NLP chatbot for Discharge Summaries

Harsh Lal
UCSD(CSE)
PID: A53244610
Email: hlal@ucsd.edu

Soheil Karimi
UCSD(CSE)
PID: A11170099
Email: skarimik@ucsd.edu

Thomas Liu
UCSD(CSE)
PID: A10276714
Email: tyl015@ucsd.edu

Ali Alattar
UCSD(SOM)
PID: A11818663
Email: aalattar@ucsd.edu

Contents

I Introduction 1

II Related Work 2

III Dataset 2
III-A **Exploratory Analysis** . . . 3

IV Feature Engineering 5
IV-A **Preprocessing** . . . 5
IV-B **N-gram embeddings** . . . 5

V System Architecture 5
V-A **Topic Modeling** . . . 6
V-B **Latent Dirichlet allocation** . . . 6
V-C **Non-Negative Matrix Factorization** . . . 6
V-D **Discharge Summary Chatbot** . . . 6

VI Evaluation Metrics 8

VII Experiments & Results 8

VIII Future Scope 10

IX Conclusion 10

References 10

Abstract

One of the most important goals in healthcare is improving the quality of health and service rendered by healthcare industry. There has been a growing demand for ease of access to healthcare, and this necessitates the development of healthcare technology with Human Computer Interaction(HCI) design in mind. HCI usage can help both patients and healthcare providers as well[3]. With a wealth of information available to patients and providers, it can be difficult to find relevant information in a timely manner, especially for providers with busy schedules and patients with limited health literacy. We propose a Human Computer Interaction system that can answer patientś question and help improve the quality of healthcare services provided to an individual. We make use of Natural Language Processing, Machine Learning and Big Data technologies to provide a feasible deployable solution. for the same.

I. Introduction

In todays information age, a wealth of information is stored at the individual level. This is particularly pertinent in the context of health-care, where the advent of the electronic medical record has lead to an unprecedented increase in collection and recording of individualized data. In data abstracted from the electronic health record (EHR), information can is found in various forms - *Structured data* such as lab results or anesthesia times which are recorded in defined fields or, *Unstructured data* as is the case with the free text in provider notes.[2]

Due to the sheer quantity of information available, it is not feasible to collect and process this data manually or by a human interpreter. Instead, clinicians and researchers are increasingly utilizing computers and technology to assist with data abstraction and annotation. While computer algorithms can relatively easily process structured data, they meet considerable challenges when dealing with unstructured data. In a healthcare setting researchers typically leverage the processing power of

computers using language modeling and comprehension techniques. Most of these techniques fall under the broad umbrella of Natural Language Processing. Additionally, these techniques can be coupled with machine learning and intelligent extraction algorithms to efficiently identify patterns in large amounts of data.

One of the prime goals of healthcare is providing high quality service to the patients, there has been a growing demand for ease of access to healthcare. This necessitates the development of healthcare technology with Human Computer Interaction(HCI) design in mind. HCI usage can help both patients and healthcare providers to use technology in a seamless and non-overwhelming manner. So we propose a Natural Language Processing chatbot that serves as an HCI component and provides an easy way for patient to analyze data from their clinical notes. A sophisticated version of the same can help the scientists and doctors with familiarizing themselves with their patients in the fastest way possible. Any specific information would be available with a quick search request. This may also be useful for efficiently communicating to patients the details of their diagnoses without requiring human interaction.

We start with talking about related work in the field in section II. After that we give a brief overview of the dataset in section III, we then present different feature engineering techniques in section IV, after that we present the overall system architecture in section V. We present a brief explanation of different evaluation metrics used in the section VI. We finally present different experiments and results for the task in section VII, and wrap up the paper with future scope in section VIII and conclusions in section IX.

II. Related Work

In today's world, there is a rising interest about enhancing the interface usability of applications. Human machine as a technology integrates different areas, and the computational methodologies facilitate communication between users and computers using natural language. In this context we define chatbot, as a conversational agent that interacts with users, turn by turn using natural language. Different chatbots or human-computer dialogue systems have been developed using text communication starting from *ELIZA* that simulates a psychotherapist, and then *PARRY* which simulates a paranoid patient [11].

Another chatbot *Erica* is developed for a dental practice in Netherlands. This online assistant is used to answer frequently asked questions of patients and visitors on the website. Among others, Erica has the important task to answer questions about free dental billing rates (a third part integration) [11]. These days, different tools and technologies are being utilized, to provide a convenient and accessible health service experience [11]. Now-a-days with growing data, it is getting increasingly difficult for having personalized time with doctors and get queries about the health conditions answered. Also previous work with application of chatbot to help patients for counseling by answering questions has been successful [12].

One of the recent example of chatbots in a healthcare setting was shown where chatbots can be used to help diabetic patients control their diabetes and receive advice. A healthcare conversational agent was proposed that will allow users to input their health queries. This can facilitate instant replies to users, when they usually are required to wait long periods of time. In response to a user's input that does not match the keywords specified, the paper specified that a doctors contact details will be provided [12]. Our work is motivated based on same lines wherein a conversational agent can help patients and improve the overall experience of a healthcare setting.

