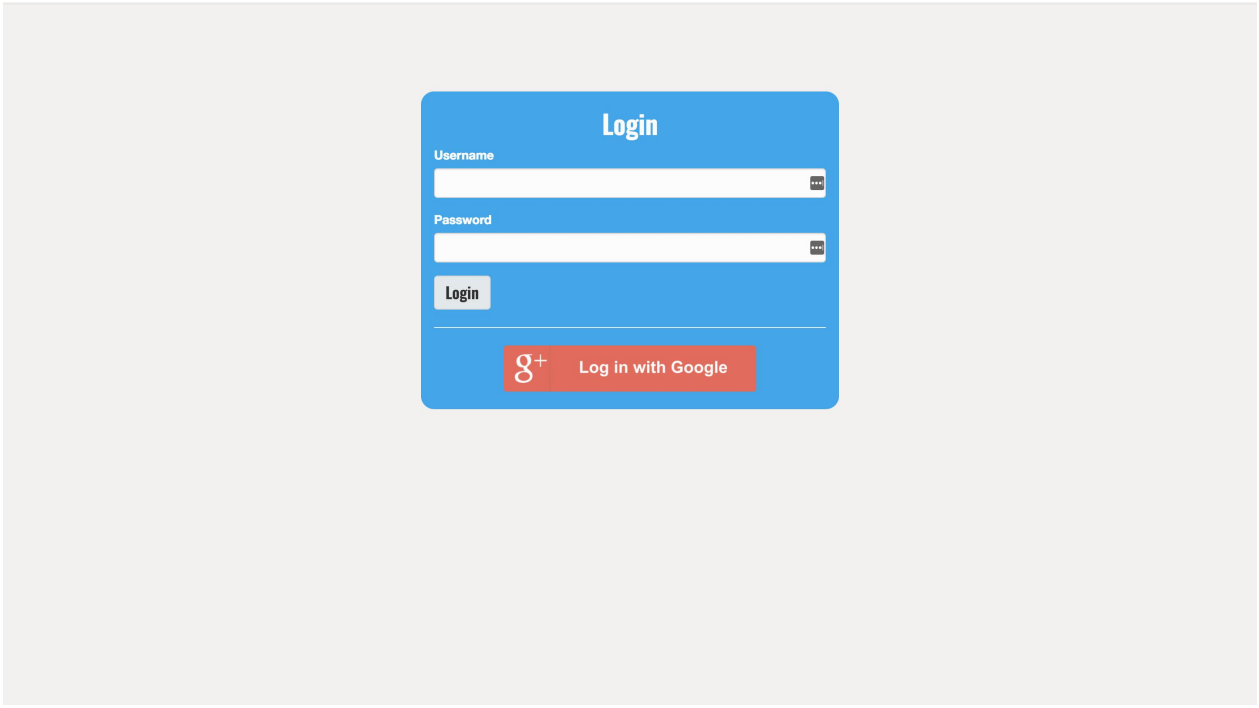


Figure 2 Dashboard

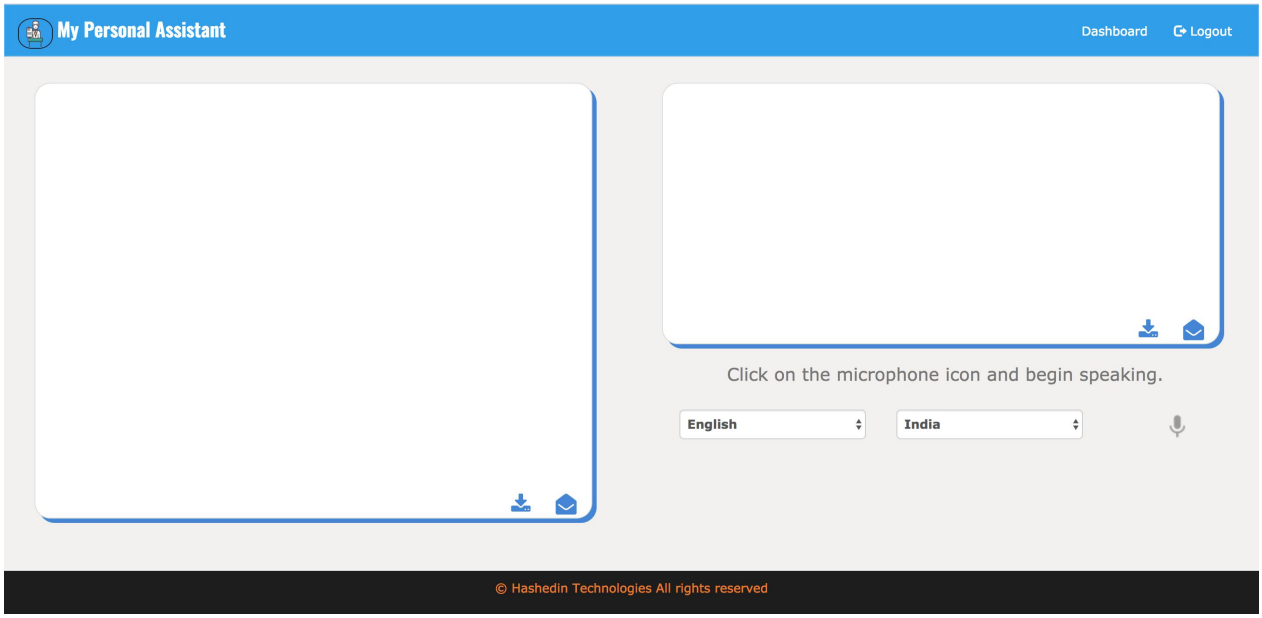


Figure 3 Main Application Page without Demo

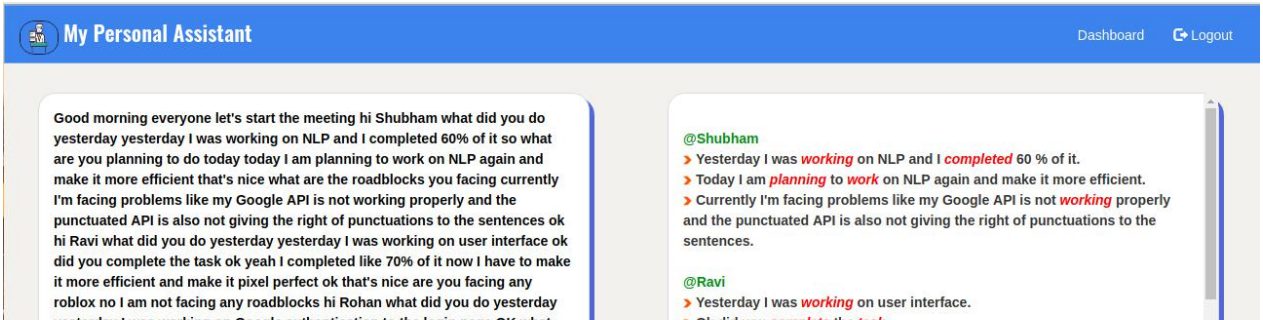


Figure 4 Main Application Page with Demo

1.5) OBJECTIVES

Short-term :- A web application

- That uses Web Speech API to convert the speech from the meeting to text and streams it live.

- This application have implemented NLP algorithms on live text so that the user gets what is important.

- The summary (entities, actionable items, meeting productivity) gets mailed to all the participants after the meeting.

Long-term :-

- Using Google's Premium Speech Recognition API for improving the speech to text conversion

- Training through multiple agents

- Sharing notes through other messaging applications(Whatsapp, iMessages)

- Integrating multiple platforms.

- Speaker Recognition through Machine Learning.

- Provide Dynamic themes.

CHAPTER 2

LITERATURE REVIEW

To fully understand the project and research the best possible solution of the problem a number of magazines, scholarly journals and research papers have been read and investigated in detail. This chapter entails all the findings and important materials from the said sources to be able to design the solution to the stated problem.

Many scientists and research scholars have published various papers related to the given topic. This part contains all the essential extracts from those papers.

2.1 Document Management System:A Notion Towards Paperless Office(2017) by Mahendra K. Ugale, Shweta J. Patil, Vijaya B. Musande

The objective of this paper is to demonstrate how their System waives out the conventional physical filing systems, also discusses the technique and technologies which aids enterprise to move towards a paperless development. Their digital management system helps to keep a records for searching a document, store it, tag it and minimize paper usage. Methods for data storage and data retrieval were also described.

2.2 Bidirectional Recurrent Neural Network with Attention Mechanism for Punctuation Restoration(2016) by Ottokar Tilk, Tanel Aluma ۲e

The aim of this paper is to explain the authors algorithm for adding punctuations to the English languages complete sentences based on neural network and trained data set.

This paper had played a crucial role in the development of this project. This algorithm helped in punctuating the sentence we are getting from the web speech to text API for further implementation of NLP algorithms onto it and creation of smart notes.

