Natural Language Processing
Digital Assessment 1

Name: Om Ashish Mishra

Registration Number: 16BCE0789

Slot: C1

The Question:

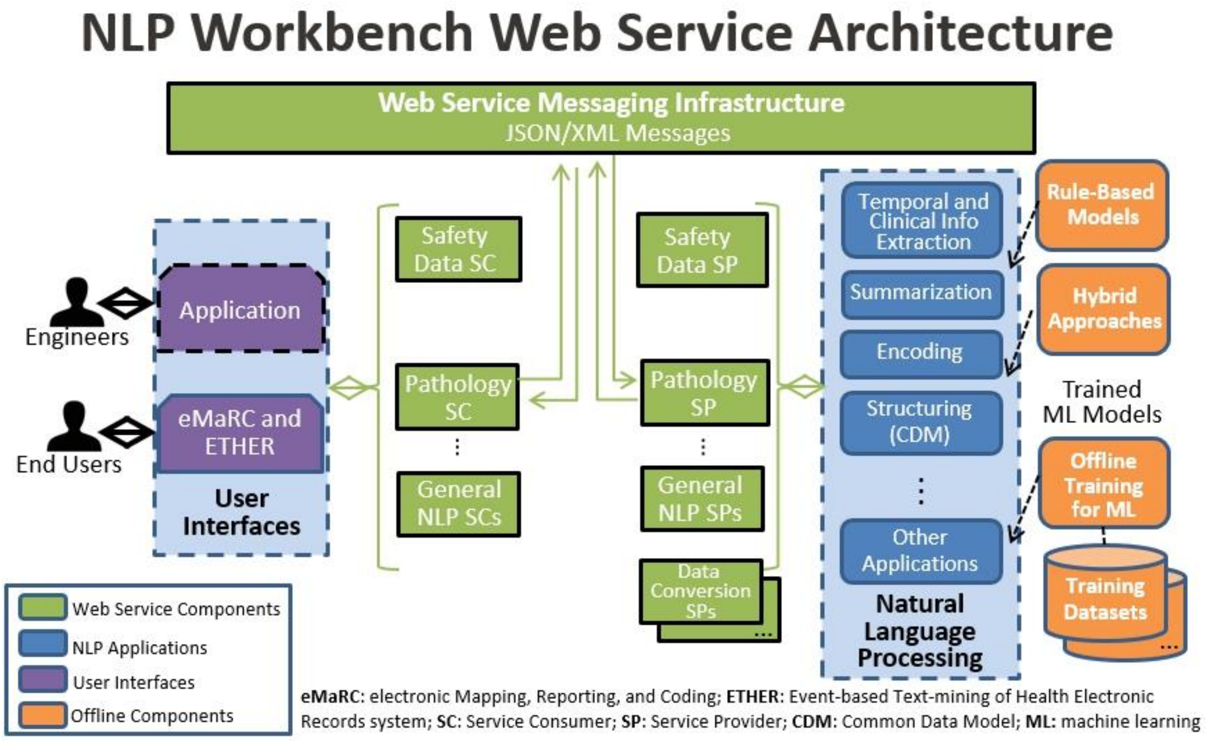
Develop a Natural Language Processing system for an Information Retrieval from web, Chat bot and machine translation.

- System Architecture
- Output Modules
- Application Design
- Explain with techniques and challenges

The Answer:

Web:

System Architecture:



eMaRC: electronic Mapping, Reporting, and Coding; **ETHER**: Event-based Text-mining of Health Electronic Records system; **SC**: Service Consumer; **SP**: Service Provider; **CDM**: Common Data Model; **ML**: machine learning

Output Modules:

User Interfaces
The NLP Workbench will provide access to two types of users—

- Engineers will have access to tools and features needed to develop NLP applications and address specific domain needs.
- End users will use the NLP applications developed by the engineers. For example, end users will use eMaRC Plus and ETHER External (Event-based Text-mining of Health Electronic Records system) interfaces to access the corresponding NLP web services developed by CDC and FDA engineers to process the unstructured pathology and safety data, respectively.

Web Service Components
In the web service messaging infrastructure, web service consumers (safety data, pathology, and general NLP service consumers) send requests consisting of JSON or XML messages to web service providers (safety data, pathology, general NLP, and data conversion service providers), which respond in kind. The web service messaging infrastructure interacts with the user interfaces and the NLP applications.

NLP Applications
NLP applications include extracting temporal and clinical information, summarizing this information, encoding it, and structuring it in the Common Data Model. Other NLP applications may be developed as well.

Offline Components

- Rule-based models will be used to summarize the temporal and clinical information.
- Hybrid approaches will be used to structure the encoded in the Common Data Model.
- Trained machine learning models will be used to develop offline training for machine learning and training datasets.

