Chatbots: An Overview Types, Architecture, Tools and Future Possibilities

Ketakee Nimavat[1] Prof. Tushar Champaneria[2]
[1]UG student [2]Assistant Professor
[1,2]Department of Computer Engineering
[1,2]L. D. College of Engineering, India

Abstract— With social media penetration and internet connectivity poised to increase accompanied by advances in natural language processing and artificial intelligence, chatbots are expected to dominate the market. For a developer it is important to understand what the chatbot will offer and what category the chatbot falls into. This would help pick the algorithms or platforms and tools to use to build the bot. It also helps the end-users understand what to expect. Here we discuss the types of chatbots, the tools and algorithms that can be used for different types of chatbots and provide a general architecture that can be followed while building bots. We also address the areas where in chatbots are lacking and identify the research areas hence needing attention.

Key words: Chatbots, Conversational Agents, Recast.Ai, Wit.Ai, Classification of Chatbots, Chatbot Pipeline, Chatbot Architecture, Alexa

I. Introduction

Today, Alexa lets us control our lights with voice, Google assistant suggests places we would want to go when we are discussing dinner plans with our friends and Tesla can drive for us. Siri and Cortana live inside our phones and take commands. Screenless conversations are expected to dominate. From ELIZA [1]to Alice[2] to Alexa[3], we have come a long way.[4] Today, due to the penetration of social media and internet along with the progress in artificial intelligence, not only are bots coming up as a way to reach out to the users but also, they are making way for conversational interfaces such as Alexa to be omnipresent and go screenless. In the case of chatbots, with the launch of messenger bots and platforms such as slack, bots have gotten a boost. Facebook bots have grown from 34000 bots in November 2016[5] to 100,000 bots in April 2017[6] . As social media widens its penetration, companies will shift their focus to bots for reaching and serving users. This being because presence on a social media platform makes it easier for the user to access. It also means fewer resources to invest in for the company and the user. Users also do not have to deal with the hassle of downloading an app. They can just text the respective bot on their favored social media platform. It also means that the service is available 24/7. Had this been for a human employed customer service, it would cost way more to hire and train humans. It would mean having people in multiple shifts to provide 24/7 service. Bots are not yet as accurate and understanding as humans but they can get the basic job done and when they are stuck, a human can always take over[4]. Because of these benefits, bots are seen as a domain with huge potential for customer outreach and driving sales.[7]

However, most of the bots today are rule based bots that give the user a menu and the user navigates through the menu like telephonic complaint booking systems but in text. Secondly, most of the bots are closed domain bots meaning they are focused on one particular task and are trained for that field only. Alexa, Siri are examples of open domain bots. However, bots for restaurant booking are closed domain bots. Closed domain bots can be both, powered by artificial intelligence or be rule based. While the users expect open domain bots that are intelligent in all aspects, the goal of a bot system isn't just that. The goal of a bot system remains to automate a service using a conversational interface that allows the user to access the service from the platforms they frequently visit. This goes to say one mustn't be dissuaded from building a rule based bot that indeed works for the reason that the bot isn't intelligent like a super human. That being said, we must pursue building intelligent systems since the more intelligent they are in terms of understanding the end user, more will they be useful.

In this paper, we discuss the types of bots and where each one is useful followed by general architecture of a conversational system. Next we discuss the platforms that can be used to build a bot and compare them for the tasks they perform. Followed by which we discuss the future possibilities for research in the area of text based conversations.

II. Types of Chatbots

It was discussed in the previous section that based on the knowledge a chatbot has or accesses, chatbots can be classified into two kinds: open domain and closed domain. Here, we see how chatbots can be classified using other parameters too such as level of interaction, and method of response generation.

