IMPLEMENTING A RESPONSE SYSTEM FOR PARTICULAR DOMAIN USING MACHINE LEARNING

Mohamed Najeem Fazil[1], Saira Banu[2]
Department of Computer Science and Engineering,
[1]Student, [2]Assistant Professor (Sr. Grade),
Crescent Institute of Science and Technology, Chennai.

Abstract
This paper describes about the response system that takes place within particular domain. It is used to concern with building systems that automatically responses for the questions made by humans in a natural language. It is a computer science discipline that works under fields of information retrieval and language processing. Here the queries and response are completely based on closed domain. The system uses the combined techniques of machine learning and natural language processing. The system aims to provide conversation between both human and machine. In the existing model, system used to work based on the intents classifiers. It is used to provide the response for the intents and the responses can be sent only by the manual selection. The proposed concept works based on pattern matching technique and makes an automated response to question . Here the user query is considered first and tokenized to match with the patterns. The response can be generated by matching the input sentence from user . Here database used as knowledge storage and interpreter has been selected as stored programs of function .The system uses a database of responses, that can be selected from a collection of text provided for a particular domain. This system improves the efficiency of customer based service for the closed domain.

KEYWORDS: Automated Response, Natural Language, Machine Learning, Closed Domain

1 . INTRODUCTION

An automatic response system is a pre-defined reply that can be generated by a software program for incoming messages. For example, a user may set up an automatic reply for incoming e-mail which is used to make the sender to know their e-mail was received. Automated response system builds its answers by querying a structured database. There are two types of domains related to queries namely Closed Domain and Open Domain. Closed-domain answering deals with questions for a specific domain. Open-domain question deals with questions about everything, and it is completely based on world knowledge. Many approaches employ natural language processing technology to understand questions given in natural language text, which is incomplete and error-free. In addition, instead of getting exact answer, many approaches used to return hyperlinks to documents containing the answers, which is inconvenient for the students or learners. Here we develop technique to identify the type of a question, based on which the proper technique for extracting the answer can be done. Automated response system is able to translate and interpret human natural language input. This is done through a combination of NLP (Natural Language Processing) and Machine Learning. In the simpler response system, any response (provided it was correct grammar beforehand)has no value because of any grammatical error. This happens due to the already existing sentences in the storage. It might however be unable to handle any input it does not recognize because of human grammatical errors or not matching sentences. The newer smarter response system are the exact opposite, if they are well “trained” they can recognize the human natural language and can response accordingly to any queries. The rest of this paper proceeds as follows. Section II reviews the literature related to this paper. Section III presents the basic model of the response system. In Section IV, the application of machine learning techniques and the response generation for queries can be studied .

2 . RELATED WORKS

[1] proposed the approach for creating open domain, conversational systems based on large dialogue corpora with the help of generative models. This models generates responses word-by-word for flexible interactions. Based on this, they proposed hierarchical recurrent encoder-decoder neural network to the dialogue domain. It is also proved that this model is competitive with back off n-gram models and state-of-the-art neural language models. Here investigated the limitations of this and same kind of approaches, and showed how to improve the performance by bootstrapping the learning from a larger question answer pair corpus and from pre trained word embeddings. They contributed in terms of the direction of end-to-end trainable, non-goal-driven systems based on generative probabilistic models. Here the generative dialogue problem is defined for modeling the utterances and interactive structure of the dialogue. [2] proposed a work in Question Answering that is used to focus on web-based systems that extract answers using lexicosyntactic patterns. It presented an another approach in which patterns are used to select highly precise relational information offline, that is used to efficiently answer questions. Here evaluated the idea on a challenging subset of questions, against a state of the art web-based Question Answering system. The result showed that the extracted relations answers 25% more questions in a correct manner. Here proposed a idea in which information is extracted automatically from electronic texts offline, and stored for quick and easy access. [3] proposed an end-to-end method for automatically generating short email responses, called Smart Reply. It generated semantically different kind of suggestions that can be used as complete email responses with just one click on mobile. The normal task of the Smart Reply system was to select the most likely response for an original message. Here new model is introduced for semantic clustering of content. It requires only a few amount of explicitly labelled data. [4] proposed a simple approach for this task which uses the recently proposed sequence to sequence framework. This model communicates by predicting the next sentence from previous sentence or sentences in a conversation. The advantage of this model is that it can be trained end-to-end and generates simple conversations from a large conversational training dataset. The preliminary results suggested that, despite making the incorrect objective function, the model is able to converse in good manner. This model obtain knowledge from both a domain specific dataset and general domain dataset like movies, weathers etc. For a domain-specific support dataset, the model can identify a solution to a technical problem via conversations. But here we find that the lack of consistency is a common failure mode of our model. [5] presented a machine learning algorithm to classify the question. It defined a hierarchical classifier that is guided by a layered semantic hierarchy of answer types, and eventually classifies questions into fine grained classes. It showed accurate results on a huge collection of free-form questions. It suggested that it is helpful to consider this type of classification task as a multi-label classification and found that it is possible to achieve good classification results despite the fact that the number of different labels used is fairly large.

