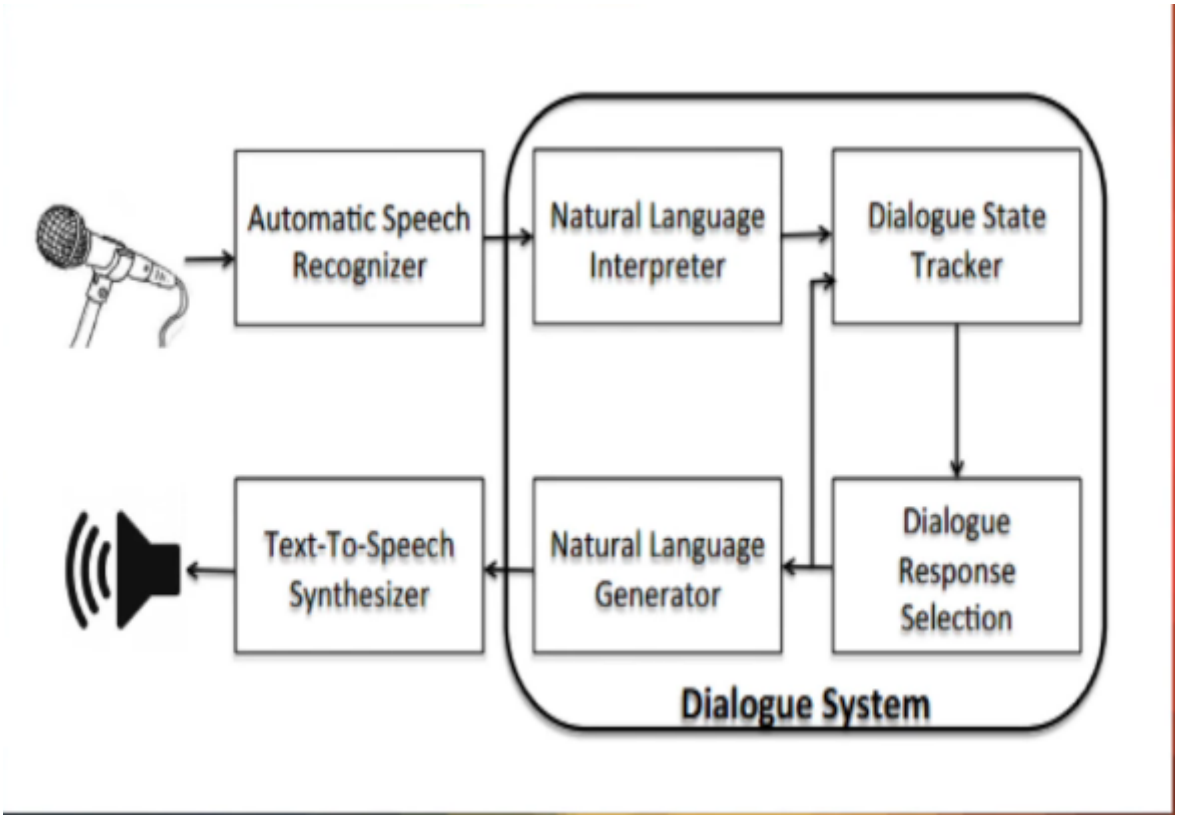
Chatbot using Chatterbot library:

https://github.com/gunthercox/chatterbot-corpus (Github link)

https://chatterbot.readthedocs.io/en/stable/ (Chatbot info link)
https://www.youtube.com/watch?v=_U146hWhDhM&feature=youtu.be



Supervised and Unsupervised Machine Learning Algorithms

https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/

Superwise learnig

It is called supervised learning because the process of an algorithm learning from the training dataset can be thought of as a teacher supervising the learning process. We know the correct answers, the algorithm iteratively makes predictions on the training data and is

corrected by the teacher. Learning stops when the algorithm achieves an acceptable level of performance.

Unsuperwise learning

These are called unsupervised learning because unlike supervised learning above there is no correct answers and there is no teacher. Algorithms are left to their own devises to discover and present the interesting structure in the data.

How artificial intelligence & machine learning produced robots we can talk to

https://www.businessinsider.com/what-is-chatbot-talking-ai-robot-chat-simulators-2017-10?IR=T

Top 15 Python Libraries for Data Science

https://medium.com/activewizards-machine-learning-company/top-15-python-libraries-for-data-science-in-in-2017-ab61b4f9b4a7

for AI chatbot we use

1)Numpy

The most fundamental package, around which the scientific computation stack is built, is NumPy (stands for Numerical Python). It provides an abundance of useful features for operations on n-arrays and matrices in Python. The library provides vectorization of mathematical operations on the NumPy array type, which ameliorates performance and accordingly speeds up the execution.