- Knowledge Domain
- Service Provided
- Goal
- Response Generation Method

A. *Knowledge Domain*

Here, the chatbots are classified based on the knowledge they access or the amount of data they are trained upon[8]. The classes are as follows:

1) Open Domain

Such bots can talk about general topics and respond appropriately

2) Closed Domain

Such bots are focused on a particular knowledge domain and might fail to respond to other questions. For example, a restaurant booking bot won't tell you the name of the first black president of America. It might tell you a joke or respond to how is your day but it isn't expected to do so since its job is to book a table and give you information about the restaurant.

All rights reserved by www.ijsrd.com

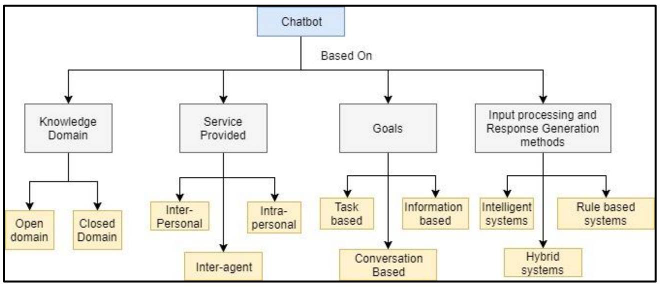

Fig. 1: Classification of Chatbots

B. *Service Provided*

Here, the bots are classified based on a proxemics like classification but instead of physical proximity, it is based on the sentimental proximity of the bot to the user, the amount of intimate interaction that takes place and hence is also dependent upon the task the bot is performing.

1) Interpersonal

Chatbots that lie in the domain of communication that belongs to the in the Social or Personal distance range in the Proxemics chart fall under this category. Bots that provide services such as Restaurant booking, Flight booking, FAQ bots etc. These chatbots are not supposed to be companions of the user, they are supposed to get information and pass them on to the user, they are just enablers. They can have a personality and can be friendly and will probably remember information about the user but they are not obliged or expected to do so.

2) Intrapersonal

These chatbots will exist within the personal domain of the user such as chat apps like messenger, slack and whatsapp and perform tasks that are lie in the personal domain of the user. Managing calendar, storing the user's opinion etc. They will be companions to the user and will understand the user like a human does. These might not be dominant in current scenario but as natural language understanding progresses they will soon emerge.

3) Inter-agent

Bots like this will be prevalent in IOT dominant areas. In this two systems communicate with each other to accomplish a task. As bots become omnipresent, all bots will require some inter-bot communication possibilities. The need for protocols for interbot communication will soon emerge. While a bot cannot be completely inter-agent bot, but it can be a service that handles other bots or handles communication making it easier for developers and users to integrate different services in the conversational ecosystem. The Alexa-Cortana integration is an example of inter-agent communication

C. *Goals*

Here the bots are classified based on the primary goal they aim to achieve.

1) Informative

These bots are designed to provide the user with information that is stored beforehand or is available from a fixed source. Usually, they are information retrieval algorithm based and would either fetch the result of a query form the database or would perform string matching. Most of the times, they will refer to a static source of information such a site's FAQ page or a warehouse database with inventory entry.

a) Example: FAQ bots.

– Tasks involved: String matching, named entity extraction, response generation

2) Chat based/Conversational

These bots talk to the user, like another human being. Their goal is to respond correctly to the sentence they've been given. Hence they are often built with the purpose of continuing conversation with the user based on techniques like cross questioning, evasion, and deference. Example: Siri, Alexa, mitsuku, Jenny, Tay, Xaoice[9]

Tasks involved and Algorithms: named entity extraction, information retrieval, string matching, relevance detection.

3) Task based

They perform one certain task such as booking a flight or helping you browse a store. Majority of the times, the actions needed in order to perform a task are predetermined, the flow of events including exceptions is decided as well. The bots are intelligent in the context of asking for information and understanding the user's input.

– Example: Restaurant booking bots, FAQ bots etc.

D. *Input Processing and Response generation method*

This classification takes into account the method of processing inputs and generating responses.[10]

Truly intelligent systems generate responses and use natural language understanding to understand the query. These systems are used when the domain is narrow and ample data is available to train a system.

