



# Innominds

## Machine Learning Hackathon

*Develop an application that suggests best solution to a question based on past behavior & available solutions.*



# About You

A few words about you. What do you do, Where do you work, what are your interests..

Currently an undergrad student at *Jawaharlal Nehru Technological University Hyderabad*. At the moment I am Data Science Intern at Analytics Vidhya where I am responsible building in house machine learning products for the company

## Here's list of my achievements:

- 1st Runner-up and winner of ₹1,00,000 prize at **TechGig** in Machine Learning Hackathon by **CreditSuisse**, 2018
- Rank 2 with 1636 points at student rankings by Analytics Vidhya with just 27 hackathon participation.
- Rank 1 and ₹50,000 grand prize winner at national level data science hackathon by JNTUH and Colaberry - USA, 2018
- Rank 1 out of 200 applicants at Analytics Vidhya Hiring Hackathon, 2018
- Rank 9 out of 261 participants at “Data Tales – Beyond Infinity” competition on Analytics Vidhya, 2017

# Core Idea

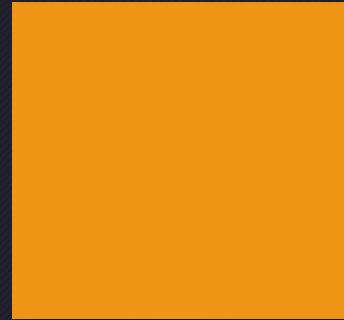
Explain the given problem statement in your words

As the reach of technology roots to new customers in the Internet era there is increase in customer demands and queries. Current implementation requires L1 technician to go through the manuals and find the apt solution.

This is posing new problems when the requests are scaled at high demand and also the resolution is not correct. Due to this the company is loosing its loyal customers which is not good for a business.

# Implementation Overview

Explain the solution submitted in your words



The current project idea presents standard Natural Language Processing Techniques and use Deep learning model to train on the available data to better predict the use cases and resolutions.

The data from the input files is cleaned and processed under [data.csv](#) in the root directory of back-end folder. The models are trained on this data to predict the resolution for a passed query.

The data.csv is processed file which contains the resources from given data, FAQs, Tickets and summarized into a single .csv file for training purpose.



# Technology/Tool/Components used

Explain the Technologies, Languages used in your solution.



The tools used are:

1. NLTK
2. Doc2Vec Model
3. Flask
4. PostgreSQL
5. Flask Restful
6. Cross Origin Extension
7. Stopwords
8. Deep Learning (Gensim)

# Architecture overview

Explain the Architecture you followed in your solution

The main approach adopted is generating a generalized **Doc2Vec** model. First the queries and the resolutions are tokenized. The sentences are later converted into **TaggedDocuments** with the help of gensim library.

Later the tagging is applied from query to resolution. The gensim model is trained for 100 epochs for titles and 600 epochs for the body of document. The models individually are saved. The **deep learning model** is trained on the data and the models train in about **1 min 58 seconds**

Later the query is taken as input from the front-end search / bot interface. At back-end, function **GetSolution** takes the query string and preprocess into tokenization and later inferred vectors. After the query string is vectorized. The cosine similarity of the query is compared with the existing documents.

The items with highest similarities are then sorted and returned back to front-end

# Elaboration of the usage

How does your application work.

The application currently takes keywords/tags or sentences in the search string and outputs relevant resolutions.

The points to be noted are:

- The keywords should not be misspelled
- Unambiguous search will lead to random results.
- The results are often ordered according to similarity scores. Hence, the solutions from the top 5 should be considerable.
- If the resolution is not expected. Click the magic button again to sent request again to process the vector. This should give more precise results with more data

# Number of Output Questions Working

In the output sheet we provided how many questions are showing the correct answers and what are those.

Almost all the question work. The deep learning model currently suffers due to lack of data of past handled tickets. In the challenge we have abundant data for computer queries which work 100% correctly. Due to less data the login and ticket issues require double checks on "Magic Button" to get correct resolutions. Here are screen shorts of working demo.

How can I Delete Internet Cookies	
Magic Engine	
0	microsoft internet explorer users can go to tools or the little cog icon in the top left then go to safety and choose delete browsing history you can then choose to delete your internet cookies in google chrome go to more tools and choose clear browsing data firefox users can go to history then choose clear recent history
1	visit link <a href="https://www.ccric.edu/it/helpdesk/images/network_flowchart.gif">https://www.ccric.edu/it/helpdesk/images/network_flowchart.gif</a>



## email issue

Magic Engine

- |   |  |
|---|--|
| 0 | verify that your computer is able to see the internet and or other computers to ensure that your computer is not encountering connection issue which would be causing your mail issue ensure that your internet mail server or your network mail server is not encountering issues by contacting either your internet service provider or your network administrator |
|---|--|

## Computer is not getting switched on

Magic Engine

- |   |  |
|---|--|
| 0 | make sure both the computer and monitor are on make sure the monitor is securely plugged into the computer make sure the power cable is firmly plugged into the monitor some computers have multiple display ports so make sure you have plugged the monitor into the correct one try each one in turn switching the monitor off then on in between moves most monitors have status window displayed when you turn it on check if you can see this status window when you press the power button on the monitor you can also try this with the menu button on the monitor which should bring up an options menu on the screen this shows the screen is working ok so it may be an issue with the video cable from the monitor or the machine itself check the brightness contrast levels of the monitor via the menu button to make sure it has not been set too dark move the |
|---|--|

# Prototype Demo

Add a video link of the workable solution. Please cover as many output scenario's as you can show from the output excel provided. (Mandatory)

<https://youtu.be/UqJnwbmMfk0>

# Source Code

GIT Link and Compile Instructions, Solution Hosted URL

<https://github.com/shaz13/InnBot>

Instructions are written in the README.md of repo

# Completeness

Mark the area's which have been completed and tested. They will be a part of the evaluation parameters.

Area	Description
Performance - How did you measure the performance of this system.	The output was measured with the Sample output provided. If the results appear in predicted first 3 results then the model performs well for that case.
Usability - How usable is your solution	Its completely usable and deployable.
Completeness - What has been completed.	The search backend, deep learning model and front end is done.
Pending - What are the items which are pending	Scalable items that tracks events from front end are left over.



# Possible Improvement

Any Improvements you want to add?

Sure, do. I wanted to take this implementation one step further in future as in –

- Predicting the follow up question to the asked question
- Personalization to user/technician level. And also query specific bots that excel in their domain. IT/Service/Hardware/Software
- Question Continuity. Example – My Ctrl key doesn't work. Next time user asks it, It works but seldom. Then the bot should be relating the query to the Ctrl key back
- Adding more data. With more data the model will perform with higher accuracies on past tickets.



# Thank You InnoMinds!

*You really got us innovating our minds again.*

*~Shaz*