3. PROBLEM STATEMENT

The purpose of a response system is to make a human machine conversation. The automated response system architecture integrates a language model and performs computational algorithms to make communication between a user and a computer using natural language.. This uses artificial intelligence technology to interact between men and machines using natural language possible. Based on pattern matching technique, the improvement can be done by including the intents for each query and generation of automated response in closed domain.

4. SOLUTION APPROACH

In the response system, NLP software does not search for keywords in text as it does in search engine. . It means the system response has been programmed to identify certain things people want from it, and act upon those queries. It works based on knowledge of sentence structure, tokenization, and machine-learned pattern recognition to match with what is used as an “intent” which has been classified. It uses mapping process to transform ontologies and knowledge into structure of database and then use that knowledge to work with the chats. The proposed model avoids several disadvantages that includes: the necessary to learn and use particular language such as AIML, and the use of non-matured technology. Here it used to combine the user's query in natural language and based on the query it will generate the response to the user.

5. METHODOLGY:

5.1 SYSTEM MODEL

The response system is completely based on the pattern present in the query. Here the question can be asked in the user's native language and response can be generated in the same way. Here the input statement is processed by each of the logic adapters. Then the system select the known statement that closely matches the users input. And a predefined matches returned to the selected match and also provides confidence value based on matching.

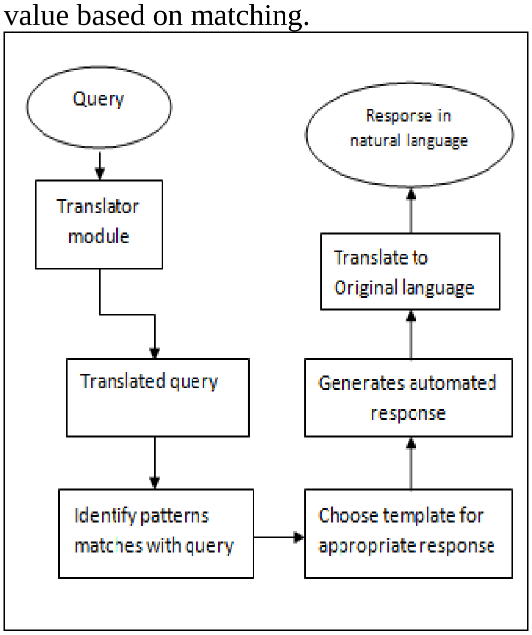


Fig. 1 Automated response system

Here the system simply works on the basis of a sequence of character-string-matches, and the particular character strings could be in any language. The functions / dialogue acts (DAs) are often domain specific. In other words, instead of asking whether the function of the user's choice is a question or answer, we ask whether the function is to, for example, find flights or cancel a reservation in a flight reservation program. It ignores structure of sentence, order, and syntax, and count the number of occurrences of each word. Here vector space model is used, in which stop words (e.g. a, the, etc.) are removed, and morphological variants (e.g. talk, talks, talked, etc.) go through a process called stemming and are stored as stemmed words for each classes (e.g. talk). In the response phase, with the help of rule-based system, the stemmed words will be matched against documents stored in the system's knowledge database to find the documents containing similar keywords. The bag of words model is easier because it does not require any syntax knowledge. At the same time it is not precise enough to solve more complex problems.

5.2 RESPONSE MODELS

Generative models are one of automated response, which is helpful to make system smarter. This approach is not widely used by developers, it is used in labs currently.

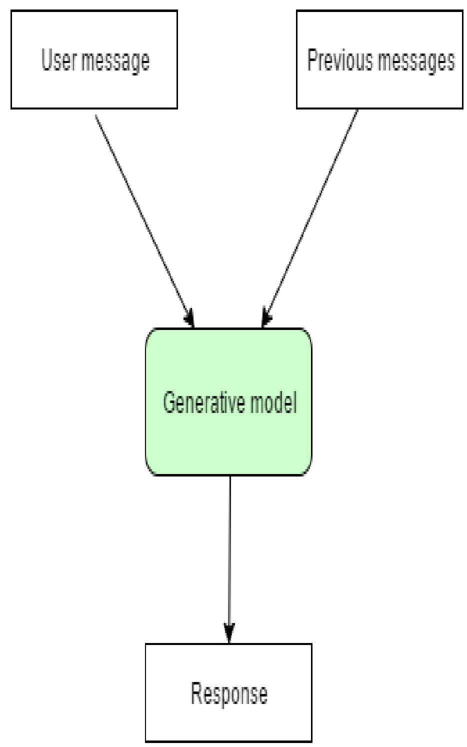


Fig. 2 Generative model

Retrieval-based models are much easier to build. They also provide more predictable results. They are used to provide at least all possible responses and ensure that there are no grammatically incorrect responses.