Application Design:

Apple’s Siri, IBM’s Watson, Nuance’s Dragon etc, there is certainly have no shortage of hype at the moment surrounding NLP. Truly, after decades of research, these technologies are finally hitting the scientist for utilizing in both consumer and enterprise commercial applications.

NLP strives to enable computers to make sense of human language.

Explain with techniques and Challenges:

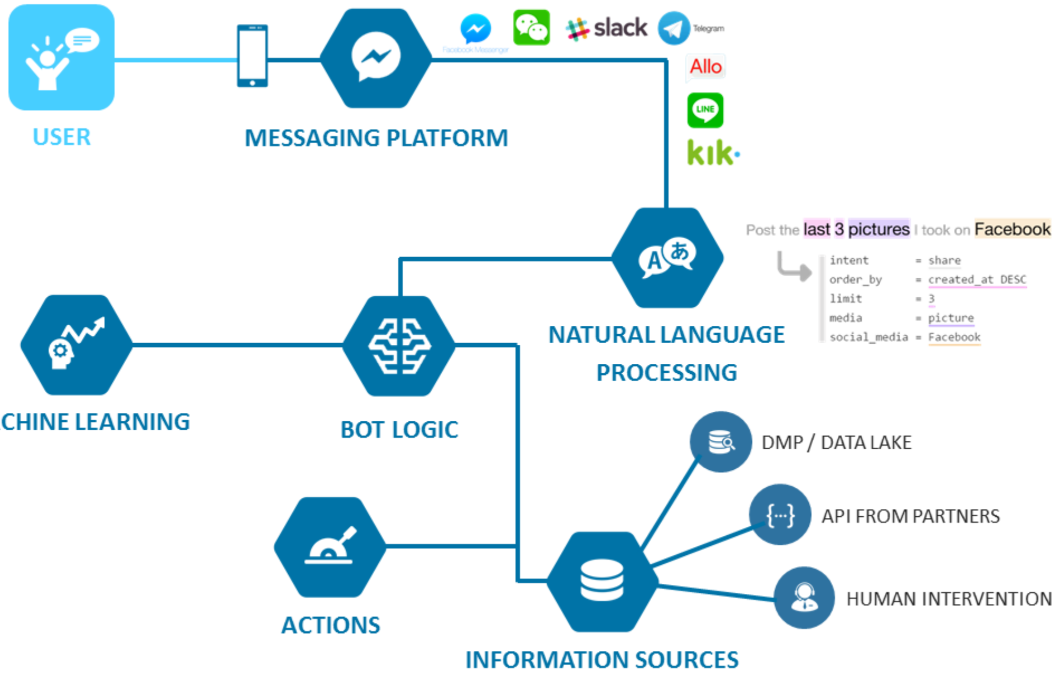
LDA is a core NLP algorithm used to take a group of text documents, and identify the most relevant topic tags for each. LDA is perfect for anybody needing to categorize the key topics and tags from text. Many web developers use LDA to dynamically generate tags for blog posts and news articles. This LDA algorithm takes an object containing an array of strings, and returns an array of objects with the associated topic tags.

The site map algorithm starts with a single URL, and crawls all the pages on the domain before returning a graph representing the link structure of the site. You can use the site map algorithm for many purposes, such as building dynamic XML site maps for search engines, doing competitive analysis, or simply auditing your own site to better understand what pages link where, and if there are any orphaned, or dead-end pages.

NLP is a good field to start research .There are so many component which are already built but not reliable. This is the current snapshot for NLP challenges ,Still companies like Google and Apple etc are making their own efforts .They are solving the problems and providing the solutions like Google virtual Assistant.

Chat bot:

System Architecture:



```
intent       = share
order_by     = created_at DESC
limit        = 3
media        = picture
social_media = Facebook
```

Output Modules:

- Users interact through a **device** on a **messaging platform**, his message is processed through **NLP.**
- Then the bot can launch an **action**, answer with real-time information from a **database/API**, or **handover to a human.**
- The more message he receives, the more the bot improves : it’s called **machine learning**. Sometime a human helps the bot, it’s called **supervised learning**.

Application Design:

- **API.ai for NLP and Supervised Learning** : great small talk feature, best platform for French language, beautiful interface…
- **Dashbot for Analytics and Human Takeover** : very detailed stats, easy integration, transcripts
- **Botpress as a NodeJS framework** : it’s quick to setup and pretty flexible

- **FbMessenger for messaging interface** : 1.2 billion users, great conversational UX

- **Airtable as a database** that can be updated by non-tech people (and sometime MongoDB if needed)

- And many APIs depending on the project (Spotify, Youtube, Google Maps…)

Explain with techniques and challenges:

1. Context in Chatbots

The key to the evolution of any chatbot is its integration with context and meaningful responses, as conversation without any context would be vague. It becomes challenging for companies to build, develop and maintain the memory of bots that offers personalized responses.