2.3 Paperless university – how we can make it work?(2016) by Masuda Isaeva, Yoon, Hyen Young

The paper describes who the Management Information System made the use of paper totally obsolete for University administration processes

Also the reason and benefits of going paperless along with some great examples of how many different Universities are trying to implement Information Systems for reduce the use of paper for their day to day processes. They claimed that the modal presented in the paper will be improved as the IT solution grows.

2.4 Realization and Research of Paperless Examination System based on WEB(2010) by Zhikao Ren, Minghua Liu, Chen Ye,Xiuying Wang, Chuansheng Wang

This paper discusses the current state of digital examination also, describes how technologies can lead to the aim of paperless examination system, their pros and cons.

The maintenance for the system, databases architecture for storage and computation power for the data manipulations.

2.5 SQL Paperless Examination System Design (2010) by Gaoyan Zhang, Haifeng Ke

The aim for this paper is to explain the Paperless Examination System, who it is adding a important value to our modern education system, Who these kind of systems improving the productivity and efficiency of the existing examination system. More so however the current examination system is already a multiple-choice type concept and still we are using pen and paper as a method to execution for such kind of examination. The only flaw for this system could be that it currently does not support subject type answer judgement for that we still will require a person to verify and allocate marks. In the paper author

introduced a SQL based examination that supports paperless exam execution on a practical level.

2.6 A Method to Extract Essential Keywords from a Tweet using NLP Tools (2016) by Tharindu Weerasooriya, Nandula Perera, S.R. Liyanage

In this paper the authors presented how a post on a social media merely around 140 characters or less can call for a great use of Natural Language Processing. This research enhances the machine learning based Stanford CoreNLP Part-of-Speech (POS) tagger with the Twitter model to extract essential keywords from a tweet. The system was enhanced using two rule-based parsers and a corpus. The enhancement caused the system to score a 83.33% in Turing test. Discussed what would be further measure for improvement in the score.

