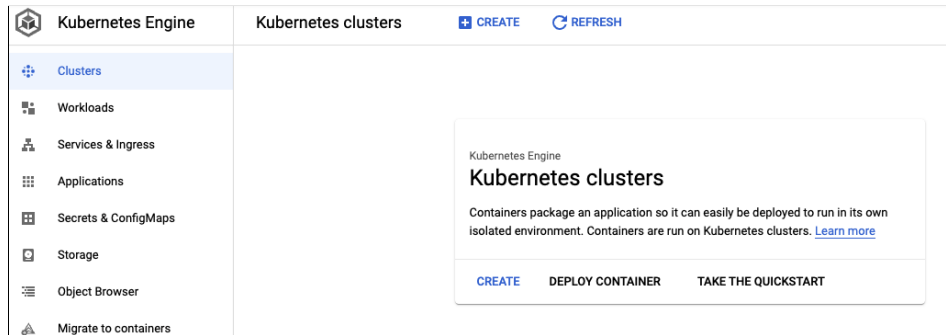


# Deploying Automated Essay Scoring System

## 1. Creating a cluster on GKE by using the GCP UI

Click Create then choose GKE standart



Configuration is like below:

Create a Kubernetes cluster

ADD NODE POOL

REMOVE NODE POOL

USE A SETUP GUIDE

Cluster basics

NODE POOLS

default-pool

CLUSTER

Automation

Networking

Security

Metadata

Features

Cluster basics

The new cluster will be created with the name, version, and in the location you specify here. After the cluster is created, name and location can't be changed.

To experiment with an affordable cluster, try **My first cluster** in the Cluster set-up guides

Name

ml-cluster

Location type

Resource prices may vary between certain regions. [Learn more](#)

☒ Zonal

☐ Regional

Zone

us-central1-c

☐ Specify default node locations

Current default: us-central1-c

Control plane version

Choose a release channel for automatic management of your cluster's version and upgrade cadence. Choose a static version for more direct management of your cluster's version. [Learn more](#).

☐ Static version

☒ Release channel

Release channel

Regular channel (default)

Version

1.21.6-gke.1503 (default)

These versions have passed internal validation and are considered production-quality, but don't have enough historical data to guarantee their stability. Known issues generally have known workarounds. [Release notes](#)

←

Create a Kubernetes cluster

+ ADD NODE POOL

REMOVE NODE POOL

USE A SETUP GUIDE ▾

Cluster basics

NODE POOLS

default-pool

Nodes

Security

Metadata

CLUSTER

Automation

Networking

Security

Metadata

Features

Node pool details

The new cluster will be created with at least one node pool. A node pool is a template for groups of nodes created in this cluster. More node pools can be added and removed after cluster creation.

Name

default-pool

Control plane version - 1.21.6-gke.1503

Size

Number of nodes \*

1

Pod address range limits the maximum size of the cluster. [Learn more](#)

☐ Enable autoscaling ?

☐ Specify node locations ?

Default: us-central1-c

Automation

☒ Enable auto-upgrade ?

⚠

Clusters smaller than 3 nodes may experience downtime during upgrades

☒ Enable auto-repair ?

Surge upgrade ?

Max surge \*

1

Max unavailable \*

0

←

Create a Kubernetes cluster

+ ADD NODE POOL

REMOVE NODE POOL

USE A SETUP GUIDE ▾

Cluster basics

NODE POOLS

default-pool

Nodes

Security

Metadata

CLUSTER

Automation

Networking

Nodes

These node settings will be used when new nodes are created using this node pool.

Image type

Container-Optimized OS with containerd (cos\_containerd) (default) ▾ ?

⚠

The default Linux node image for newly created clusters and node pools with version 1.21.6-gke.1503 or later is Container-optimized OS with Containerd. For Windows node pools using version 1.21 or later, Containerd is also the recommended runtime. Since Dockershim is being deprecated by Kubernetes project, [GKE will deprecate Docker node images](#). We recommend that you [migrate to containerd node images](#) as soon as possible. Learn more about the different [node images](#).

Machine Configuration ?

### Machine Configuration ?

**Machine family**


[GENERAL-PURPOSE](#)
[COMPUTE-OPTIMIZED](#)
[MEMORY-OPTIMIZED](#)
[GPU](#)

High-performance machine types for compute-intensive workloads

**Series** C2

Powered by Intel Cascade Lake CPU platform

**Machine type** c2-standard-4 (4 vCPU, 16 GB memory)



**vCPU**

4

**Memory**

16 GB

**✓ CPU PLATFORM AND GPU**

**Boot disk type** Standard persistent disk

**Boot disk size (GB)** 100

☐ Enable customer-managed encryption for boot disk ?

