

System summary:

- The current model is basically an automated questions answering system suitable for customer support.
- Hyphen platform allows users to train bots by uploading questions and answers in text form which is later used by the bot to answer similar questions.
- In other words, the main purpose of using Machine Learning and Natural Language Processing techniques is to decide whether two questions\ statements are similar or not to allow the system to compare between the bot-user's input and the data (Q&A) uploaded by the platform user, and choose the right answer efficiently, and accurately in runtime speed.

Two types of users:

1. Platform user (company user) : This is the user, Hyphen directly interacts with. The user of our system who uses the platform to train (create) a chat-bot. This user provides data in text form representing possible questions with their answers. This user will provide text data to the platform and gets a trained chat-bot.
2. -chat-bot user (customer user): This is the user that will be using the trained bot. it will provide the bot with text question and get text answer similar to the trained data.

System characteristics:

- The model is simple, generic, and general for any type of question.
- Fully automated, platform users can use the platform to train the bot and use it directly without any need for human intervention.

System Software Design Description:

In order to achieve this we developed two main libraries, several classes and scripts, collected different data, and used many tools. This code is representing the intelligence of the bot and it is fully developed in python 3.

It can be divided into the following:

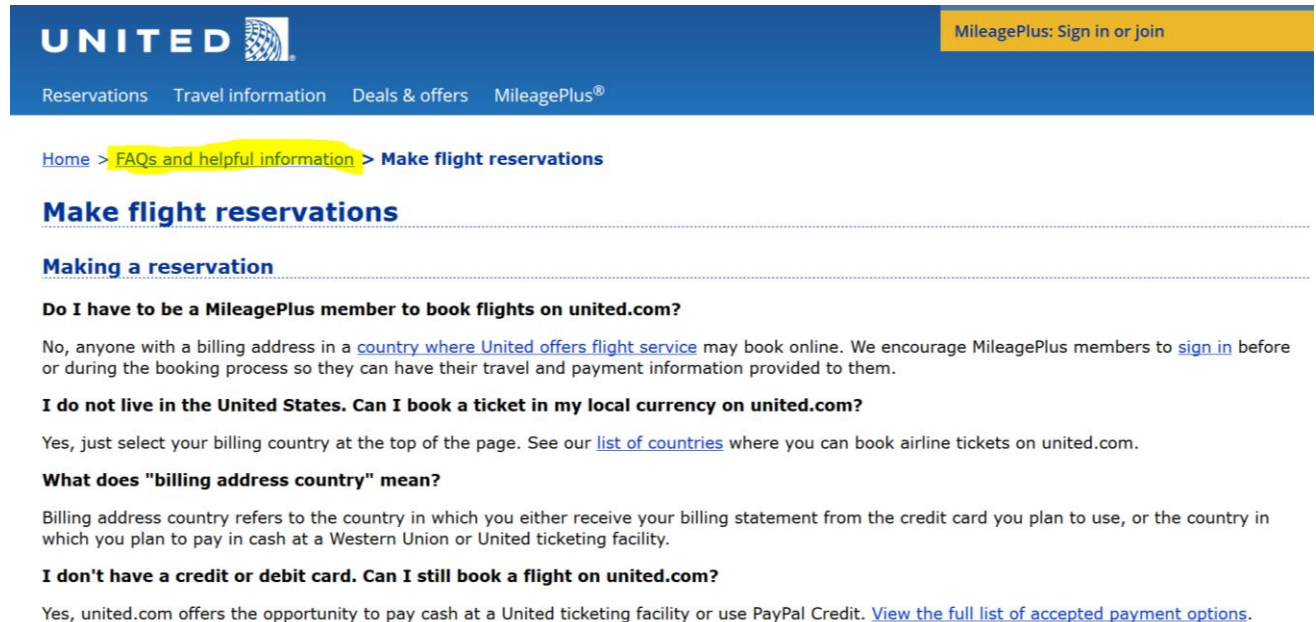
- ← Fmodel library: keras model (Neural networks Api) based on Stanford Natural Language Inference with the use of GloVe word embeddings)
- ← Dmodel library: LSTM model with word2vec
- ← Baseline model: simple SGD Classifier
- ← Utilities (different classes and scripts)
- ← Word2vec model (Arabic language)
- ← Data

- **Fmodel:**
- **The motivation:**

Important part of online Customer support is mainly about answering the customers inquiries and questions.