That's when AI technologies like Machine Learning or NLP- Natural Language Processing come into the picture and overcome the challenge of understanding the depth of conversation; up-to an extent. NLP understands the databases and data sets when bots are structured, in predefined sequential order and then converts it into a language that users understand.

However, humans don't interact in a defined order, as a result intelligent slot filling, which stores the preferences of the regular users is the alternative to maintain the memory of a bot effectively. This insures that your virtual agents are not interacting in the same old predefined order but in a more personalized fashion.

2. Limited User Attention

Users have limited time span for their queries and expect lightning-fast replies. It's quite challenging for firms to develop chatbots that holds user's attention till the end.

Conversational UI, here plays an important role in exhibiting human like conversations and better customer experiences. It initiates interactions to be more social than being technological in nature. The conversations as a result, should be natural, creative and emotional in order for your chatbot to be successful. In some cases, however a machine wouldn't always render the same empathy that a human could and this is when a human replacement should take care of the user's request.

3. Chatbot Testing

Chatbot testing is another main issue where most of the complexity lies. Chatbots are continuously evolving due to its upgradation in natural language models. Thus, it becomes vital to test and run chatbot to check its accuracy. Testing a chatbot will depend on what type of method you want to experiment.

- First method involves automated testing of chatbots. There are many automation testing platforms like Zypnos, TestyourBot, Bot Testing, Dimon etc. These platforms allow detailed reports of the results and coding of test scripts, which could be run for all the test cases.

- The other method involves testing of conversational logic i.e. manual testing executed by a closed group of testers. They act as users and check the bot for all the unexpected slots possible. This method can be time consuming and partially accurate. However, it has its own benefits that outrage automation, by checking the logic against human conversations.

The best alternative is to combine both the methods to insure that your users are being served better.

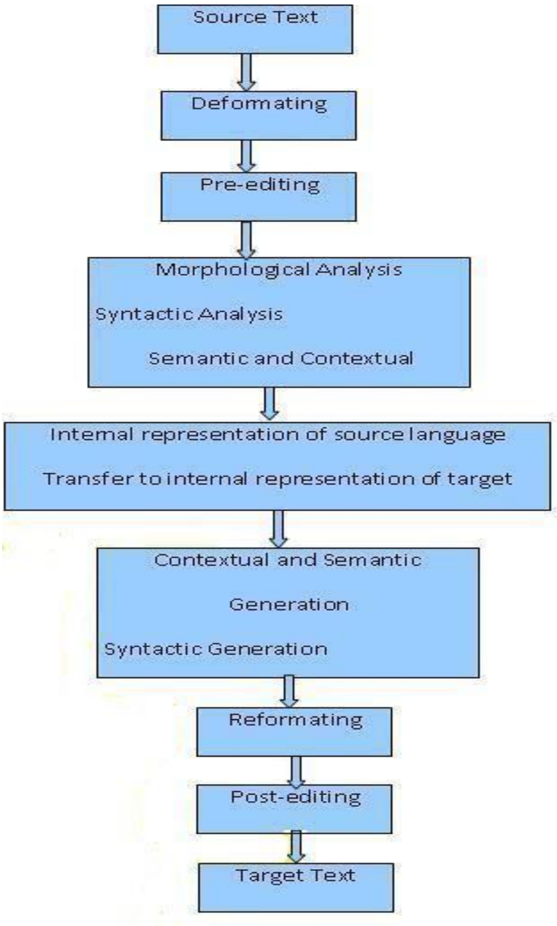
4. Viability of Data

There is no point of having lots of data, intelligent slot filling or technologically advanced chatbot- if it actually doesn't deliver the USP of your organisation. It is vital for a chatbot to not only be enriched with meaningful data, but also to be equipped to deliver the brand identity to your target audience. In order to check the viability of your virtual agents you should consider asking yourself the following:

Machine Translation:

System Architecture:

Machine translation is the process of translating from source language text into the target language. The following diagram shows all the phases involved.



Text Input: This is the first phase in the **machine translation process** and is the first module in any MT system. The sentence categories can be classified based on the degree of difficulty of translation. Sentences that have relations, expectations, assumptions, and conditions make the MT system understand very difficult. Speaker's intentions and mental status expressed in the sentences require discourse analysis for interpretation. This is due to the inter-relationship among adjacent sentences.