Rule based systems use pattern matching and prove to be rigid. These can be used when the number of possible outcomes is fixed and scenarios are imaginable in number.

1) Hybrid

These systems are a mix of rules and machine learning. An example would be a system that uses a flow chart for managing the direction of conversation but provides responses that are generated using natural language processing.

Bots do not have to exclusively belong to one category or other. These categories exist in each bot in varying proportions. For example, all bots will require some kind of chat capabilities, perhaps a bot for a store will need to use information extraction when it comes to FAQs and search the site when it comes to giving product results. A service that the bot provides hence can include all three kinds of algorithms. Taking another example, a bot will be closed domain but will be programmed for small talk or a bot will be open domain and intrapersonal such as a productivity bot but it will also be extremely focused on conversations related to productivity.

This classification helps tell the user what to expect and build an image of the bot in the user's mind reducing the gap between user expectations and the delivered bot. Next we discuss the architecture of chatbots and various modules of the bot that enable the features of the bot.

III. PIPELINE & TOOLS

As seen above, multiple categories of chatbots are possible. A chatbot will be a combination of one or more categories. For example, it can be informative and rule based or it can

All rights reserved by *www.ijsrd.com*

be task based but uses generative techniques. However, all the chatbots follow a general flow or a general architecture, a pipeline. For each level in the general pipeline, for a certain category, the algorithms or methods pertaining to that category will be used. Even in the variety of chatbots that have won the Loebner prize, a general pipeline for the flow of information can be observed[11][12]. Below we discuss the general pipeline of a chatbot. A chatbot consists of four stages sequential or modules as shown in the Figure 2, the input is taken and processed into appropriate format. Next, named entities are extracted and intent is discovered. These are used to generate possible responses is generated or multiple responses are generated or selected and finally the most appropriate response is presented to the user.

A. Various tasks included in various stages are as follows[13][14][10][15]

Chunking, Sentence Boundary Detection, Sentence Parsing, Part of Speech Tagging are some of the basic input processing algorithms that always take place. As one uses libraries, these tasks get masked in functions. Next Named Entity Extraction, Intent detection, ambiguity detection, context detection, sentiment analysis, query classification, concepts and synonym detection etc. take place for understanding the input. [16][17].

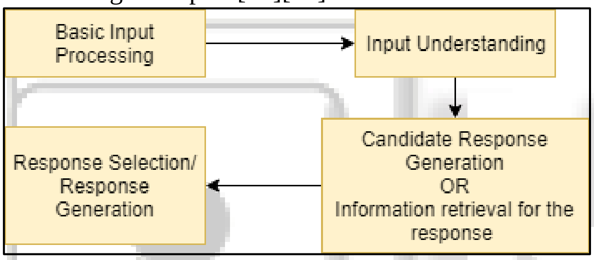

Fig. 2: Chatbot Pipeline

Once the query or input is structured into the desirable form, the response is generated. It can be generated through a predefined format or via machine learning. Summarization and simplification techniques can also be used if the response is fetched from a text source. Once there is a set of candidate responses, the responses can be checked for relevancy and the suitable response can be returned.

As seen in the above section, chatbots are of two kinds based on what action they perform in response generation:

1) Retrieval based models

The ones that are rule based or fetch responses from a predetermined set of responses. Retrieval Based models can be used when the data is limited and the conversation domain is constricted to a few conversation scenarios. Hence, if all the possible conversations for a use case with the bot are imaginable, a retrieval based model works well. Also, when the bot is not expected to display intelligence in all scenarios but can afford to deny knowing the answer, such retrieval based models work well. Booking systems, FAQ systems or any other systems that fetch information could work well even with a retrieval based bot.