III. Dataset

We have used MIMIC-III (Medical Information Mart for Intensive Care) Critical Care Database for the analysis. MIMIC-III is a large publicly-available database comprising de-identified health-related data associated with over forty thousand patients who stayed in critical care units of the Beth Israel Deaconess Medical Center between 2001 and 2012. The database includes information such as demographics, vital sign measurement made at bedside, laboratory test results, procedures, medications, caregiver notes, imaging reports and mortality.[4]

We will be using the data from the table "NOTEEVENTS" for our analysis. The table contains de-identified notes, including nursing and physician notes, ECG reports, imaging reports, and discharge summaries. It has a total of 2,083,180 rows in total. We filtered these by *Category* "Discharge Summary" to get a reduced data set of about 50,000 that contains all the discharge summary notes. We use this data for further analysis.

NOTEEVENTS table		
Field Name	Type	Description
ROW ID	INT	Primary Key
SUBJECT ID	INT	Unique to patient
HADM ID	INT	Hospital stay
CHARTDATE	TIMESTAMP(0)	Charting date
CATEGORY	VARCHAR(50)	Type of note
DESCRIPTION	VARCHAR(300)	Details of note
CGID	INT	Caregiver identifier
ISERROR	CHAR(1)	Error or not
TEXT	TEXT	The note text