2)TensorFlow

Coming from developers at Google, it is an open-source library of data flow graphs computations, which are sharpened for Machine Learning. It was designed to meet the high-demand requirements of Google environment for training Neural Networks and is a successor of DistBelief, a Machine Learning system, based on Neural Networks. However, TensorFlow isn't strictly for scientific use in border's of Google —it is general enough to use it in a variety of real-world application.

The key feature of TensorFlow is their multi-layered nodes system that enables quick training of artificial neural networks on large datasets. This powers Google's voice recognition and object identification from pictures.

3)NLTK

The name of this suite of libraries stands for Natural Language Toolkit and, as the name implies, it used for common tasks of symbolic and statistical Natural Language Processing. NLTK was intended to facilitate teaching and research of NLP and the related fields (Linguistics, Cognitive Science Artificial Intelligence, etc.) and it is being used with a focus on this today.

The functionality of NLTK allows a lot of operations such as text tagging, classification, and tokenizing, name entities identification, building corpus tree that reveals inter and intra-sentence dependencies, stemming, semantic reasoning. All of the building blocks allow for building complex research systems for different tasks, for example, sentiment analytics, automatic summarization.

Library	Type	Commits	Contributors	Releases	Watch	Star	Fork	Commits / Contributors	Commits / Releases	Star/ Contributors
NumPy	Data wrangling	15980	522	125	280	4286	2012	31	128	8
SciPy	Data wrangling	17213	489	91	244	3043	1775	35	189	6
Pandas	Data wrangling	15089	762	76	626	9394	3709	20	199	12
Matplotlib	Visualization	21754	588	60	413	5190	2517	37	363	9
Seaborn	Visualization	1699	71	11	176	3878	580	24	154	55
Bokeh	Visualization	15724	223	40	322	5720	1401	71	393	26
Plotly	Visualization	2486	33	7	149	2044	512	75	355	62
SciKit-Learn	Machine learning	21793	842	80	1650	18246	9997	26	272	22
Keras	Machine learning	3519	428	28	1025	15043	5227	8	126	35
TensorFlow	Machine learning	16785	795	29	5002	55486	26433	21	579	70
Theano	Machine learning	25870	300	23	520	6171	2116	86	1125	21
Scrapy	Data scraping	6325	243	78	1427	20124	5353	26	81	83
NLTK	NLP	12449	196	20	376	4649	1358	64	622	24
Gensim	NLP	2878	179	43	300	4182	1595	16	67	23
Statsmodels	Statistics	8960	119	19	194	2019	977	75	472	17

ActiveWizards.com
28.04.2017

Implementing Artificial Neural Network training process in Python

https://www.geeksforgeeks.org/implementing-ann-training-process-in-python/

Underfitting and Overfitting in Machine Learning

https://www.geeksforgeeks.org/underfitting-and-overfitting-in-machine-learning/

Underfitting:

A statistical model or a machine learning algorithm is said to have underfitting when it cannot capture the underlying trend of the data. *(It's just like trying to fit undersized pants!)* Underfitting destroys the accuracy of our machine learning model. Its occurrence simply means that our model or the algorithm does not fit the data well enough. It usually happens when we have less data to build an accurate model and also when we try to build a linear model with a non-linear data. In such cases the rules of the machine learning model are too easy and flexible to be applied on such a minimal data and therefore the model will probably make a lot of wrong predictions. Underfitting can be avoided by using more data and also reducing the features by feature selection.

Overfitting:
A statistical model is said to be overfitted, when we train it with a lot of data *(just like fitting ourselves in an oversized pants!)*. When a model gets trained with so much of data, it starts learning from the noise and inaccurate data entries in our data set. Then the model does not categorize the data correctly, because of too much of details and noise. The causes of overfitting are the non-parametric and non-linear methods because these types of machine learning algorithms have more freedom in building the model based on the dataset and therefore they can really build unrealistic models. A solution to avoid overfitting is using a linear algorithm if we have linear data or using the parameters like the maximal depth if we are using decision trees.

How to avoid Overfitting:

The commonly used methodologies are:

- **Cross- Validation:** A standard way to find out-of-sample prediction error is to use 5-fold cross validation.
- **Early Stopping:** Its rules provide us the guidance as to how many iterations can be run before learner begins to over-fit.
- **Pruning:** Pruning is extensively used while building related models. It simply removes the nodes which add little predictive power for the problem in hand.
- **Regularization:** It introduces a cost term for bringing in more features with the objective function. Hence it tries to push the coefficients for many variables to zero and hence reduce cost term.

https://www.youtube.com/watch?v=nj_hChhSrOI