2) Generative Models

These systems, such as in[18] which intends to maintain the hierarchical structure of underlying meanings of the language, can be used when a large amount of data is available and the system can be trained on that data. They often use NLP and NLU algorithms to process input and generate sentences. Artificial Neural Networks, LSTM [19][20][21], SVM[22][23], sequence models[24], Generative algorithms[25][26] etc. are some algorithms that are used in this model[27]. We do not yet have purely generative models[28], even the most advanced systems such as Alexa, Siri and Cortana are semi-rule based.

However, not all of this has to be done from scratch. Various tools and platforms are available that provide a level of extraction for performing the task. Next, we discuss various tools available that make it easier to make chatbots.

B. For Named Entity Extraction and Intent Detection

1) Recast.ai[29]

It allows the user to create custom queries and train the bot on them. Apart from providing machine learning services to extract entities and intent, it provides everything from templates to analytics for deployed bots. It provides hosting and bot integration across various platforms. One can also add external integrations and create custom logic in the bot. It has built in support for sentiment analysis and multiple languages. On top of allowing to adding custom queries, it also has a drag and drop interface for designing the conversation flow

2) Wit.ai[30]

Wit.ai helps classify the intent and extract named entities from a given input. It then returns the labels to the app or bot. It allows one to extract information such as location, time, date, weather and also allows one to create one's own intents. It parses the user input into structured data and also helps in conversation by sending the text. However, for performing actions on the structured data, the output of wit.ai must be sent to the app or server whose response wit.ai then returns to the conversation platform such as messenger or Kik.

3) API.ai[31]

It helps extract intents and entities in the query and train the bot on custom intents and entities. Like Wit.ai, it provides only the text processing service and not the bot hosting or external logic addition services.

C. For generating story flow based and Rule based chatbots

1) Chatfuel[32]

Chatfuel provides a drag and drop interface for making a rule based bot. Its artificial intelligence module lets you train the bot to map input statements to output but in comparison to recast.ai and other modules, it is quite rigid in terms of conversation flows. It also allows response prompts and integration with services such as email and IFTTT. It also allows a json integration for accommodating custom logic into the bot. The most attractive point of the service is that is extremely simple to build a rule based bot which is suitable for small businesses. It also allows the user to opt in or out for mailing lists and can send broadcast messages. It only need integration with the facebook page and does not need any other setups.

All rights reserved by *www.ijsrd.com*

2) Pandora bot

Pandora bots provides Artificial intelligence as a service along with a platform to build bots. It also makes custom bots on a paid basis. It uses AIML and provides an IDE for building bots as well as AIaaS API. It allows coding in AIML where in the developer is supposed to mention the input out responses as a framework. The model can then be trained on data and used.

3) Text Processing Platforms

Apart from that, various Cloud platforms such as amazon, IBM, Google, Microsoft have released their own conversational APIs that allow developers to analyze data, convert speech to text and vice versa and generate appropriate responses even. The benefit of using these platforms lies in the fact that it is easier to integrate them with an already existing cloud infrastructure.

These models are built upon massive amounts of data and employ powerful deep learning capabilities giving these platforms the kind of accuracy that a developer can't achieve with simple resources. Example: Microsoft bot framework, LUIS, Amazon Lex and Poly, IBM Watson, Google Speech API, Google Natural Language API.

4) For training on your own data

One can also build their prototype models from scratch and test them on the dataset. They can be trained using platforms such as Tensorflow and Weka. The model can then be scaled using either one's own server or the machine learning services provided by platform such as Amazon, Google and the others mentioned above.

IV. FUTURE POSSIBILITIES

Looking at the current explosion of chatbots, one might think that the chatbots have succeeded and we have agents that converse like human beings. Yet using general chatbots turns out to be disappointing since they do not do what the users expect them to do. However, on the other hand, narrow ones do very well. The reason why chatbots are disappointing are because our expectations are hyped about what they can do. We expect the agents to be able to respond to anything without understanding that the agent's job is to respond to relevant queries. Humans wouldn't be any better either if you asked a salesman what is the height of the tallest mountain. We do aim to make artificial general intelligence yes. But that wouldn't require that every bot made in the journey imitates it. As long as the bot does the task it is designed to simplify, it should be satisfying.