2.7 Design of Paperless Examination System for Princeples of Database Systems (2009) by Gaoyan Zhang, Haifeng Ke

The aim for this paper is to explain the Paperless Examination System, who it is adding a important value to our modern education system, Who these kind of systems improving the productivity and efficiency of the existing examination system. More so however the current examination system is already a multiple-choice type concept and still we are using pen and paper as a method to execution for such kind of examination. The only flaw for this system could be that it currently does not support subject type answer judgement for that we still will require a person to verify and allocate marks. In the paper author introduced a SQL based examination that supports paperless exam execution on a practical level.

2.8 NLP Based Sentiment Analysis on Twitter Data Using Ensemble Classifiers (2015) by Monisha Kanakaraj and Ram Mohana Reddy Guddeti

This paper presents how generic sentiment analysis systems use word set approach as training data for the algorithm, also discussed several machine learning algorithms like Naive Bayes, SVM algorithm Etc. as a source of highly reliable algorithms for classic NLP problems. The authors proposed an approach which is based on NLP(Natural Language Processing) to advance the sentimental analysis by incorporating additional semantics in the feature vector. Due to this the accuracy of prediction increases by many folds because of semantically similar words and context-sense identities. Experiments conducted demonstrate that this approach beats the conventional methods with single machine learning classifier by 3-5%.

2.9 Using NLTK for educational and scientific purposes (2011) by Mykhailo Lobur, Andriy Romanyuk, Mariana Romanyshyn

This paper manages the significance of Natural Language Toolkit for the course of Computational Linguistics and for logical research in the field of regular dialect handling. Idiosyncrasies of Python programming dialect, utilized as a part of Natural Language Toolkit, are depicted. The particular experience of concentrate Natural Language Toolkit over the span of Computational Linguistics is considered.

2.10 VOICE CONTROLLED HOME AUTOMATION SYSTEM USING NATURAL LANGUAGE PROCESSING (NLP) AND INTERNET OF THINGS (IoT) (2017) by Mrs. Paul Jasmin Rani, Jason Bakthakumar, Praveen Kumaar.B, Praveen Kumaar.U and Santhosh Kumar

The essential goal of this paper is to develop a completely utilitarian voice-based Home computerization framework that utilizations Internet of Things, Artificial Intelligence and Natural Language Processing (NLP) to give a financially savvy, effective approach to cooperate with home machines. There are numerous savvy home arrangements in the market that intend to mechanize the fundamental tasks of these home apparatuses utilizing different advancements, for example, GSM (Global System for Mobile), NFC (Near-Field Communication) and so forth. Be that as it may, a large portion of these frameworks center around mirroring the fundamental activity of the electrical switch. Our venture goes for giving a completely robotized voice based arrangement that our clients can depend on, to perform something other than exchanging on/off the machines. The client sends a charge through discourse to the cell phone, which deciphers the message and sends the proper summon to the particular machine. We anticipate executing four

essential home apparatuses as a "Proof-of-Concept" for this task which incorporates Fan, Light, Coffee Machine and Door Alarms. The voice order given by the client is deciphered by the cell phone utilizing Natural Language handling. The cell phone goes about as a focal console; it figures out what activity must be finished by which machine to satisfy the client's demand. The focal comfort may similarly be either a work area application, web application or a PDA application as almost the greater part of the information exchanged can be prepared by the cloud. Nonetheless, for the accommodation of the client and expanded versatile capacities we will utilize an advanced mobile phone in this task. The apparatuses are related with the cell phone through an Arduino Board that sets up the idea of Internet of Things. The Arduino Boards are interfaced with the machines and modified in a way that they react to portable data sources.

2.11 NLP based Intelligent News Search Engine using Information Extraction from e-Newspapers (2014) by Monisha Kanakaraj, Sowmya Kamath S

The authors of the paper are extricating content data from a web news page is a testing errand as a large portion of the E-News content is furnished with help from backend Content Management Systems (CMSs). In this paper, we display a customized news web index that spotlights on building a store of news articles by applying effective extraction of content data from a web news page from differed e-news gateways. The framework depends on the idea of Document Object Model(DOM) tree control for extricating content and altering the website page structure to reject insignificant substance like advertisements and client remarks. We additionally utilize WordNet, a thesaurus of English dialect in view of psycholinguist examines for coordinating the extricated content semantically to the title of the page. TF-IDF (Term Frequency Inverse Document Frequency) is utilized for recognizing the website page pieces conveying data important to the pages title. Notwithstanding the extraction of data, functionalities to accumulate related data from various web news papers and to abridge the assembled data in light of client inclinations have likewise been incorporated. We watched that the framework could accomplish great review and high accuracy for both summed up and particular questions.

2.12 NLTK: The Natural Language Toolkit by Steven Bird Edward Loper

The Natural Language Toolkit is a suite of program modules, informational collections, instructional exercises and activities, covering representative and factual characteristic

dialect preparing. NLTK is composed in Python and appropriated under the GPL open source permit. In the course of recent years, NLTK has turned out to be mainstream in instructing and research. We depict the toolbox and give an account of its present condition of advancement

2.13 Using the Wiki to Deliver Paperless Software Documentation (2011) by Kay Rettich

This paper aim on an examination that was led to assess the conveyance of programming documentation as a dynamic paperless wiki to the clients of a product advancement organization. The favorable circumstances and inconveniences of utilizing a wiki to compose and convey programming documentation were explored. Preparing modules were created to instruct the organization representatives viable strategies for utilizing the organization wiki to cooperatively compose programming client documentation. Proof was additionally given in a white paper that could be conveyed to the clients to demonstrate that accepting their documentation as a wiki would be preferable for them over getting customary static hard-or delicate duplicate documentation. The objective of the venture was to empower the workers to furnish the clients with unrivaled programming documentation as a wiki, which can be consistently refreshed and kept present as the product item develops and takes out the requirement for printing and keeping up a printed copy library of the client documentation.