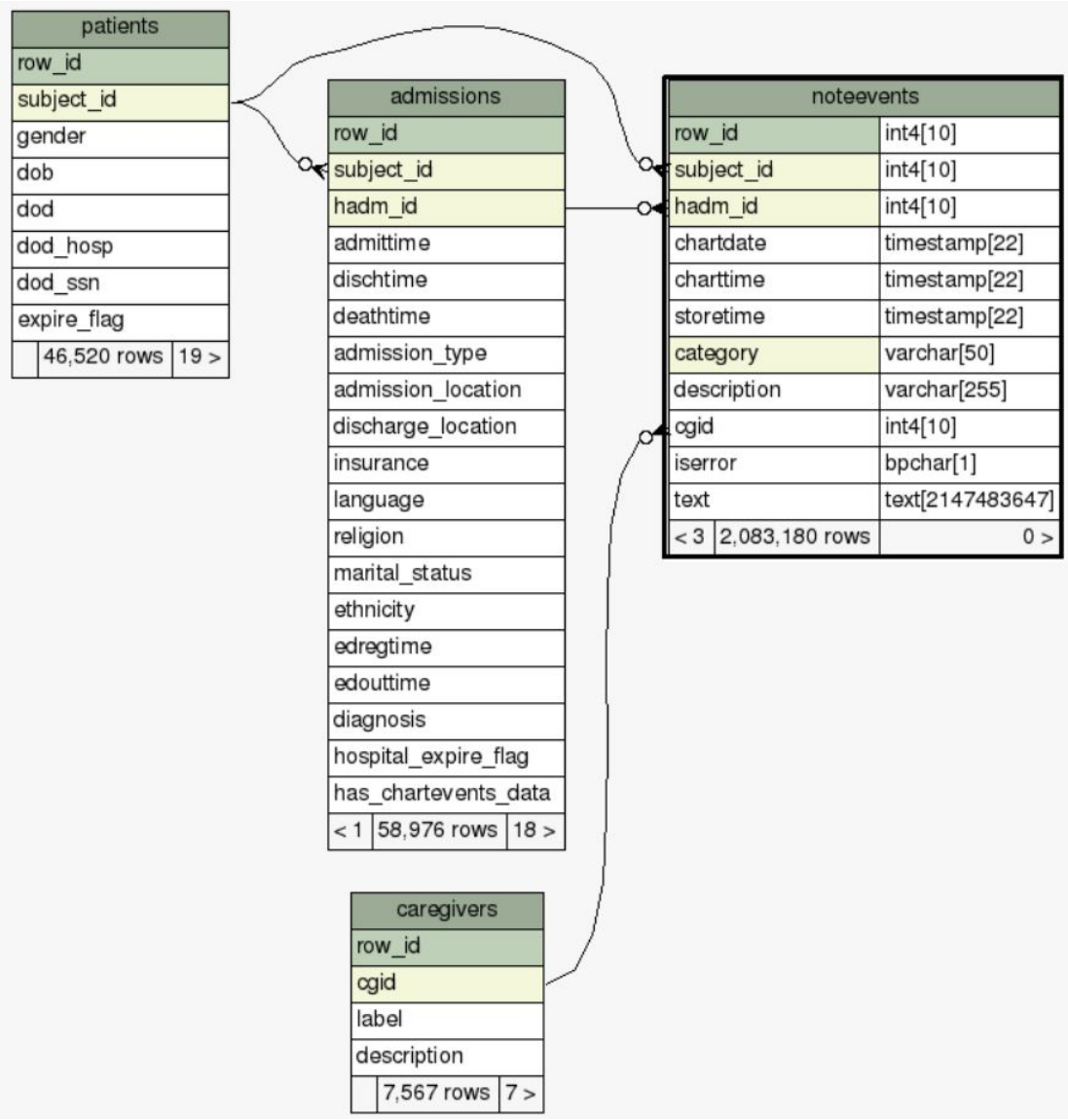

Fig. 1. NOTEVENTS entity relationship

A. *Exploratory Analysis*

We do an initial analysis of the data to see different trends that can be studied. We can see in figure 2 that there are most entries for category ”Nursing/Other”. There are 59652 available ”Discharge Summary” records. In figure 3 we can see the variation in number of discharge summaries across time. In figure 4 we present a histogram of number of discharge summary per day. Figure 5 shows a density distribution of discharge summaries per day. Finally Figure 6 shows different description for discharge summaries available in the dataset.