Also, many companies have customer support web pages with Q&A stored.

For example:



The screenshot shows the United MileagePlus website. The header includes the United logo and navigation links: Reservations, Travel information, Deals & offers, and MileagePlus®. A yellow banner at the top right says "MileagePlus: Sign in or join". Below the header, the breadcrumb trail is "Home > FAQs and helpful information > Make flight reservations". The main heading is "Make flight reservations". Underneath, there's a section titled "Making a reservation" with several FAQ items:

- Do I have to be a MileagePlus member to book flights on united.com?**
No, anyone with a billing address in a [country where United offers flight service](#) may book online. We encourage MileagePlus members to [sign in](#) before or during the booking process so they can have their travel and payment information provided to them.
- I do not live in the United States. Can I book a ticket in my local currency on united.com?**
Yes, just select your billing country at the top of the page. See our [list of countries](#) where you can book airline tickets on united.com.
- What does "billing address country" mean?**
Billing address country refers to the country in which you either receive your billing statement from the credit card you plan to use, or the country in which you plan to pay in cash at a Western Union or United ticketing facility.
- I don't have a credit or debit card. Can I still book a flight on united.com?**
Yes, united.com offers the opportunity to pay cash at a United ticketing facility or use PayPal Credit. [View the full list of accepted payment options.](#)

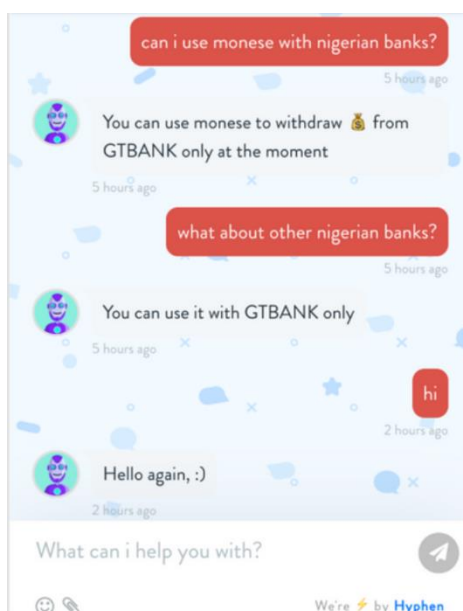
This model allow platform users to utilize such data, and train the chat-bot with. The input of **Fmodel** is pair of question, one question representing the customer input (question) to other is for the trained questions (questions uploaded by company or platform user).

Fmodel aims at matching the customer question to one of the questions that the chat-bot

were trained with. From here comes the concept of question pair.

For exmaple:

Using fmodel the platform would check any similar questions to "Can I use monese with Nigerian banks?" trained in the chat-bot. it does this by running questions in pairs, but efficiently and in runtime speed.



The **Quora** dataset has more than 400,000 pair of questions with same format. Which was a great resource for Fmodel Neural Network training.

Quora dataset pairs sample

id	qid1	qid2	question1	question2	is_duplicate
0	1	2	What is the step by step guide to invest in share market in india?	What is the step by step guide to invest in share market?	0
1	3	4	What is the story of Kohinoor (Koh-i-Noor) Diamond?	What would happen if the Indian government stole the Kohinoor (Koh-i-Noor) diamond back?	0
2	5	6	How can I increase the speed of my internet connection while using a VPN?	How can Internet speed be increased by hacking through DNS?	0
3	7	8	Why am I mentally very lonely? How can I solve it?	Find the remainder when 23^{24} is divided by 24,23?	0
4	9	10	Which one dissolve in water quickly sugar, salt, methane and carbon dioxide?	Which fish would survive in salt water?	0
5	11	12	Astrology: I am a Capricorn Sun Cap moon and cap rising...what does that say about me?	I'm a triple Capricorn (Sun, Moon and ascendant in Capricorn) What does this say about me?	1
6	13	14	Should I buy tiago?	What keeps children active and far from phone and video games?	0
7	15	16	How can I be a good geologist?	What should I do to be a great geologist?	1

Fmodel is a rule-based/ ML hybrid approach.