2.14 An Advanced NLP Framework for High-Quality Text-to-Speech Synthesis (2011) by Catalin Ungurean Dragos Burileanu

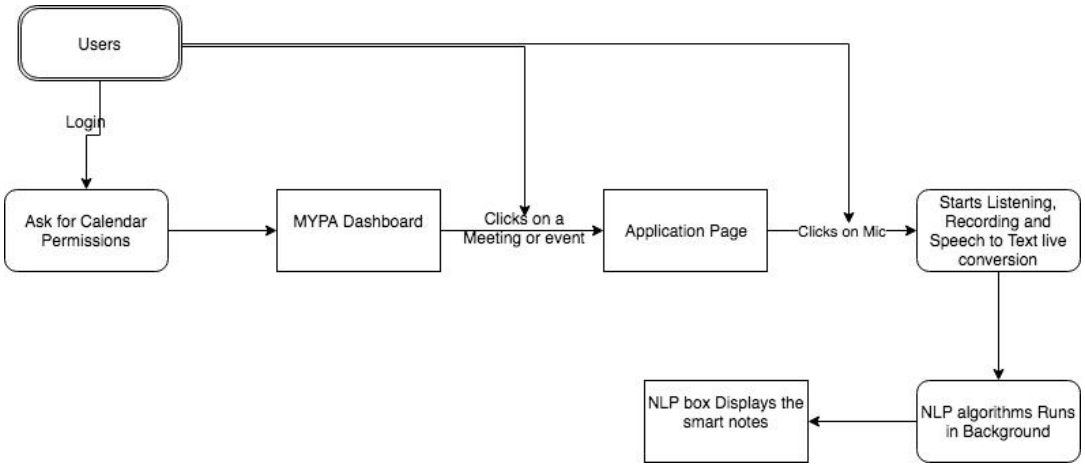
This paper presents a specific end goal to fabricate a TTS (Text-to-Speech) combination framework one must give two key parts: a NLP (Natural Language Processing) arrange, which basically works on the info content, and a discourse age stage to deliver the coveted yield. These two particular levels must trade the two information and orders to create comprehensible and common discourse. As the entire TTS errand depends on numerous particular logical regions, any accomplishment toward institutionalization can limit the exertion and increment the dynamic of the outcomes.

This paper gives a diagram of the NLP organize in the TTS framework for Romanian dialect worked by our group, and portrays the incorporation into the arrangement of SSML (Speech Synthesis Markup Language), as a these days all around perceived standard for TTS record creating and between modules correspondence.

CHAPTER -3 SYSTEM DEVELOPMENT

In this area examined information stream chart, Entity relationship outline. These things are spoken to as graphs with appropriate documentation.

3.1 Data Flow Diagram



Processes

1) WBS and PERT

The first step of building a project ids the planning phase where an estimate is made on how long its gonna take to complete the milestones.

Work Breakdown Structure is the breakdown of the project into a number of iterations. In Agile development the project is done in iterations.

Program Evaluation Review Technique is the calculated estimate for each task based on the estimates given.

2) Trello

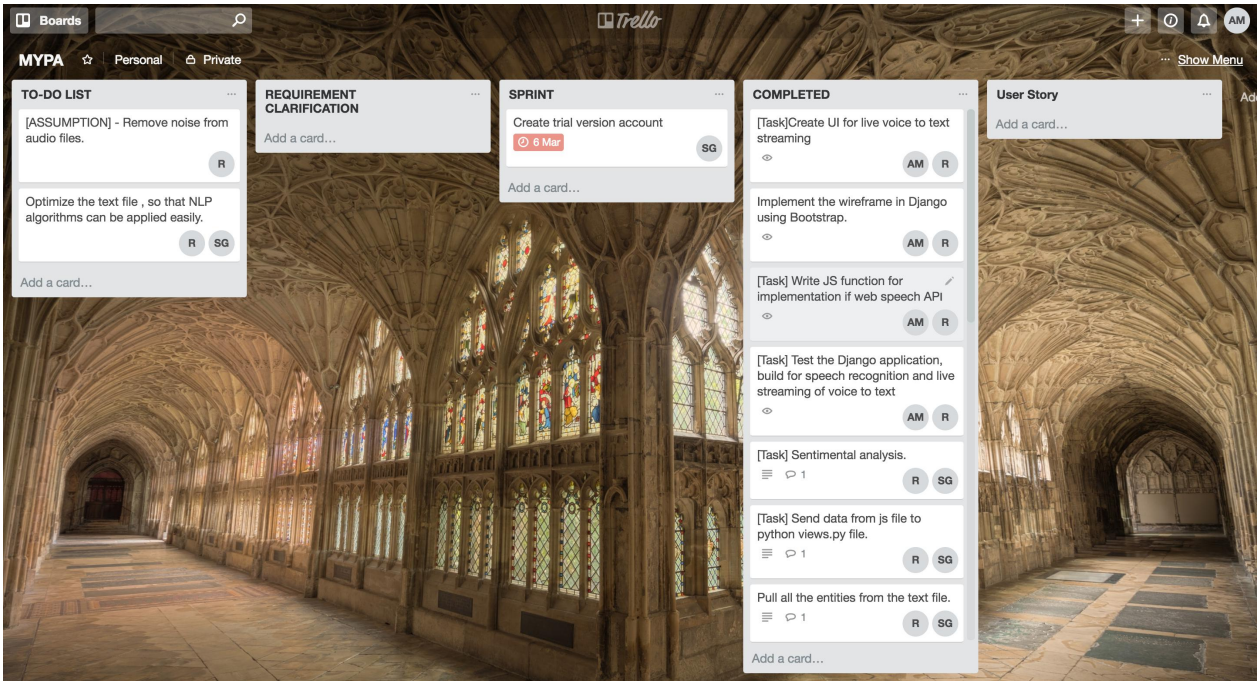


Figure 5 MYPA Trello Board

Trello is the web based project management application we used to keep track of the tasks completed by each member of the team and the tasks each member is working on.