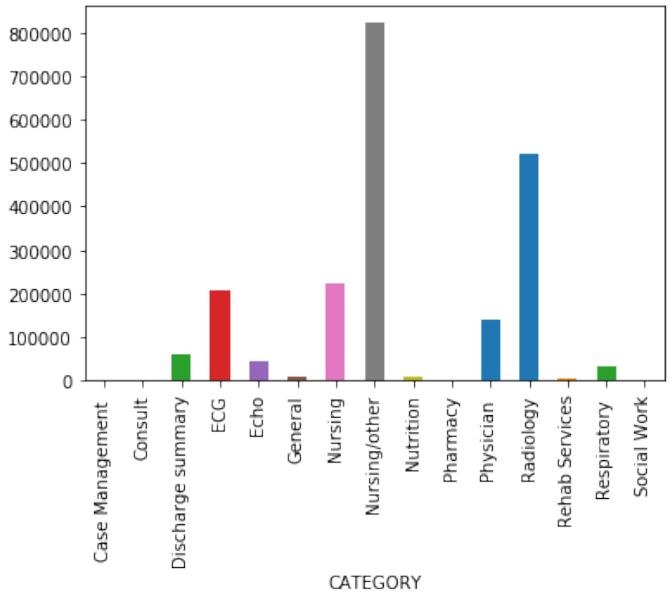

Fig. 2. Category Distribution in MIMIC-III

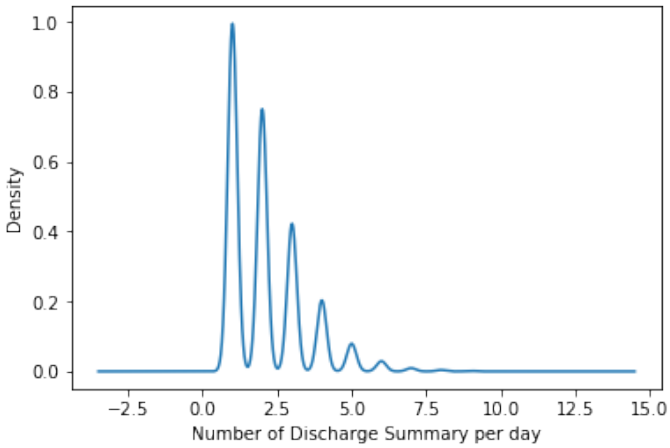

Fig. 5. Distribution of Discharge Summary per day

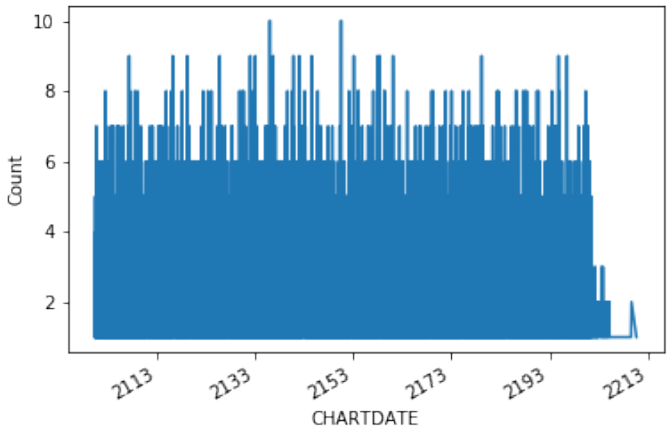

Fig. 3. Time-series variation of Discharge Summary

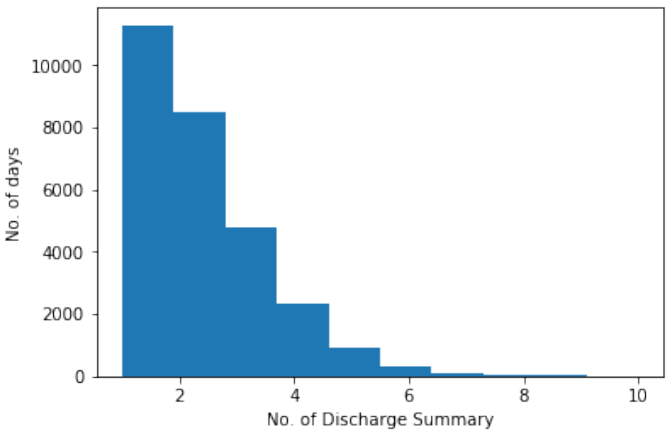

Fig. 4. Histogram of Discharge Summary

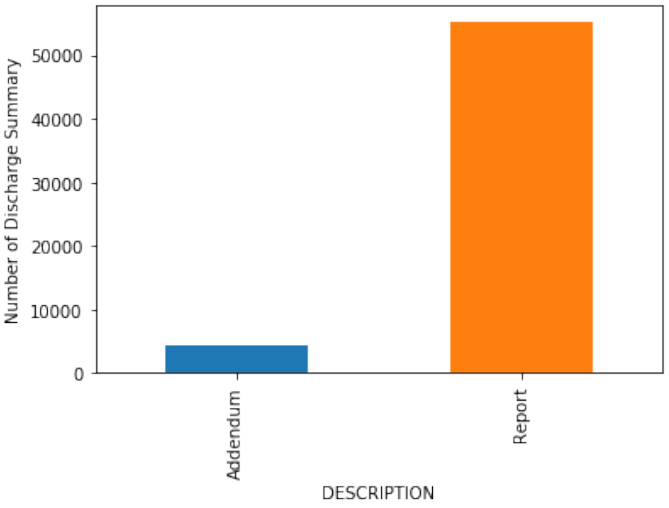

Fig. 6. Plot of Descriptions for Discharge Summary

IV. Feature Engineering

This section highlights the techniques used to design suitable features to be used for Topic Modeling. We use different feature engineering techniques suitable for different models. We start with pre-processing of the raw data. This is followed by Dimensionality Reductions techniques. We then present the specific feature engineering for the models that we used in the sections below.

A. *Preprocessing*

In this we use pre-processing mainly for cleaning, transformation, normalization, & feature extraction etc. The data that we used from MIMIC-III was mostly free text discharge summary notes and contained raw text that was not structured. The biggest challenge was to extract information from this raw text and convert it to structured form for further use. We made use of regular expressions for initial processing like extraction of lines one at a time, removal of special characters and symbols etc. We then make use of different Natural Language Processing Libraries like **ntlk** and nlp libraries within sklearn to further process the data.[5], [4]

Preprocessing was done on the raw dataset to make it suitable for feature extraction. Following is the overview of different pre-processing techniques used:

- Number removal - Removing numeric data from feature text.
- Special Symbols removal - Removing non alphabetic data from feature text.
- Removal of unwanted spaces Removing spaces from feature text.
- Stop words removal - Removing filler words like I, You, is etc.
- Case agnostic processing - Converting all text to same case (upper or lower).
- Stemming - Reducing words to their word stem, base or root form.
- Lemmatization - Reducing the words to its root form in English.

B. *N-gram embeddings*

Words that tend to appear in similar contexts are likely to be related. A N-gram is a sequence of N words. We use the information of the sequence to create embeddings. We make use of the context words to create this embedding. We create a *vocabulary* V, consisting of the most commonly-occurring words. We create another shorter list C of at most C_n of the most commonly-occurring words, which we shall call context words henceforth. For each word $w \epsilon V$, and each occurrence of it in the text stream, we look at the surrounding window of N words. Let say that $N = 2$ then we see words (two before, two after):

$$w1, w2, w3, w4$$

We keep count of how often context words from C appear in these positions around word w. That is, for $w \epsilon V$; $c \epsilon C$, we define

$$n(w, c) = \#\text{times } c \text{ occurs in a window around } w$$

Using these counts, we construct the probability distribution $Pr(c|w)$ of context words around w (for each $w \epsilon V$), as well as the overall distribution $Pr(c)$ of context words. We then represent each vocabulary item w by a $|C|$-dimensional vector $\Phi(w)$, whose c'th coordinate is:

$$\Phi_c(w) = max(0, log \frac{Pr(c|w)}{Pr(c)})$$

$\Phi(w)$ is the embedding that we use. We experimented with different vocabulary and window sizes and also made used of different dimensionality reduction techniques like Principal Component Analysis to obtain varying embedding sizes and experiment with the same. This finally give us an embedding representation of a word in terms of context words.

V. System Architecture

We present a Natural Language Processing system that makes use of topic modeling and word embeddings to create a **chatbot** that would help answering patient's query based on their personal *discharge summary*, and the knowledge representation learnt from the entire corpus of different discharge summaries. We explain each part of the system in more detail later in this section.