Deformatting and Reformatting: This is to make the machine translation process easier and qualitative. The source language text may contain figures, flowcharts, etc that do not require any translation. So only translation portions should be identified. Once the text is translated the target text is to be reformatted after post-editing. Reformatting is to see that the target text also contains the non-translation portion.

Pre-editing and Post Editing: The level of pre-editing and post-editing depend on the efficiency of the particular MT system. For some systems segmenting the long sentences into short sentences may be required. Fixing up punctuation marks and blocking material that does not require translation are also done during pre-editing. Post editing is done to make sure that the quality of the translation is up to the mark. Post-editing is unavoidable especially for translation of crucial information such as one for health.

Analysis, Transfer and Generation: Morphological analysis determines the word form such as inflections, tense, number, part of speech, etc. Syntactic analysis determines whether the word is subject or object. Semantic and contextual analysis determine a proper interpretation of a sentence from the results produced by the syntactic analysis. Syntactic and semantic analysis are often executed simultaneously and produce syntactic tree structure and semantic network respectively. This results in internal structure of a sentence. The sentence generation phase is just reverse of the process of analysis.

Parsing and Tagging: Tagging means the identification of linguistic properties of the individual words and parsing is the assessment of the functions of the words in relation to each other.

Semantic and contextual Analysis and Generation: Semantic analysis composes the meaning representations and assign them the linguistic inputs. The semantic analysers uses lexicon and grammar to create context independent meanings. The source of knowledge consists of meaning of words, meanings associated with grammatical structures, knowledge about the discourse context and common-sense knowledge. **Output Modules**

There are many ways to build a machine translation system using different algorithms eg. Bidirectional RNN, or encoder-decoder model etc. The main motive of machine translation is to translate text or speech from one language to another. Traditionally, it involves large statistical models developed using highly sophisticated linguistic knowledge.

Application:

Machine translation in industry for business use: Although big players like Google Translate and Microsoft Translator offer near-accurate, real-time translations, some "domains" or industries call for highly-specific training data related to the particular domain in order to improve accuracy and relevancy. Here, generic translators would not be of much help as their machine-learning models are trained on generic data.

These applications are used by small, medium and large enterprises. These solutions, although automated for the most part, still depend on human translators for pre- and post-editing processes. Some fields that warrant domain-specific machine translation solutions are Government, software and technology, military and defence, healthcare, finance, legal, EDiscovery and e-commerce.

Online/App Machine Translation for consumer use: These machine learning applications perform instant translation for textual, audio, and image files (images of words on screens, papers, signboards, etc.) from a source language into a target language. They are usually lightweight, cloud-based apps or wearable devices that are typically trained on crowd-sourced data. These are mostly used by individual consumers, such as travelers, students, etc. Real-time translation applications most commonly offer Text-to Text, Text-to-speech, Speech to text, speech to speech, and image to text.

B2B Domain Specific Machine Translation: Over the years, multiple organizations have cropped up with intelligent services and products offering domain-specific machine translation. General translators, that are usually used in consumer applications, such as Google Translate, may have difficulty translating data related to specific business domains, which have their own nuances and terminologies.

These business domains are vast and include industries like technology, software, automotive, e-discovery, legal, financial, military & defence, healthcare, e-learning, ecommerce institutions, and so on.

A machine translation system can be adapted to a specific domain by using training data from the same domain. For example, in order to adapt an MT system for the legal domain, training data including the most commonly used contextual terms, keywords, phrases, terminology, etc., in the legal domain are compiled into corpora, which act as an exhaustive data repository for the MT system to refer to and train on.

Challenges

Lexical challenges: To fully understand the complexity of designing a MT system, it is necessary to analyse the functions of the human brain during a translation process.

When human beings translate they usually start by attempting to decipher the source text on three levels:

- Semantic level: understanding words out of context, as in a dictionary.
- Syntactic level: understanding words in a sentence.
- Pragmatic level: understanding words in situations and context.

Syntactic challenges: Structural differences between languages also present challenges for machine translation. Languages often differ in the basic word order of Subject-Verb-Object. English, Bahasa Malaysia and Chinese are SVO (Subject-Verb-Object) languages, meaning that the verb usually comes in between the subject and object. In contrast, Japanese and Hindi are SOV (Subject-Object-Verb) languages while Arabic is a VSO (Verb-Object-Subject) language.

Therefore, developing a system for language pairs with different word orders such as English to Japanese will be more difficult as compared to development for language pairs with the same word orders such as English to Bahasa Malaysia.

Additionally, tenses that exist in one language may not exist in another language. English, for example, has explicit present progressive and present structure, whereas Arabic has only one tense that encompasses both of these English structures.