The tasks are segregated in rown based on ther status, whether they are completed, in progress, pending or is a backlog.

3) Sprint

Every cycle in Agile advancement is known as a Sprint. Work to be finished in each run is talked about upon in the arranging period of the task. The length of the dash is chosen purchase the scrum ace. Amid the dash, the group holds every day stand up meeting to examine advance and conceptualize answers for challenges.

Toward the finish of the dash, the group introduces its finished work to the venture proprietor and the task proprietor utilizes the criteria set up at the run arranging meeting to either acknowledge or dismiss the work.

4) Scrum

Scrum is a lithe method to deal with a task, as a rule programming improvement. Dexterous programming advancement with Scrum is regularly seen as a procedure; but instead than survey Scrum as strategy, consider it a system for dealing with a procedure.

The day begins with an exceptional gathering assembled scrum conference where every part is appointed the assignments to be finished by him for the day. We examine the every one of the errands finished the earlier day and accumulations assuming any.

5) Weekly Triage

Triage Meetings are venture gatherings in which open bugs are separated into classes. These gatherings are held to break down deformities and to determine moves to be made on them. Essentially need and seriousness are characterized for the bugs. Different exercises include relegating or dismissing new deformities made from the last triage meeting. Aside from that, current deformities are reassigned if require emerges.

Week by week triage gatherings happen each Friday to track the advance of the undertaking.

6) Deployment

Docker is a PC program that performs working framework level virtualization otherwise called containerization. Docker is principally created for Linux, where it utilizes the asset seclusion highlights of the Linux piece, for example, cgroups and portion namespaces, and an association competent document framework, for example, OverlayFS and others to permit autonomous "compartments" to keep running inside a solitary Linux occasion, staying away from the overhead of beginning and keeping up virtual machines (Vms).

Docker picture is made and pushed on to the heroku application. At that point the relocations are pushed to the heroku database with the goal that all the configuration can be sent out. At that point the information is feeded to the database from heroku administrator.

3.2 Software Development model

The Software Development Model used to build this project is **Agile Software Development**

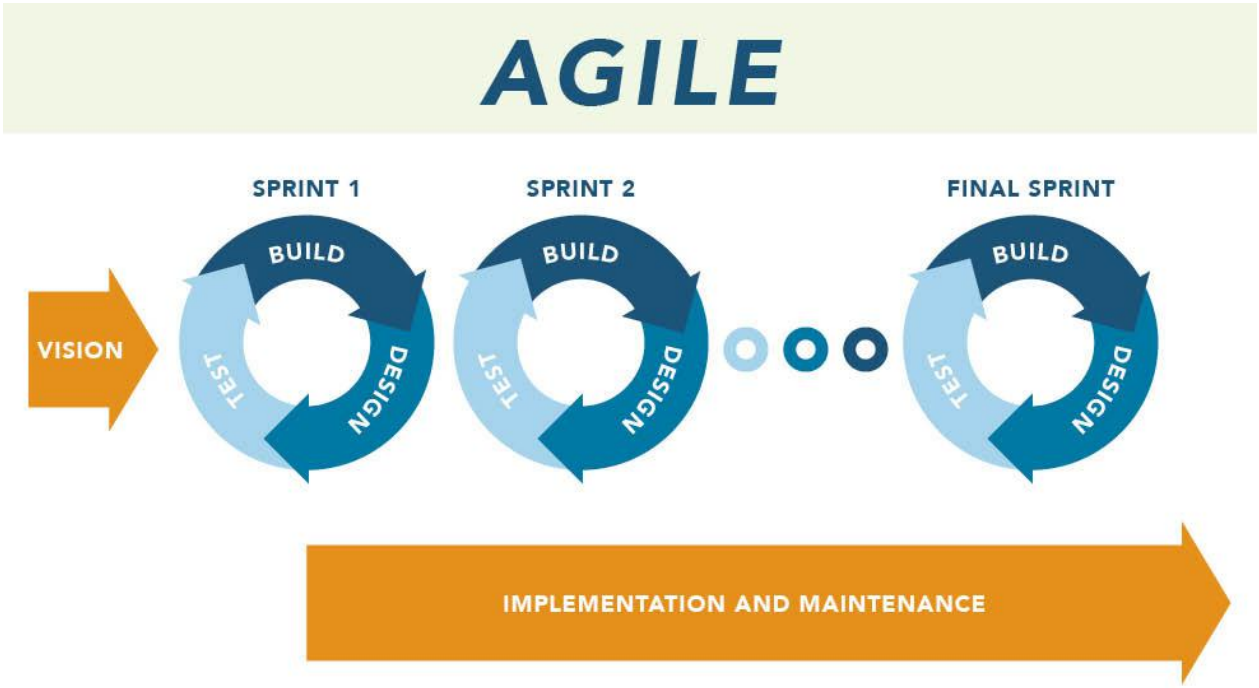


Figure 6: Prototyping Model

3.3 Software requirements

The software required to perform the implementation are

- Windows/Mac/Linux Operating System
- celery==4.0.2
- Django==1.11.1
  django-filter==1.0.2
  django-rest-swagger==2.1.2
  djangorestframework==3.6.3
  whitenoise==3.3.0
- dj-database-url==0.4.2
- dj-email-url==0.0.10
- django-twilio
- coverage==4.4.1
- django-coverage-plugin==1.4.2