A. *Topic Modeling*

Topic models are also known as probabilistic topic models. It basically refers to statistical algorithms for discovering the latent semantic structures of a text body. A topic contains a cluster of words that frequently occurs together. A topic modeling can connect words with similar meanings and distinguish between uses of words with multiple meanings [9]. In Natural Language Processing, statistical topic modeling is used to discovering the abstract "topics" that occur in a collection of documents. Topic modeling is a frequently used text-mining tool for analysis & discovery of hidden semantic structures in a body of free text. For example, given that a document is about a particular topic, one would expect particular words to appear in the document more or less frequently. Say "dog" and "bone" will appear more often in documents about dogs, "cat" and "meow" will appear in documents about cats, and "the" and "is" will appear equally in both.

The "topics" produced by topic modeling techniques are clusters of similar words. A topic model captures this intuition in a mathematical framework, which allows examining a set of documents and discovering, based on the statistics of the words in each, what the topics might be and what each document's balance of topics is. In the age of information, the amount of the written material we encounter each day is simply beyond our processing capacity. Topic models can help to organize and offer insights for us to understand large collections of unstructured text bodies.

B. *Latent Dirichlet allocation*

latent Dirichlet allocation (LDA) is a generative statistical model. It allows sets of observations to be explained by unobserved groups, that explain why some parts of the data are similar. For example, if observations are words collected into documents, it posits that each document is a mixture of a small number of topics and that each word's creation is attributable to one of the document's topics. LDA is a three-level hierarchical Bayesian model, in which each item of a collection is modeled as a finite mixture over an underlying set of topics. Each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities. In the context of text modeling, the topic probabilities provide an explicit representation of a document [7].

LDA uses two probability values: $P(word|topics)$ and $P(topics|documents)$. These values are calculated based on an initial random assignment, after which they are repeated for each word in each document, to decide their topic assignment. In an iterative procedure, these probabilities are calculated multiple times, until the convergence of the algorithm. We have used this topic modeling technique for topic extraction in the chatbot presented in the paper.

C. *Non-Negative Matrix Factorization*

Non-negative Matrix Factorization is a Linear-algebraic model, that factors high-dimensional vectors into a low-dimensionality representation. Similar to dimensionality reduction technique Principal Component Analysis (PCA), NMF takes advantage of the fact that the vectors are non-negative. By factoring them into the lower-dimensional form, NMF forces the coefficients to also be non-negative [10].

Given the original matrix A, we can obtain two matrices W and H, such that $A = WH$. NMF has an inherent clustering property, such that W and H represent the following information about A:

- A (Document-word matrix)ŁŁinput that contains which words appear in which documents.
- W (Basis vectors)ŁŁthe topics (clusters) discovered from the documents.
- H (Coefficient matrix)ŁŁthe membership weights for the topics in each document.

We the l2-norm of difference between A and WH as the objective function and optimize it to get an accurate representation of the topics.

D. *Discharge Summary Chatbot*

We make use Natural Language Processing to build a basic chatbot that answers a patientś question relevant to their discharge summaries. The system leverages individual "Discharge summary" of a patient to answer specific questions based on topic modeling.

We have defined three types of question that the chatbot handles:

1) *Instruction Question* - These are special defined commands or instructions that are used to invoke specific functionality. For example, writing "summary" to chatbot displays patient entire discharge summary on the screen.

2) *Extraction Question* - These are the questions whose answers can be directly extracted from the discharge summaries. For example Date of birth, Admission Date, Discharge Date, Gender, Services rendered, Allergies etc. The chatbot leverages the structure of the discharge Summaries to answer these question. Analysis of the discharge summaries

revealed that most of these questions were directly mentioned in a specified format in the notes. We extract these using regular expression and return them directly as an answer.

3) *Direct topic Question* - For this type of questions we try finding the topic of the question in the discharge summary document. If an exact topic match is found in the document, we return that particular line as the answer. We create a topic map for each selected discharge summary to efficiently answer these types of question in real time.

4) *Indirect topic Question* - If a question cannot be answered using the above two strategies then we resort to this technique. Here we look for similarity between the question topic, and all other topics present in the entire corpus (all the discharge summaries for all the patients available). We use cosine similarity to measure similarity between the topics. We then select all the similar topics that has a cosine similarity score of higher than a threshold (say 0.9). We then see if one of the similar topics appear exactly in the patient's discharge summary. If a match is found then that particular line is returned as the answer, otherwise system response with a *don't know* response.

As the system is initialized, we process entire corpus of discharge summary text and extract topics using topic modeling techniques described above, from each of them to get a large collection of topics. We then create embedding for each of these topics and save it as *Topic Embedding Map*. We apply dimensionality reduction technique before saving the embedding to the map for getting a compact representation. We then as user to select discharge summary of interest based on *Patient Id*. We read this discharge summary and split it into sentences. We extract topic using topic modeling techniques described above for each sentence and save a mapping of topic to sentences as *Topic Map*.

For any question we process the raw text as is described in section IV-A. We then use Latent Dirichlet Allocation (LDA) for topic modeling of question. This gives us question topic which we call Q_{Topic} here. We then answer the question in the order in which it is defined above. Firstly, it is checked whether the question is a *Instruction Question* and processed if found, otherwise it is checked whether the question is a *Extraction Question* and processed if found, otherwise it is checked if the question is a *Direct topic Question* and processed, and finally if no match is found then the question is treated as a *Indirect topic Question*.