Fmodel is the main library used in the system. The purpose of this library is to decide whether two questions\ statements are similar or not, with certain confidence.

- Input: a pair of text strings (two questions\ statements)
- Output: similar or not with similarity score, with several features extracted.

Example 1:

Input:

Q1: where is the main building of CNN?**Q2: who works at the main building of CNN?**Output: *(the highlighted text is the system output and the quoted is comments on the output)*

```
{'sim': 0, "this means the two questions are not similar"
'sim_per': 43.127069680881142 "the score of similarity out of 100%"
'keras': 43.39699650461943 "keras model similarity score"
'keywords': [['main building', 'cnn'] "keywords of first question"
, ['main building', 'work', 'works', 'cnn']] "keywords of second question"
, 'max_keywords': 7, "maximum number of unique keywords in addition to question words"
'entities': [['CNN'], ['CNN']], "extracted named entities e.g. organizations"
'keywords_sim': 42.857142857142854, "keywords only similarity"
'class': [4, 3], "question class, '4' for location, '3' for people"
'f_class': False "not same class"
'sentiment': [0.0, 0.0, 0.0], "sentiment is neutral for both questions"
'numbers': [[], []], "no numbers in the questions"
}
```

--

Exmample 2:**Q1: who is the person****Q2: what is the person name**

```
{'keras': 100, 'sim': 1, 'keywords': [['person'], ['person']],
'max_keywords': 3, 'entities': [[], []], 'keywords_sim':
66.66666666666666, 'sim_per': 83.33333333333333, 'class': [3, 3],
'sentiment': [0.0, 0.0, 0.0], 'numbers': [[], []], 'f_class': True}
```

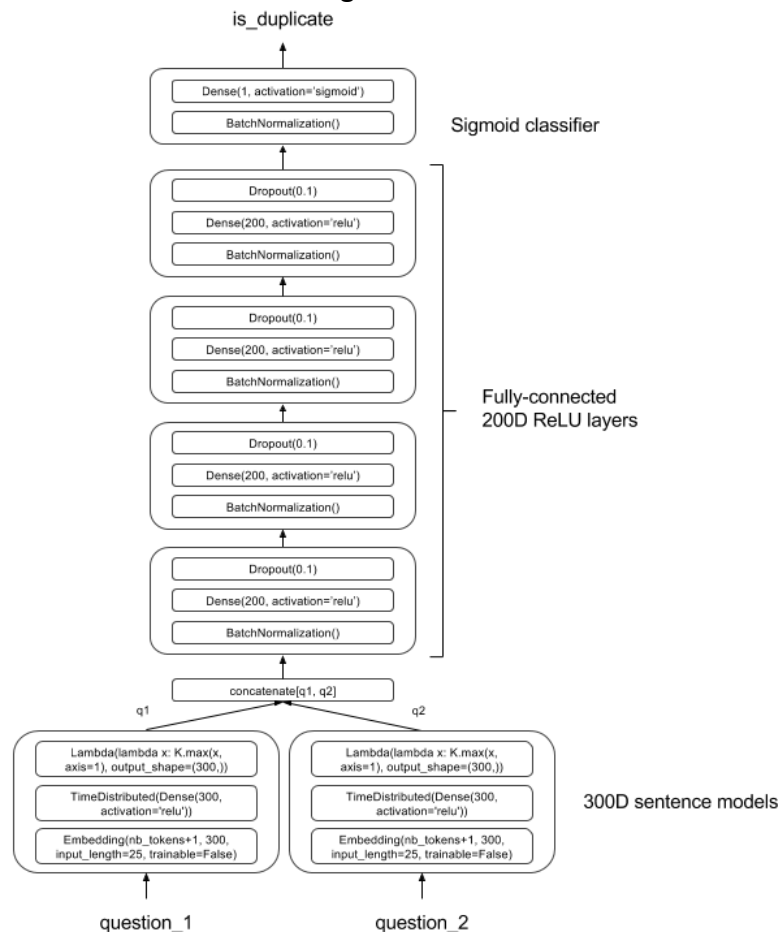
- Components:
 - **Keras model:** the main part of the output and decision making criteria is the score of Keras similarity model. Which is a multi-layer Neural Network model trained with Quora question-pairs dataset (more than 400,000 question pairs) with glove word embeddings . The model will be discussed in “utilities” section in the document.
 - **Sentiment analysis utility**
 - **Natural Language Toolkit (NLTK):** using NLTK different functions `word_tokenize`, `pos_tag`, `ne_chunk`, and `tree2conlltags` the model extract set of named entities.
 - **Mini similar:** special function for questions with only one word change. The model use Word Net Similarity to check the similarity of the two words.
 - **Preprocessing:** text normalization, spelling correction, stopwords removal, and words tokenizing .
 - **Negation:** extract negated words from the two questions
 - **Question classifier:** Classify questions according to questions intent (e.g. who is this -> class: person)
 - **Extract features:** module to extract keywords from the two question (important word)
 - **Spacy Named Entities:** using spacy which is an open source Python library to extract another set of named entities.
 - **Numbers:** numbers in questions are extracted as extra keywords (different formats are normalized e.g. 2 == two)