- nltk==3.2.5
- numpy==1.14.1
- simplejson==3.13.2
- django-bootstrap-form==3.3
- django-bootstrap3 == 9.1
- social-auth-app-django == 2.1.0
- redis == 2.10.6
- django-twilio
- coverage==4.4.1
- django-coverage-plugin==1.4.2
- nltk==3.2.5
- numpy==1.14.1
- simplejson==3.13.2
- gunicorn==19.7.1
- Logentries==0.17

3.4 System Testing

Framework testing incorporates the testing of the model. Framework testing is the period of utilization that is away to guarantee that the structure works unequivocally and capably before the live task starts. Testing is basic to the achievement of the system. System testing makes the steady doubt that if each one of the parts of the structure are correct, at that point the target will be viably expert. A movement of testing is proficient for the proposed structure before the system is set up for the customer affirmation testing.

3.5 Algorithm

After a period of observation, the detection algorithm then works in a series of simple steps:

1. Ranking Algorithm

In ranking algorithms, we calculate a frequency of each word of text and then give rank to each word. Sort all those words according to their ranks after that pick top words and

determine which sentence contains those word. If the sentence contains that word then sentence considered to be an important sentence. Store all important sentence in separate file.

2) NaiveBayesClassifier Algorithm

Naive Bayes classifiers are a gathering of order calculations in view of Bayes' Theorem. It's anything but a solitary calculation however a group of calculations where every one of them share a typical standard, i.e. each match of highlights being ordered is free of each other.

To begin with, let us consider a dataset.

Consider an anecdotal dataset that portrays the climate conditions for playing a session of golf. Given the climate conditions, each tuple groups the conditions as fit("Yes") or unfit("No") for plaing golf.

Bayes' Theorem

Bayes' Theorem finds the likelihood of an occasion happening given the likelihood of another occasion that has just happened. Bayes' hypothesis is expressed numerically as the accompanying condition:

$P(A|B) = P(B|A)P(A) / P(B)$

2. Relevant and Irrelevant List Algorithm

The relevant list basically contains all the important words and the irrelevant list contains all non-important words. So if a sentence contains more relevant words as compared to irrelevant words then sentence considered to be important.

Natural Language Processing

NLP is a region of innovation and AI required with the communications amongst PCs and human (common) dialects, especially an approach to program PCs to beneficially technique gigantic measures of dialect information.

Difficulties in regular dialect process intermittently include discourse acknowledgment, normal dialect comprehension, and common dialect age.

Since the sketchy "factual unrest" inside the late Nineteen Eighties and center Nineties, a considerable measure of regular dialect process examination has depended vigorously on machine learning.

Once in the past, a few dialect preparing errands for the most part concerned the immediate hand cryptography of guidelines, that isn't ordinarily strong to regular dialect variety. The machine-learning worldview calls rather for abuse connected math legitimate reasoning to mechanically learn such principles through the examination of immense corpora of common true cases (a corpus (plural, "corpora") might be an arrangement of reports, apparently with human or pc explanations).

A wide range of classifications of machine learning calculations are connected to regular dialect process undertakings. These calculations take as info an outsized arrangement of "highlights" that square measure produced from the PC record. some of the most punctual

utilized calculations, similar to call trees, made frameworks of debilitating if-then guidelines relatively like the frameworks of manually written tenets that were then normal. to an ever increasing extent, in any case, examination has fixated on connected math models, that make delicate, probabilistic choices upheld joining genuine esteemed weights to each information highlight. Such models have the favorable position that they'll all out the relative assurance of the numerous totally extraordinary feasible answers rather than only one, producing extra dependable outcomes once such a model is encased as a component of a greater framework.

Frameworks bolstered machine-learning calculations have a few advantages over hand-created rules:

The learning methodology utilized all through machine adapting mechanically represent considerable authority in the first basic cases, wherever as once written work leads by hand it's generally not in any way shape or form clear where the problem should be coordinated.

Programmed learning techniques will make utilization of connected math legitimate reasoning calculations to supply models that square measure strong to new to enter (e.g. containing words or structures that haven't been seen previously) and to off base info (e.g. with incorrectly spelled words or words inadvertently overlooked). For the most part, taking care of such information benevolently with manually written principles—or extra ordinarily, making frameworks of transcribed tenets that make delicate choices—is exceptionally extreme, failing and long.

Frameworks upheld mechanically taking in the standards will be made extra right just by supply extra PC record. In any case, frameworks upheld written by hand standards will exclusively be made extra right by expanding the nature of the standards, that might be a much more intense assignment. especially, there's a breaking point to the nature of frameworks upheld carefully assembled rules, on the far side that the frameworks turn into extra and extra unmanageable. In any case, influencing extra information to contribution to machine-to learning frameworks only needs a comparing increment inside

the scope of worker hours worked, regularly while not imperative will increment inside the nature of the comment strategy.

Tokenizing Words and Sentences with NLTK

- The NLTK module is a monstrous toolbox, went for helping you with the whole Natural Language Processing (NLP) philosophy. NLTK will help you with everything from part sentences from sections, part up words, perceiving the grammatical feature of those words, featuring the primary subjects, and after that even with helping your machine to comprehend what the content is about. In this arrangement, we will handle the field of supposition mining, or sentimental examination.