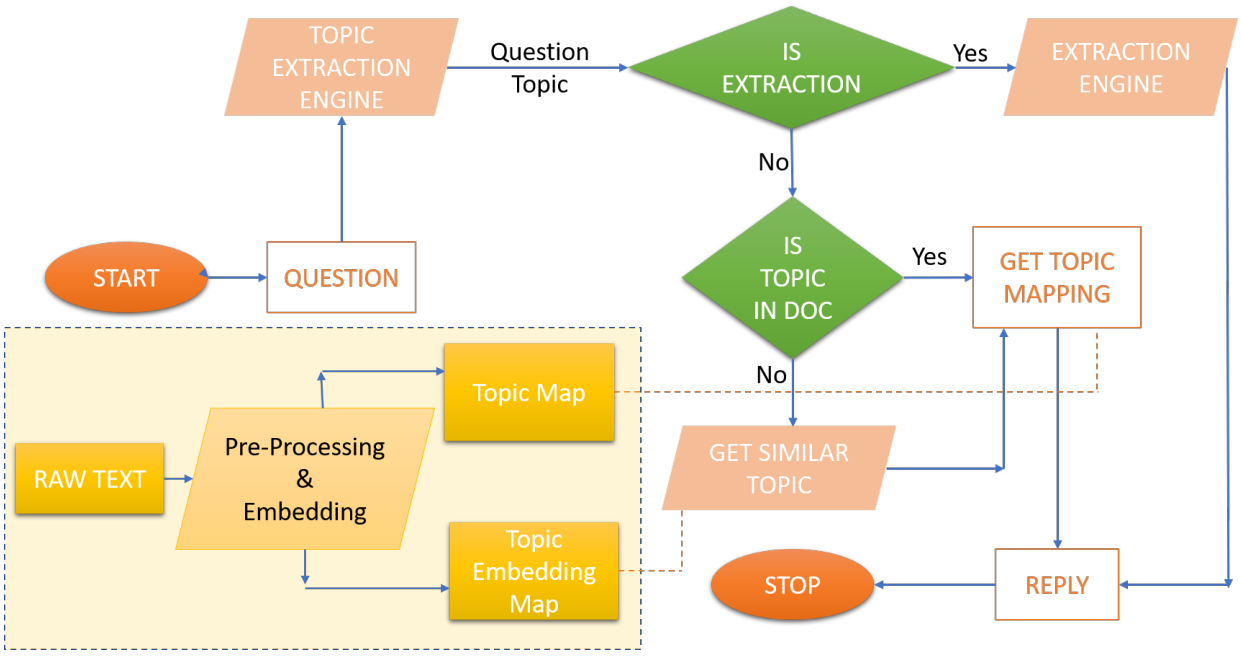

Fig. 7. Chatbot Architecture

There are two levels of topic modeling that we use:

1) *Document Topic Modeling* - Here we extract topic from each line of text in a single discharge summary document specific to a patient and store the (topic, line) map henceforth known as *Topic Map*.

2) *Corpus Topic Modeling* - Here we extract topic from each line of text for every discharge summary and then create an embedding for each of the topics as described in IV-B. We store this topic embedding map hence forth known as *Topic Embedding Map*.

Following is a summary of control flow of the chatbot:

- Topic is extracted from the question using LDA.
- If the topic is one which requires information extraction like Date of Birth, Sex etc. we use regular expression to extract the information and use it as an answer.
- Otherwise if the topic is present in the *Topic Map* described earlier, we use the corresponding mapping value to answer the question.
- Otherwise we generate topic embedding from the entire corpus and find the most similar topics in terms of embeddings from the *Topic Embedding Map*. We use cosine similarity score for deciding the level of similarity. We then check if the similar topics are present in the *Topic Map*. If found then we return the mapping value as answer.
- If none of the above steps are able to return an answer then the chatbot replies with a "Cannot find" response.

VI. Evaluation Metrics

Traditionally there aren't specific evaluation metrics like precision or recall for analyzing a chatbot system. So we define the following criteria for evaluating the performance:

- Functionality - Describe the different possible uses of the system.
- Scalability - Describe how the system scales as the data grows.
- Speed - Describe how fast is the system in responding to user queries.
- Interoperability - Describe how can the system interact with other systems.

VII. Experiments & Results

We basically experimented with two different topic modeling techniques as described in the section V-A. We tried different combinations of pre-processing and different window sizes for the generated embedding as is described in section in section IV above. We made use of techniques like PCA to produce compact representation for the N-gram words embeddings.

Preprocessing turned to be a huge task as the discharge summaries were not structured text. We had to try different heuristic for different tasks based on the functionality that we wanted chatbot to provide. We ran several experiments with chatbot and improved its functionality based on the output. We tried stemming of topics so that similar base word topics group together. We also tried different ways to split the sentences in the text. We tried splitting on new line character, and on fullstops as well. Different experiments were with different topic extraction algorithms like Latent Dirichlet allocation or Non-Negative Matrix Factorization. We also tried different regular expressions based on the patterns identified in the text for answering specific questions. Also as the context words were used for embeddings, their removal and inclusion changes result.