However even with the expectations adjusted, it is difficult to convey the command when the command contains a deep query such as "Message Lisa reminding about the thing in my last note" or even continuity of conversation. It is difficult for bots to get these kind of constructs, machine learning helps extract entities and label parts however, this requires training them on domain specific data and for many domains, tons of labeled data is hard to come by. Without artificial intelligence truly powering the bots, the bots become the text version of a telephonic service line which says "press 1 for menu, press 2 for complaint booking." but more naturally and on text. However primitive, they do provide the same benefit of being present where the user is and use up resources of the user. It reduces the number of steps required and makes the service more accessible. We hence next describe various challenges that still stand in the way of natural language understanding and generation. They are general checkpoints we would want the systems to perform in order for the systems to feel natural.

A. Context Awareness

Conversational agents are not context aware as of now. If a user says "Text Jason and tell him that the time of the evening event is shifted to 7 p.m." The agent should be able to check the calendar for events and predict or ask about which event the user is talking about. Secondly, if the user says "I would like to order a black coffee in size medium" and then says " Change that to large" the agent should be able to recognize which factor must be changed.

B. Diversity of responses

Agents as of now lack diversity when they are rule based or when they are trained to extract from a certain set of responses. Generative algorithms that can generate diverse responses based on the situation after learning from a set of prebuilt responses while maintaining the meaning of the response could be worked upon.

C. Intention Driven Responses

Generating responses based on the intention would give way to more diverse responses and would reduce the need for domain specific training data. Since, responses are not built based on training data rather, they are built on events. The bot is aware of the information it has to convey and it conveys the information using dynamically generated response.

D. Personality[34]

As conversational agents proliferate, we will want them to be more human. One aspect of humanizing the bots will be adding personality to them. Pre-trained models respond according to the responses it is trained on. For a personality, all the responses in the training set would have to be entered in accordance to the personality. Doing so would be tedious. Hence, methods for generating responses in accordance with a given set of characteristics would help make the bots more human.

E. User Awareness

The conversational agents, especially the intrapersonal bots, would need to be aware of their users. They will need to remember who the user is and what the user is preference is. It might be easy to integrate preference in the bot when the bot is connected or has access to the user's social media profile but it will be difficult to do so based on only conversational information. Research is needed in extracting user persona based on user conversations and embedding the information about the user in the bot's replies. Not only would this make the bot more human but it would make the user feel affiliated to the bot on a personal level.

F. Continuity of conversation

For virtual assistants that aim to be friends or companions of the user, the bot should be able to converse when the user continues conversations that the user left midway. For

All rights reserved by *www.ijsrd.com*

example, if the user had been talking about an item and then says "Do you remember that mug I was talking about that had a blue handle?" the bot should be able to recall all other associated properties of the mug the user had mentioned.

Sr. No.	Platform	Intent Classification	Entity extraction	Sentiment Analysis	Keyword Extraction	Relies on rules considerably	Languages
1	Recast.AI	Yes	Yes	Yes	No	Yes	3 advanced languages: English, French and Spanish have all functionalities, 17 others have limited capabilities of entity extraction and intent classification. One agent can support multiple languages at the same time.
2	Wit.ai	Yes	Yes	Yes	No	Yes	50 languages available out of which 39 are in beta.
3	Api.ai	Yes	Yes	No	No	No	15 languages are supported. With every query, language tag must be sent. One language per agent and the language cannot be changed once the creation of the agent.
4	Chatfuel	Yes (weak)	No	No	No	Yes	More than 40 languages supported
5	Pandora Bots	Yes	Yes	No	Yes	Yes	Multilingual
6	Microsoft Bots, LUIS	Yes	Yes	Yes	Yes	No	More than 30 languages available for translation
7	Google Natural language API	No	Yes	Yes	Yes	No	Supports 9 languages, however, entity sentiment detection and text classification are available only for English
8	IBM Watson	No	Yes	Yes	Yes	No	Different services are available for different languages with English and Spanish having the most services, and keyword detection, entity detection present for all.[33]

Table 1: Comparison of platforms for building chatbots

G. *Narration*

The bot should be able to narrate the sequence of events as they occurred. For example, if the user placed an order with a site and then I ask my bot to cancel it, the bot should be able to narrate its own actions. Or, if the user asks the bot about the events of today, the bot should be able to link all the events together and narrate them sequentially. As bots get more personal, we would also want them to recall stories that another family member told them and narrate them back to us.