Stop words with NLTK

This is a clearly huge test, however there are ventures to doing it that anybody can take after. The primary thought, be that as it may, is that PCs essentially don't, and won't, ever comprehend words straightforwardly. People don't either *shocker*. One of the real types of pre-handling will be sifting through pointless information. In common dialect handling, pointless words (information), are alluded to as stop words.

Stemming words with NLTK

Stemming is a kind of normalizing technique. Numerous varieties of words convey a similar importance, other than when tense is included.

The motivation behind why we stem is to abbreviate the query, and standardize sentences.

Consider:

I was taking a ride in the auto.

I was riding in the auto.

This sentence implies a similar thing. in the car is the same.

Chunking with NLTK

We can do what is called piecing, and assemble words into ideally significant lumps. One of the principle objectives of piecing is to gather into what are known as "thing phrases." These are expressions of at least one words that contain a thing, perhaps some engaging words, possibly a verb, and possibly something like a qualifier.

Chinking with NLTK

You may find that, after a lot of chunking, you have some words in your chunk you still do not want, but you have no idea how to get rid of them by chunking. You may find that chinking is your solution. Chinking is a lot like chunking, it is basically a way for you to remove a chunk from a chunk. The chunk that you remove from your chunk is your chink.

The code is very similar, you just denote the chink, after the chunk, with }{ instead of the chunk's {}.

3. Lemmatizing with NLTK

A fundamentally the same as activity to stemming is called lemmatizing. The significant contrast between these is, as you saw prior, stemming can regularly make non-existent words, though lemmas are genuine words.

CHAPTER 4 PERFORMANCE ANALYSIS

Execution benchmarking

It's horrible simply speculating or accepting where the wasteful aspects lie in your code.

Django apparatuses

django-troubleshoot toolbar is an exceptionally convenient instrument that gives bits of knowledge into what your code is doing and how much time it spends doing it. Specifically it can demonstrate to all of you the SQL inquiries your page is producing, and to what extent every one has taken.

Outsider boards are likewise accessible for the toolbar, that can (for instance) give an account of store execution and layout rendering times.

Reserving

Frequently it is costly (that is, asset eager and moderate) to process an esteem, so there can be enormous advantage in sparing the incentive to a rapidly available reserve, prepared for whenever it's required.

It's an adequately critical and capable system that Django incorporates a complete reserving structure, and other littler bits of storing usefulness.

The storing system

Django's storing system offers extremely critical open doors for execution picks up, by sparing dynamic substance with the goal that it doesn't should be figured for each demand.

For accommodation, Django offers diverse levels of reserve granularity: you can store the yield of particular perspectives, or just the pieces that are hard to create, or even a whole site.

Executing storing ought not be viewed as an other option to enhancing code that is performing ineffectively on the grounds that it has been composed seriously. It's one of the last strides towards delivering admirably performing code, not an alternate way.

cached_property

It's normal to need to call a class occurrence's strategy more than once. On the off chance that that capacity is costly, at that point doing as such can be inefficient.

Utilizing the cached_property decorator spares the esteem returned by a property; whenever the capacity is approached that occurrence, it will restore the spared esteem as opposed to re-processing it. Note this lone chips away at strategies that take self as their lone contention and that it changes the strategy to a property.

Certain Django segments additionally have their own particular storing usefulness; these are talked about beneath in the areas identified with those segments.

HTTP execution

Middleware

Django accompanies a couple of accommodating bits of middleware that can help streamline your site's execution. They include:

ConditionalGetMiddleware

Adds bolster for current programs to restrictively GET reactions in light of the ETag and Last-Modified headers. It additionally ascertains and sets an ETag if necessary.

GZipMiddleware

Packs reactions for every single present day program, sparing data transmission and exchange time. Note that GZipMiddleware is as of now considered a security chance, and is helpless against assaults that invalidate the insurance gave by TLS/SSL. See the notice in GZipMiddleware for more data.

Sessions

Utilizing stored sessions

Utilizing reserved sessions might be an approach to expand execution by disposing of the need to stack session information from a slower stockpiling source like the database and rather putting away as often as possible utilized session information in memory.

Static records

Static records, which by definition are not dynamic, make a great focus for streamlining picks up.

CachedStaticFilesStorage

By exploiting web programs' reserving capacities, you can dispense with organize hits altogether for a given record after the underlying download.

CachedStaticFilesStorage annexes a substance subordinate tag to the filenames of static records to make it alright for programs to store them long haul without missing future changes - when a document changes, so will the tag, so programs will reload the benefit naturally.

"Minification"

A few outsider Django apparatuses and bundles give the capacity to "minify" HTML, CSS, and JavaScript. They evacuate superfluous whitespace, newlines, and remarks, and abbreviate variable names, and in this way lessen the extent of the archives that your site distributes.

Layout execution

Note that:

- using {% piece %} is speedier than utilizing {% incorporate %}

- heavily-divided layouts, gathered from numerous little pieces, can influence execution

The stored layout loader

Empowering the stored layout loader frequently enhances execution definitely, as it abstains from aggregating every format each time it should be rendered.

CHAPTER 5 CONCLUSION

5.1) Conclusion

My Personal Assistant is a web-based application, which listens entire conversation happening in a meeting and convert it into text using API. Our application has implemented NLP algorithms which takes the raw text as an input and provides all the important details that have been talked at the meeting. So this application completely eliminates the need of the third person known as the assistant, who write down all the important information.