We present an analysis of our chatbot on the metrics presented in section VI:

- **Functionality** - The chatbot presented in the paper serves the purpose of answering questions based on a patient's discharge summaries. In a way it helps naive patient make sense of their discharge summaries. It answers extraction type questions, which are structured details extracted directly from the discharge summaries. It also answers question based on exact topic match, and topic similarity. It has added functionality wherein we can provide instructions and chatbot does system specific tasks. For example, printing out the discharge summary.

- **Scalability** - The system is scalable, as the task of embedding creation, topic map creation, and getting similarity can easily be parallelized on different machines. All the data does not need to be same machine, and we can utilize the map-reduce paradigm to achieve parallelism in the task of embedding creation, similarity comparison and topic extraction.

- **Speed** - Apart from the initial startup time when the chatbot initializes by loading the data and creating embeddings and topic map, the system works pretty much in real time. This is because of the fact that system uses regular expressions for extraction which are pretty fast even for long documents. Also since we create and save *Topic Map* as a hash map, the lookup for topic is almost constant time and hence fast. The only time taking operation is finding similar topics, which is optimized to run faster using matrix operations which are highly optimized in python libraries.

- **Interoperability** - Currently the system we presented does not share data or interact with other systems, but it can easily be extended to do so using python function calls or rest API. But this third party operations will be a time costly operation. An example of this can be integration with disease identification engine. The chatbot can send discharge summary text and the disease identification engine can respond with the list of words that are possible diseases mentioned in the discharge summary.

Following are some of the answers generated by a running demo of our chatbot system:
Bot: How can I help ?
Person:What is my date of birth?
Bot: Date of Birth: [**2109-10-8**]
Bot: How can I help ?
Person:When was I discharged?
Bot: Discharge Date: [**2172-3-8**]
Bot: How can I help ?
Person:What is my gender?
Bot: Sex: F
Bot: How can I help ?
Person:What are the services I had?
Bot: Service: NEUROSURGERY
Bot: How can I help ?
Person:Am I married?
Bot: ['Social History She is married']
Bot: How can I help ?
Person:How was my MRI?
Bot: ['Pertinent Results MRI Right middle cranial fossa mass likely represents a meningioma and is stable since MRI of']
Bot: How can I help ?
Person:How can I make an appointment?
Bot: ['This appointment can be made with the Nurse Practitioner', 'Please make this appointment by calling TelephoneFax']
Bot: How can I help ?
Person:Do I have sinus?
Bot: ['She was found to hve a right cavernous sinus and nasopharyngeal mass']
Bot: How can I help ?
Person:How should I take my medication?
Bot: ['If you are being sent home on steroid medication make sure you are taking a medication to protect your stomach Prilosec Protonix or Pepcid as these medications can cause stomach irritation', 'Pain or headache that is continually increasing or not relieved by pain medication']
Bot: How can I help ?
Person:How are my symptoms?
Bot: ['But her symptoms resolved with treatment for autoimmune disease']
Bot: How can I help ?
Person:Do I need to visit the clinic?
Bot: ['She will need to be scheduled for brain tumor clinic']
Bot: How can I help ?
Person:What to do if I have seizures?
Bot: ['CALL YOUR SURGEON IMMEDIATELY IF YOU EXPERIENCE ANY OF THE FOLLOWING New onset of tremors or seizures']
Bot: How can I help ?
Person:What medications should I take?
Bot: ['Medications on Admission bactrim famotidinefolic acid fosamax lorezapam methotrexate mvi orencia plaquenil prednisone qd Discharge Medications']

We can see above that the answers that the bot gives above pretty reasonable answer based on topic modeling, which can be deciphered by reading the text. The bot also gives good responses for extraction questions like getting date of birth or admit dates from the discharge summary. The preliminary results in this paper show that there is potential for a NLP chatbot to answer patient's question based on patient's discharge summaries.

VIII. Future Scope

We restricted our work to using a fixed window size for generation of embeddings. The technique can be extended to include a combination of different window size giving a mixture model [8] which can further improve the performance. We also generated the embedding based only on the distribution of context words around the target topic words. These embedding can be further improved and enriched by making use of other features that are available along with the free text information. We also have not handled negation question answer scenarios, or conversational aspect of the chatbot.

In terms of Human Computer Interaction chatbot component, the performance can be improved by adding the ability of having a normal conversation to chatbot in addition of what is being described and used in this project. Advanced topic modeling techniques can be experimented with. Additional features based on parse structure can be taken into account while deciding the topics and answers as well. Experiments with different pre-processing techniques can also be an interesting study. One of the interesting functionality additions can be adding a sentiment analysis component along with enhancing its interoperability with different systems. For example in one of the talks in the class an interesting ideas was to ask chatbot to send an email to the patient based on response of the patients. But given the time restriction we made an attempt to provide a working example of the techniques described in the paper to enhance the quality and experience of healthcare services. Nonetheless, the suggestions made in this section can be an avenue for interesting future experiments.