Output calculations:

By testing against different datasets (Quora and manual collected datasets) the following rules were found to be the best criteria for the model.

- Each keyword, question class, number, and named entity increment a counter with one, in either questions.
- This counter represents the maximum score for keywords.
- Using the keywords similarity and the keras similarity the overall similarity is calculated.
- The system has some special cases:
 - When the two questions have different named entities they are counted as not similar.
 - When the two questions have big sentiment difference they are also counted as not similar.
 - If the difference between the two questions is one word e.g. “who is that?” , “where is that”. A different function is run.

- Keras model:
 - It is Neural networks model using Keras tensors based on Stanford Natural Language Inference with the use of GloVe word embeddings.
 - The model is based on the project: <https://github.com/bradleypallen/keras-quora-question-pairs>
 - The model was trained with Quora dataset with more than 400,000 pairs of questions labeled as duplicates or not.
 - The model has the following architecture:



- The model showed good results in testing, but the results were not very stable when it's tested with more general live data that's why we had to develop the full fmodel and use keras as a feature in the model.

- Dmodel:
 - It is an implementation of deep Siamese LSTM Network using Theano tensors to evaluate two sentence similarity with words embeddings using word2vec in genism.
 - LSTM or Long Short Term Memory networks are special case of Recurrent Neural Networks
 - The code is based on the following project:
<https://github.com/aditya1503/Siamese-LSTM>
 - Word2vec model is imported from Gensim library and trained with GoogleNews-vectors-negative300.bin
 - The RNNs were trained with SemEval 2014 and tested with the same dataset
 - Testing with Fmodel (with keras model included) showed better results in other datasets such as Quora dataset and other ones, that is why Dmodel was dedicated for testing purposes and not used in the main model (fmodel)
- Utilities:
 - Preprocessing:
 - Includes spelling correction.
 - Text normilziation
 - Sentiment analysis :
 - Using the library “vaderSentiment” and using our own testing methods we set the needed thresholds.
 - Spelling check and correction:
 - First the script compares the word against list of English words, which can be manually manipulated for each user.
 - The spelling correction scripts have the usual base of getting all the possible variants of the wrong word and excludes the wrong spelled ones. With the addition of getting “hamming distance” between words and frequency of each word using certain text corpus.
 - Keywords extractor:
 - The script extracts keywords using our own defined grammar, Spacy library, Textblob library, Rake library, and NLTK.
 - Features extractor:
 - The script extract normalized numbers and noun chunks.
 - Coference resolution:
 - The script do Coreference resolution for text.
 - Questions classifier:
 - A script to classify questions into different classes.

- Word2vec model (Arabic language):
 - Due to the scarce resources for Arabic language we had to use an unsupervised approach for similarity calculations. We used gensim model with our own datasets which consists of thousands of movie subtitles in Arabic.
- Data and Tools:
 - Quora question pairs
 - glove.840B.300d
 - GoogleNews-vectors-negative300.bin
 - 20_newsgroup
 - Real life Q&A datasets
 - Stopwords lists
 - Words
 - NLTK
 - Spacy
 - Word Net Similarity
 - Theano
 - Gensim