V. CONCLUSION

Chatbots won't be limited to the messenger window. The architecture mentioned above can be generalized for all systems of interaction. In the end, it is about using language to understand the construct of the world. Forms of language will change as time progresses, from text to emojis[35] to neuralink, but what will remain constant is the representations of the world in our mind and our need to convey our thoughts. Bots are the beginning of an interface between humans and artificial general intelligence. The architecture and the tools we surveyed help one have an idea of the kind of bot to be built for their system, what to expect from the bot and what tools to use to build it. We also hope that this provides a direction for further possible innovations and research areas in text based conversational interfaces.

REFERENCES

[1] J. Weizenbaum, “ELIZA--A Computer Program for the Study of Natural Language Communication between Man and Machine,” Commun. ACM, vol. 9, no. 1, pp. 36–45, 1966.
[2] B. A. Shawar and E. Atwell, “A comparison between Alice and Elizabeth Chatbot Systems,” Univ. Leeds, 2002.
[3] Amazon Developer, “Alexa.” [Online]. Available: https://developer.amazon.com/alexa. [Accessed: 23-Sep-2017].
[4] Oracle, “Chatbots 101,” pp. 1–4, 2017.
[5] VentureBeat, “Facebook Messenger chief says platform’s 34,000 chatbots are finally improving user experience | VentureBeat.” [Online]. Available: https://venturebeat.com/2016/11/11/facebook-messenger-chief-says-platforms-34000-chatbots-are-finally-improving-user-experience/. [Accessed: 23-Sep-2017].
[6] The Register, “Please don’t call them Facebook chatbots, says Facebook’s bot boss • The Register.” [Online]. Available: https://www.theregister.co.uk/2017/04/19/chatbots_facebook/. [Accessed: 23-Sep-2017].
[7] N. Jain and A. Jain, “A Survey on Popularity of Chat-Bots,” vol. 5, no. March, pp. 277–280, 2017.