IX. Conclusion

The current work demonstrates that NLP techniques along with HCI inspired designed chatbots can be a huge asset in the field of healthcare. As is the case of our study, we have shown that a chatbot can be used to answer user queries regarding a patients provider notes. For example, a user might query the Chat-bot for admission date or attending on their case. The Chat-bot would make use of techniques described in section above to find the answer from the document for a user and return the result.

This is particularly helpful as the patients usually have simple questions about their medical condition, and the discharge summaries contain huge amount of data and details that is usually not comprehensible to a common person. NLP based chatbots as one described in this paper can become really helpful in such scenario. For example, a patient might just want to know that when does he need to visit doctor next or what are the medication he should take or what are the precautions that he should take. All these questions can in theory be efficiently answered by a chatbot trained on the patient's discharge summary. We are eager to experiment with different scopes that we presented in section VIII to see how our preliminary results evolve by trying different variant ideas.

Using Natural Language Processing and artificial Intelligence in the area of healthcare has the potential to provide great benefits for patients and system should be developed around the idea making use of the tremendous data repository that is available. There are several advantages of using a conversational agent in healthcare, including cost reduction, improving efficiency and reduction of time spent asking questions to make the right diagnosis. Our paper presents a simple conversation agent chatbot that attempts at doing the same.

References

[1] S. Leonard Syme, *The prevention of disease and promotion of health: the need for a new approach*, European Journal of Public Health, Volume 17, Issue 4, 1 August 2007, Pages 329330.

[2] Douglas G Manuel, Laura C Rosella, Therese A Stukel *Importance of accurately identifying disease in studies using electronic health records*, BMJ 2010;341:c4226

[3] Pere Ponsa, Daniel Guasch, *A humancomputer interaction approach for healthcare*, Universal Access in the Information Society, March 2018, Volume 17, Issue 1, pp 13

[4] Pollard, T. J. & Johnson, A. E. W. *The MIMIC-III Clinical Database* http://dx.doi.org/10.13026/C2XW26 (2016).

[5] Lal,Harsh & Pahwa,Gaurav *Root cause analysis of software bugs using machine learning techniques* 2017 7th International Conference on Cloud Computing, Data Science & Engineering - Confluence

[6] Daniel Jurafsky & James H. Martin *Speech and Language Processing*. Copyright 2016.

[7] David M. Blei, Andrew Y. Ng, Michael I. Jordan *Latent Dirichlet Allocation*, Journal of Machine Learning Research 3 (2003) 993-1022

[8] Hasim Sak, Cyril Allauzen, Kaisuke Nakajima, Francoise Beaufays, *MIXTURE OF MIXTURE N-GRAM LANGUAGE MODELS*, google

[9] Rubayyi Alghamdi, Khalid Alfalqi, *A Survey of Topic Modeling in Text Mining*, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 6, No. 1, 2015

[10] Da Kuang and Jaegul Choo and Haesun Park, *Nonnegative matrix factorization for interactive topic modeling and document clustering*

[11] Benilda Eleonor V. Comendador, Bien Michael B. Francisco, Jefferson S. Medenilla, Sharleen Mae T. Nacion, and Timothy Bryle E. Serac, *Pharmabot: A Pediatric Generic Medicine Consultant Chatbot*, Journal of Automation and Control Engineering Vol. 3, No. 2, April 2015

[12] Gillian Cameron, David Cameron, Gavin Megaw, Raymond Bond, Maurice Mulvenna, Siobhan O'Neill, Cherie Armour, Michael McTear - *Towards a chatbot for digital counselling*. Proceeding HCI '17 Proceedings of the 31st British Computer Society Human Computer Interaction Conference, Article No. 24

[13] Nicole Radziwill and Morgan Benton, *Evaluating Quality of Chatbots and Intelligent Conversational Agents*, arXiv preprint arXiv:1704.04579, 2017 - arxiv.org

[14] Cognizant, *Chatbot Imperative: Intelligence, Personalization and Utilitarian Design*, https://www.cognizant.com/whitepapers/the-chatbot-imperative-intelligence-personalization-and-utilitarian-design-codex2469.pdf

[15] Nimavat, Ketakee and Champaneria, Tushar, *Chatbots: An overview. Types, Architecture, Tools and Future Possibilities*, 2017, IJSRD - International Journal for Scientific Research & Development— Vol. 5, Issue 07, 2017 — ISSN (online): 2321-0613

[16] Ellis Pratt, *Artificial Intelligence and Chatbots in Technical Communication A Primer*

[17] Leaman, R., Khare, R., & Lu, Z. (2015). *Challenges in clinical natural language processing for automated disorder normalization*. Journal of biomedical informatics, 57, 28-37.

[18] H. Kopka and P. W. Daly, *A Guide to LaTeX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.