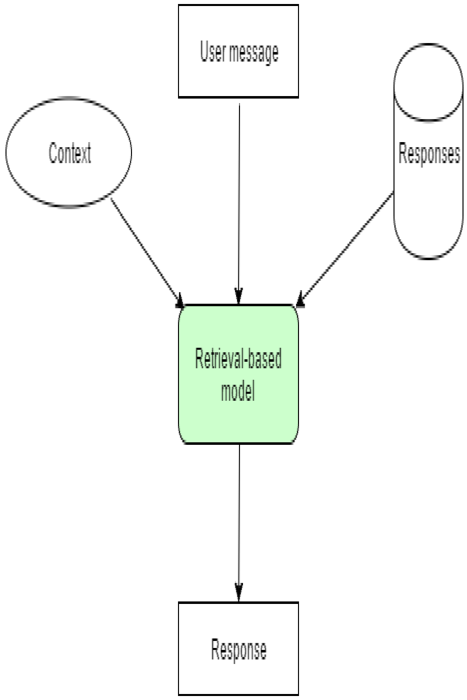


Fig 3 Retrieval model

The automated response system uses the message and context of conversation for choosing the best response from a list of predefined responses. The context can include current position with the bag of words approach.

5.3 RESPONSE GENERATION

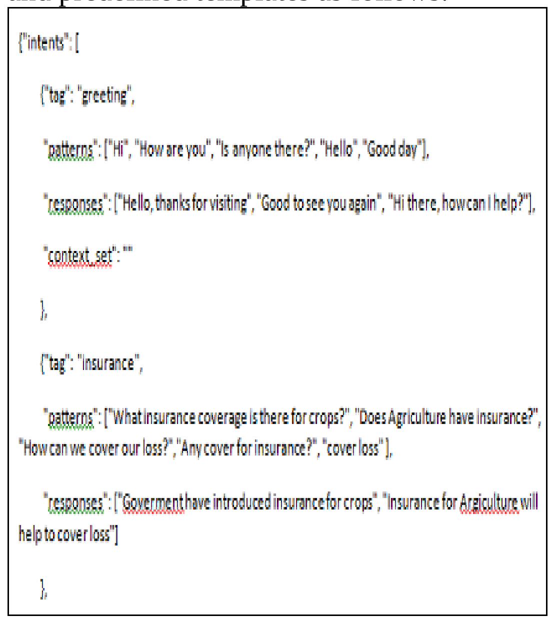
The automated response system can express the same kind of message using different words. Consider weather as an example, can say "It's going to rainy", or "Chance for rain is 80%" or " carry an umbrella today". Different users prefer different styles of response. The system can analyze previous chats and associated metrics to provide responses for the user. Intent classification module used to identify the intent of user message. Typically it is selection of one out of a number of predefined intents, though more sophisticated systems can identify multiple intents from one message. Classification of intents uses information present in context, such as previous intent message, and preferences.

The response generator can perform all the domain-specific calculations to process the request of user. It uses pattern matching algorithm with predefined templates, or even ask a human to help with response generation. The result of these calculations is a list of response templates. All these responses should be correct based on domain-specific logic.

The response generator uses both context of the conversation as well as intent and entities to support multi-message conversations.

6. RESULT

In automated response system, different types of intents (classes) are defined for the closed domain. Each intent consists a set of patterns and predefined templates as follows:



```
{"intents": [
  {"tag": "greeting",
   "patterns": ["Hi", "How are you", "Is anyone there?", "Hello", "Good day"],
   "responses": ["Hello, thanks for visiting", "Good to see you again", "Hi there, how can I help?"],
   "context_set": ""
  },
  {"tag": "insurance",
   "patterns": ["What insurance coverage is there for crops?", "Does Agriculture have insurance?", "How can we cover our loss?", "Any cover for insurance?", "cover loss"],
   "responses": ["Goverment have introduced insurance for crops", "Insurance for Argiculture will help to cover loss"]
  },
```

Fig 4 Defined Intents, Patterns and Responses

There are different number of intents associated for queries. Based on the intent, queries and responses are defined. In this system once the query is defined, the appropriate intent query for that query can be chosen. The intent match for every query can be defined in a probabilistic measure. The pattern keywords were matched with intents in the following manner.If different queries are similar to same keywords, then the matching intent for that query selected. Here the appropriate matching can be selected based on highest probability. In the following figure 6, there are two queries with word 'groundnut'. Based on patterns such as cost and soil, the appropriate response is selected.

Fig 5 Queries with same words

There are different number of intents defined. Each query can be made in different ways and the system used to give appropriate response using pattern matching technique.

