**Problem statements :**

Asymmetric type of semantic search where for a given query from customer we have to find whether the query had been told while conversation or not .

**Approach :**

Step 1 : Reading and sentence tokenize

Step 2 : Finding embedding of each sentences and the queries that we have received

Step 3 : Calculating the cosine similarity

Step 4 : Storing the sentence and cosine similarity into dictionaries

Step 5 : Fetching top 5 sentences with similarities scores

**Model used :**

I have used the BERT ( Bidirectional encoder representation from transformer ) pre-trained model .I had tested on different pre-trained models but these are the two models which I found are good .

| Sr .no | Name of model |
| --- | --- |
| 1 | msmarco-distilbert-base-dot-prod-v3 |
| 2 | msmarco-distilbert-base-v4 |

**Data used :**

Transcription of conversation between policybazaar agent and customer.

Data : [Link](https://www.kaggle.com/datasets/abdulkayumshaikh/agent-customer-conversation)

**Results :**

**Question that I have asked while conversation :**

* **'what is healthcare insurance please explain be briefly'**
* **' The agent told me that this policy cover everything but now they are asking me to pay some amount.'**
* **'Are there any specific hospital to claim the policy'**
* **'Do I have to renew the policy every year ?'**
* **'how to claim the policy ,what is the procedure to claim the policy '**

**Question that I have not asked while conversation :**

* ‘**Is this policy to cover charges of ambulance fee also ?'**
* **'dental operation is also cover in this policy agent said '**
* **'This wadapav is very tasty i think you should also try'**
* **'your agent used abusive words'**
* **'Agent are forcing to buy policy please talk to your agent'**

**Result from model 1 :**

| **Top 5 similarity scores for question that I had asked** | | | | | |
| --- | --- | --- | --- | --- | --- |
|
| **Questions** | **Top 1** | **Top 2** | **Top 3** | **Top 4** | **Top 5** |
| 1 | 0.58 | 0.55 | 0.51 | 0.5 | 0.48 |
| 2 | 0.58 | 0.57 | 0.56 | 0.55 | 0.54 |
| 3 | 0.78 | 0.61 | 0.59 | 0.59 | 0.52 |
| 4 | 0.65 | 0.64 | 0.45 | 0.43 | 0.41 |
| 5 | 0.69 | 0.65 | 0.49 | 0.49 | 0.47 |

**Green highlights that presence of sentence in the documents of particula queries .**

| **Top 5 similarity scores for question that I had not asked** | | | | | |
| --- | --- | --- | --- | --- | --- |
|
| **Questions** | **Top 1** | **Top 2** | **Top 3** | **Top 4** | **Top 5** |
| 1 | 0.57 | 0.57 | 0.55 | 0.55 | 0.55 |
| 2 | 0.55 | 0.52 | 0.51 | 0.5 | 0.49 |
| 3 | 0.44 | 0.4 | 0.4 | 0.4 | 0.39 |
| 4 | 0.37 | 0.31 | 0.3 | 0.3 | 0.3 |
| 5 | 0.51 | 0.48 | 0.42 | 0.4 | 0.4 |

**Results from model 2 :**

| **Top 5 similarity scores for question that I had asked (Model2)** | | | | | |
| --- | --- | --- | --- | --- | --- |
|
| **Questions** | **Top 1** | **Top 2** | **Top 3** | **Top 4** | **Top 5** |
| 1 | 0.58 | 0.41 | 0.39 | 0.36 | 0.36 |
| 2 | 0.65 | 0.48 | 0.45 | 0.42 | 0.42 |
| 3 | 0.58 | 0.53 | 0.53 | 0.51 | 0.49 |
| 4 | 0.58 | 0.45 | 0.29 | 0.23 | 0.2 |
| 5 | 0.63 | 0.4 | 0.34 | 0.3 | 0.29 |

**Green highlights that presence of sentence in the documents of particula queries .**

| **Top 5 similarity scores for question that I had not asked (Model2)** | | | | | |
| --- | --- | --- | --- | --- | --- |
|
| **Questions** | **Top 1** | **Top 2** | **Top 3** | **Top 4** | **Top 5** |
| 1 | 0.52 | 0.5 | 0.48 | 0.45 | 0.43 |
| 2 | 0.39 | 0.39 | 0.36 | 0.34 | 0.33 |
| 3 | 0.58 | 0.41 | 0.39 | 0.36 | 0.36 |
| 4 | 0.16 | 0.13 | 0.13 | 0.12 | 0.12 |
| 5 | 0.36 | 0.27 | 0.22 | 0.21 | 0.19 |

**Conclusion :**

From above results i can conclude that when we have any document if particular query will come it will give the top k sentence related to query and we have to fix one threshold based on cosine similarity score if any score is greater than or equal to threshold that means the particular query have been told while conversation and if it is below threshold then we can sasy that the query have not told while conversations .Model 2 seems to be performing good for this problem .