#### Usage

Run evidence.py in terminal or in PyCharm. (first time may need longer time to download the models)

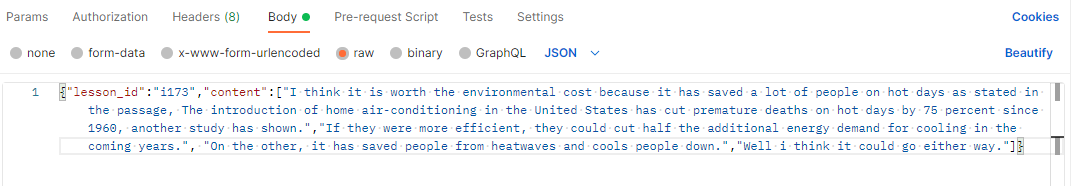
As long as the program is running on localhost we can test it with postman or curl command.

##### Postman examples

Select Post request and use <http://localhost:8008/predict> as IP address.

屏幕截图 2023-01-12 141559

Edit in body, and select json type. Then copy our example json in it or enter input you want to. After sending you will get the result from the program.



Example json:

{

"lesson\_id": "i173",

"content": [

"I think it is worth the environmental cost because it has saved a lot of people on hot days as stated in the passage, The introduction of home air-conditioning in the United States has cut premature deaths on hot days by 75 percent since 1960, another study has shown.",

"If they were more efficient, they could cut half the additional energy demand for cooling in the coming years.",

"On the other, it has saved people from heatwaves and cools people down.",

"If greenhouse gas emissions increase then the overall temperatures would increase, causing more demand for air conditioning."

]

}

##### Curl command

Type Curl command in terminal will also return the result.

Example of curl :

curl -X POST http://localhost:8008/predict -H "Content-Type:application/json" -d "{\"lesson\_id\":\"i173\", \"content\":[\"I think it is worth the environmental cost because it has saved a lot of people on hot days as stated in the passage, The introduction of home air-conditioning in the United States has cut premature deaths on hot days by 75 percent since 1960, another study has shown.\",\"If they were more efficient, they could cut half the additional energy demand for cooling in the coming years.\", \"On the other, it has saved people from heatwaves and cools people down.\",\"If greenhouse gas emissions increase then the overall temperatures would increase, causing more demand for air conditioning.\"]}"

#### Input and output explain

##### Input

{

"lesson\_id": "i173",

"content": [

"I think it is worth the environmental cost because it has saved a lot of people on hot days as stated in the passage, The introduction of home air-conditioning in the United States has cut premature deaths on hot days by 75 percent since 1960, another study has shown."

]

}

* **lesson\_id** string type

This is the ID use to look up the item. Maybe in the future we will replace with GUID generated by the other App. (now we support i172 and i173)

* **content** list type

This is the list of sentences we want to pass the program.(raw text of the student response)

**Output**

{

"fact\_all\_n": [

"growing concern of other countries",

...

],

"fact\_all\_y": [

"cut premature deaths",

...

],

"fact\_lack\_n": [

"growing concern of other countries",

...

],

"fact\_lack\_y": [

"forgoing air-conditioning can be deadly",

...

],

"fact\_present\_y": [

"cut premature deaths"

],

"facts\_present\_n": [

"greenhouse gas, global warming"

],

"num\_support\_n": 1,

"num\_support\_y": 1,

"number\_general\_info": 0,

"number\_solution": 0,

"sentence\_label\_n": [

255,

3

],

"sentence\_label\_y": [

0,

255

],

"sentences\_label\_ge": [

255,

255

],

"sentences\_label\_sol": [

255,

255

]

}

* **fact\_all\_n** (list of string) shows all categories support the no answer.(in our case is AC is not worth environmental cost)
* **fact\_all\_y** (list of string) shows all categories support the yes answer.
* **fact\_present\_y** (list of string) shows results which categories have been found by the model that support the yes answer.
* **fact\_present\_n** (list of string) shows results which categories have been found by the model that support the no answer.
* **fact\_lack\_y** (list of string) shows results which categories have not been found by the model support yes answer compare to all categories that support the yes answer.
* **fact\_lack\_n** (list of string) shows results which categories have not been found by the model support no answer compare to all categories that support the no answer.
* **Number\_support\_n** (int) shows how many sentences support the no answer.
* **Number\_support\_y** (int)shows how many sentences support the yes answer.
* **Number\_general\_info** (int)shows how many sentences are general information.
* **Number\_solution** (int)shows how many sentences are solutions.
* **Sentence\_label\_n** (list of int) label of each sentences compare to the no answer categories. 255 means no match. 1 means match with the second category in fact\_all\_n.
* **Sentence\_label\_y** (list of int) label of each sentences compare to the yes answer categories. 255 means no match. 2 means match with the third category in fact\_all\_y.
* **Sentence\_label\_ge** (list of int) label of each sentences compare to the general information. Only have 255 and 0 labels. 255 means no match.
* **Sentence\_label\_sol** (list of int) label of each sentences compare to the solution. Only have 255 and 0 labels. 255 means no match.