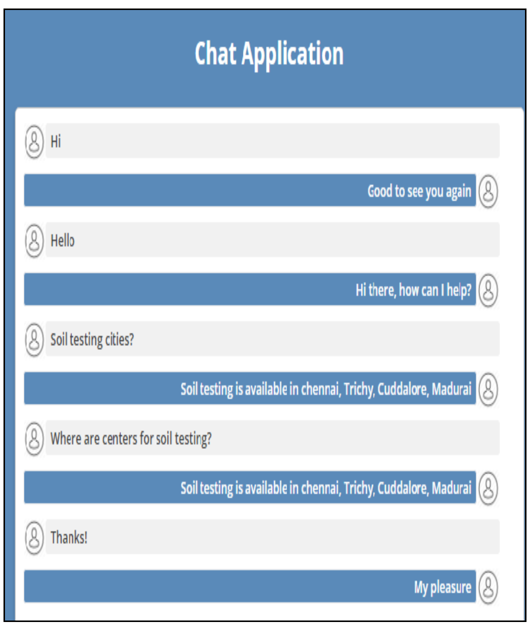


Fig 6 Pattern match with different queries

In the above fig 9, the query is related to the intent called soil testing. Here same type of query made in different ways and the appropriate response is received automatically. Using this system, many queries can be made corresponding to different intents. This can be improved by defining more number of patterns to match with the queries. Further improvement can be made by including large number of words for every queries to stemmed uniquely.

7. CONCLUSION

In this work we have generated the automated response system which works on the closed domain. It works on the combined technique of natural language processing and machine learning techniques. Here the response are generated based on the queries posed by humans by using pattern recognition algorithms.

1. This model is used to generate the response for the queries based on the intents that we have given.
2. The system can also accept queries in natural language and can be translated according to the language of patterns that we have provided.
3. Then the response can be generated based on the pattern matching and it can be translated to the natural language, so that user can view the response in their natural language for the closed domain.

   The advantages of the system are:
   - Easy to interact using a simpler interface
   - Improved efficiency with round the clock customer service
   - Easy to build and cost efficient

   The disadvantages of system are:
   - To build large set of data for queries.
   - Maintenance of the pattern to match with user related queries.
   - Shows only predefined templates as response.

8. FUTURE ENHANCEMENT

In future we can focus on the automated response system that process the natural language without the translator module. Based on text classification and pattern recognition algorithms, we can work to generate automatic responses for the open domains in multiple languages respectively.

REFERENCE

[1] I. Serban, A. Sordoni, Y. Bengio, A. Courville, and J. Pineau, "*Hierarchical neural network generative models for movie dialogues*", arXiv, 2015.

[2] Michael Fleischman, Eduard Hovy and Abdessamad Echihabi. 2003. "*Offline strategies for online question answering: answering questions before they are asked*", volume 1.

Proceedings of the 41st Annual Meeting on Association for Computational Linguistics.
[3] A. Kannan, K. Kurach, S. Ravi, T. Kaufmann, A. Tomkins, B. Miklos, G. Corrado, L. Lukacs, M. Ganea, P. Young et al.,"*Smart reply: Automated response suggestion for email*," KDD, 2016.
[4] O. Vinyals and Q. Le, "*A neural conversational model*", ICML Deep Learning Workshop, 2015. [5] Xin Li and Dan Roth. 2002. "*Learning question classifiers*", volume 1. Proceedings of the 19th international conference on Computational linguistics.
[6] S. Sukhbaatar, A. Szlam, J. Weston, and R. Fergus, "*Weakly supervised memory networks*", arXiv, 2015.
[7] J.Weston, A. Bordes, S. Chopra, A. M. Rush, A. Joulin, and T. Mikolov, "*Towards ai-complete question answering: A set of prerequisite toy tasks*", arXiv, 2015.
[8] J.Gratch, "*Architecture of a Framework for Generic Assisting Conversational Agents*", Intelligent Virtual Agents 2006, LNAI 4133, pp. 145 – 156, 2006.
[9] J. Bromley, J. W. Bentz, L. Bottou, I. Guyon, Y. LeCun, C. Moore, E. Sackinger, and R. Shah, "*Signature verification using a "siamese" time delay neural network*," IJPRAI, 1993.
[10] R. Lowe, N. Pow, I. Serban, and J. Pineau, "*The ubuntu dialogue corpus: A large dataset for research in unstructure multi-turn dialogue systems*," SIGDial, 2015.
[11] A.Graves, "*Supervised Sequence Labelling with Recurrent NeuralNetworks*", ser. Studies in Computational Intelligence. Springer, 2012.
[12] J. Wieting, M. Bansal, K. Gimpel, and K.Livescu, "*Towards universal paraphrastic sentence embeddings*," ICLR, 2015..