[8] D. Britz, "Deep Learning for chatbots." [Online]. Available: www.wildml.com/2016/04/deep-learning-for-chatbots-part-1-introduction/.
[9] G. Neff and P. Nagy, "Automation, Algorithms, and Politics| Talking to Bots: Symbiotic Agency and the Case of Tay," Int. J. Commun., vol. 10, no. 0, p. 17, 2016.
[10] Pavel Surmenok, "Chatbot Architecture | Pavel Surmenok." [Online]. Available: http://pavel.surmenok.com/2016/09/11/chatbot-architecture/. [Accessed: 23-Sep-2017].
[11] L. Bradeško and D. Mladenić, "A Survey of Chabot Systems through a Loebner Prize Competition," Researchgate.Net, vol. 2, pp. 2–5, 2012.
[12] S. A. Abdul-Kader and J. Woods, "Survey on Chatbot Design Techniques in Speech Conversation Systems," IJACSA) Int. J. Adv. Comput. Sci. Appl., vol. 6, no. 7, pp. 72–80, 2015.
[13] R. Collobert and J. Weston, "A unified architecture for natural language processing," Proc. 25th Int. Conf. Mach. Learn. - ICML '08, pp. 160–167, 2008.
[14] Chatbots Magazine by Octane AI, "Machine Learning, Neural Networks and Algorithms – Chatbots Magazine." [Online]. Available: https://chatbotsmagazine.com/machine-learning-neural-networks-and-algorithms-5c0711eb8f9a. [Accessed: 23-Sep-2017].
[15] "Natural Language Pipeline for Chatbots – Pavel Surmenok – Medium." [Online]. Available: https://medium.com/@surmenok/natural-language-pipeline-for-chatbots-897bda41482. [Accessed: 23-Sep-2017].
[16] W. Khan, A. Daud, J. A. Nasir, and T. Amjad, "A survey on the state-of-the-art machine learning models in the context of NLP," Kuwait J. Sci, vol. 43, no. 4, pp. 95–113, 2016.
[17] L. A. Ramshaw and M. P. Marcus, "Text Chunking using Transformation-Based Learning," pp. 1–13, 1995.
[18] W. Lu, H. T. Ng, W. S. Lee, and L. S. Zettlemoyer, "A generative model for parsing natural language to meaning representations," Emnlp 2008, no. October, pp. 782–791, 2008.
[19] T.-H. Wen, M. Gasic, N. Mrksic, P.-H. Su, D. Vandyke, and S. Young, "Semantically Conditioned LSTM-based Natural Language Generation for Spoken Dialogue Systems," no. 2005, 2015.
[20] M. Sundermeyer, R. Schl, and H. Ney, "LSTM Neural Networks for Language Modeling," Proc. Interspeech, pp. 194–197, 2012.
[21] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to Forget: Continual Prediction with LSTM," Neural Comput., vol. 12, no. 10, pp. 2451–2471, 2000.
[22] Y. Goldberg and M. Elhadad, "splitSVM: Fast, Space-Efficient, non-Heuristic, Polynomial Kernel Computation for NLP Applications," 46th Annu. Meet. Assoc. Comput. Linguist. Hum. Lang. Technol., pp. 269–307, 2008.
[23] S. Dumais, J. Platt, D. Heckerman, and M. Sahami, "Inductive learning algorithms and representations for text categorization," Proc. 7th Int. Conf. Inf. Knowl. Manag., pp. 148–155, 1998.
[24] A. Kumar et al., "Ask Me Anything: Dynamic Memory Networks for Natural Language Processing," Proc. 33rd Int. Conf. Mach. Learn. PMLR 48, vol. 48, pp. 1378–1387, 2016.
[25] G. N. Goertzel B., Pennachin C., "Natural Language Generation.," in Engineering General Intelligence, Part 2. Atlantis Thinking Machines, vol 6., Atlantis Press, Paris, 2014.
[26] J. M. Alonso, A. Bugarin, and E. Reiter, "Natural Language Generation with Computational Intelligence atural," IEEE Comput. Intell. Mag., no. August, p. 8, 2017.
[27] K. Kocijan, P. Machonis, and M. A. X. Silberztein, "Linguistic Resources for Automatic Natural Language Generation," Autom. Nat. Lang. Gener. Work., 2017.
[28] T. Ingold, "Jumping NLP Curves: A Review of Natural Language Processing Research," J. R. Anthropol. Inst., vol. 4, no. January, pp. 771–773, 2014.
[29] Recast.ai, "Recast.AI | Collaborative Bot Platform." [Online]. Available: https://recast.ai/. [Accessed: 23-Sep-2017].
[30] "Wit.ai." [Online]. Available: https://wit.ai/.
[31] "API.AI." [Online]. Available: https://api.ai/. [Accessed: 23-Sep-2017].
[32] "chatfuel." [Online]. Available:
[33] https://chatfuel.com/.IBM Bluemix Docs, "Bluemix Docs: Natural Language Understanding." [Online]. Available:https://console.bluemix.net/docs/services/natural-language-understanding/language-support.html#language-support.[Accessed:24-Sep2017].
[34] G. Wilcock, "Chapter 1 Adaptivity and Response Generation."
[35] B. Felbo, A. Mislove, A. Søgaard, I. Rahwan, and S. Lehmann, "Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm," 2017.

All rights reserved by *www.ijsrd.com*