This application has a huge potential in reducing the paper usage for Industries as well as this concept and technology can be implemented in various fields where we are currently dependent on papers. Following are some of the usecases for the technology introduced in this report

Making generic automated lecture notes for students.

Replacement of typists in Courts for documenting the case

Speech Sentimental Analysis and many more

While this application focuses on making Enterprise level meeting more productive, it defiantly solves the problem like wasting time on taking notes, having generic agenda for the meeting misguided due to different perspectives from different people present in the meeting by generation of smart notes after productivity and sentimental analysis of the complete meeting.

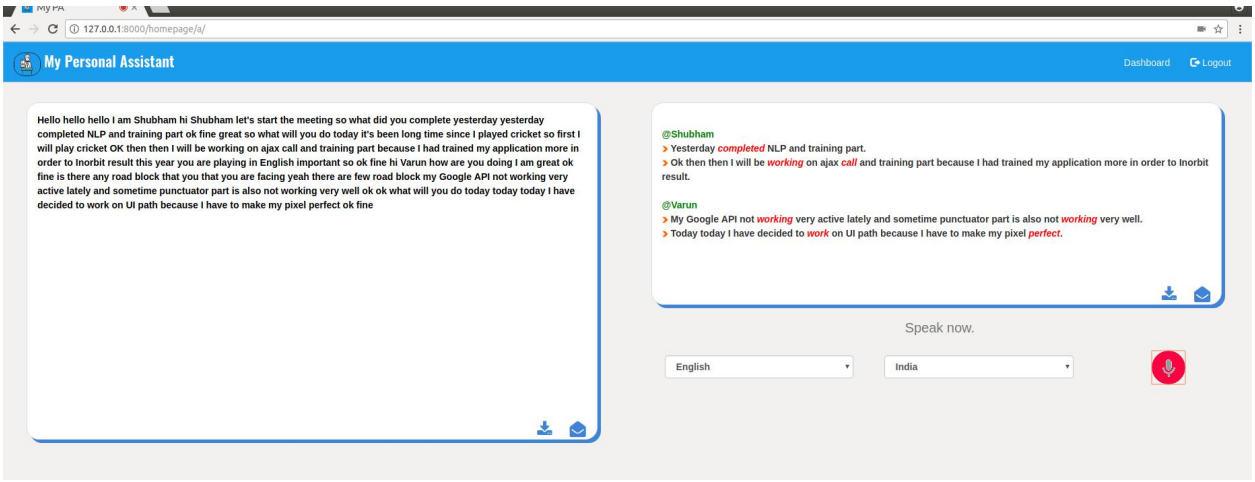


Figure 6 Final Result

Future Scope

Using Google's Premium Speech Recognition API for improving the speech to text conversion.

- Training through multiple agents.
- Sharing notes through other messaging applications.
- Integrating multiple platforms.
- Speaker Recognition through Machine Learning.
- Provide Dynamic themes.

REFERENCES

[1] M. K. Ugale, S J. Patil, V. B. Musande, Document Management System:A Notion Towards Paperless Office, IEEE 1st International Conference on Intelligent Systems and Information Management (ICISIM), 2017

[2] O Tilk, T Aluma ɣe, Bidirectional Recurrent Neural Network with Attention Mechanism for Punctuation Restoration, Interspeech, 2016

[3] M Isaeva, Yoon, H Young, Paperless university – how we can make it work?, IEEE15th International Conference on Information Technology Based Higher Education and Training (ITHET), 2016

[4] Z Ren, M Liu, C Ye, X Wang, C Wang, Realization and Research of Paperless Examination System based on WEB, IEEE 2nd International Conference on Advanced Computer Control (ICACC), 2010

[5] G Zhang, H Ke, SQL Paperless Examination System Design, IEEE Second International Conference on Computer Modeling and Simulation, 2010. ICCMS '10.,2010

[6] T Weerasooriya, N Perera, S.R. Liyanage, A Method to Extract Essential Keywords from a Tweet using NLP Tools,IEEE Sixteenth International Conference on Advances in ICT for Emerging Regions (ICTer), 2016

[7] G Zhang, H Ke, Design of Paperless Examination System for Princeples of Database Systems, ICRCCS '09. International Conference on Research Challenges in Computer Science, 2009

[8] M Kanakaraj and R M R Guddeti, NLP Based Sentiment Analysis on Twitter Data Using Ensemble Classifiers, IEEE's 3rd International Conference on Signal Processing, Communication and Networking (ICSCN), 2015

[9] M Lobur, A Romanyuk, M Romanyshyn, Using NLTK for educational and scientific purposes, IEEE's 11th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM), 2011

[10] Mrs. P J Rani, J Bakthakumar, P Kumaar.B, P Kumaar.U and S Kumar, Voice Controlled Home Automation System Using Natural Language Processing (NLP) And

Internet of Things (IoT), IEEE's Third International Conference on Science Technology Engineering & Management (ICONSTEM), 2017

[11] M Kanakaraj, S K S, NLP based Intelligent News Search Engine using Information Extraction from e-Newspapers, IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), 2014

[12] S B E Loper, NLTK: The Natural Language Toolkit, NLTK's official Documentation

[13] K Rettich, Using the Wiki to Deliver Paperless Software Documentation, IEEE International Professional Communication Conference, 2011

[14] C U D Burileanu, An Advanced NLP Framework for High-Quality Text-to-Speech Synthesis, IEEE 6th Conference on Speech Technology and Human-Computer Dialogue (SpeD), 2011