Local SSD disks ?

☐ Enable preemptible nodes ?

Click Create, creating might take 3-4 min. After creation cluster can be seen under Kubernetes Engine -> Clusters

Kubernetes Engine

Clusters

Workloads

Services & Ingress

Applications

Secrets & ConfigMaps

✕

1 Kubernetes cluster selected

DELETE

LABELS

OVERVIEW

COST OPTIMIZATION

Filter

Enter property name or value

?

⋮

<div><div><input checked="" type="checkbox"/></div></div> Status	Name <div>↑</div>	Location	Number of nodes	Total vCPUs	Total memory	Notifications	Labels
<div><div><input checked="" type="checkbox"/></div><div><div></div><div>✓</div></div></div>	<a href="#">ml-cluster</a>	us-central1-c	1	4	16 GB	—	<div>⋮</div>

Click 3 dots and select connect.

## Connect to the cluster

You can connect to your cluster via command-line or using a dashboard.

### Command-line access

Configure [kubect!](#) command line access by running the following command:

```
$ gcloud container clusters get-credentials ml-cluster --zone us-central1-c --project cs571-cci
```

[RUN IN CLOUD SHELL](#)

### Cloud Console dashboard

You can view the workloads running in your cluster in the Cloud Console [Workloads dashboard](#).

[OPEN WORKLOADS DASHBOARD](#)

OK

## 2. Creating a VM and give permission for kubernetes

Run the following command in cloudshell to create a VM and give permission. More detail [here](#)

```
gcloud compute instances create [INSTANCE_NAME] \
  --service-account [SERVICE_ACCOUNT_EMAIL] \
  --scopes [SCOPES,...]
```

Copy the command line access command then run in VM:

```
dabanoglu19588@gcloud:~$ gcloud container clusters get-credentials ml-cluster --zone us-central1-c --project cs571-cci
Fetching cluster endpoint and auth data.
kubeconfig entry generated for ml-cluster.
dabanoglu19588@gcloud:~$
```

## 2. Miniconda Installation

Download the latest shell script for miniconda installation

```
wget
https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
```

```
dabanoglu19588@cloud:~$ wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
--2022-04-06 10:47:10-- https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
Resolving repo.anaconda.com (repo.anaconda.com)... 104.16.131.3, 104.16.130.3, 2606:4700::6810:8303, ...
Connecting to repo.anaconda.com (repo.anaconda.com)|104.16.131.3|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 75660608 (72M) [application/x-sh]
Saving to: 'Miniconda3-latest-Linux-x86_64.sh'

Miniconda3-latest-Linux-x86_64.s 100%[=====>] 72.16M 221MB/s in 0.3s
2022-04-06 10:47:10 (221 MB/s) - 'Miniconda3-latest-Linux-x86_64.sh' saved [75660608/75660608]
```

Make the miniconda installation script executable

```
chmod +x Miniconda3-latest-Linux-x86_64.sh
```

Run miniconda installation script

```
./Miniconda3-latest-Linux-x86_64.sh
```

```
Do you accept the license terms? [yes|no]
[no] >>>
Please answer 'yes' or 'no':
>>>
Please answer 'yes' or 'no':
>>> yes

Miniconda3 will now be installed into this location:
/home/dabanoglu19588/miniconda3

- Press ENTER to confirm the location
- Press CTRL-C to abort the installation
- Or specify a different location below

[/home/dabanoglu19588/miniconda3] >>>
PREFIX=/home/dabanoglu19588/miniconda3
Unpacking payload ...
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

environment location: /home/dabanoglu19588/miniconda3
```

Type Yes

```
Preparing transaction: done
Executing transaction: done
installation finished.
Do you wish the installer to initialize Miniconda3
by running conda init? [yes|no]
[no] >>> █
```

```
Thank you for installing Miniconda3!  
dabanoglu19588@cloudshell:~ (cs571-cci) $
```

## IMPORTANT NOTE!!

Activate conda environment to run conda commands:

```
. <path_to_your_conda_install>/etc/profile.d/conda.sh
```

```
conda activate base
```

```
dabanoglu19588@gcloud:~$ . miniconda3/etc/profile.d/conda.sh  
dabanoglu19588@gcloud:~$ conda activate base  
(base) dabanoglu19588@gcloud:~$
```

You should see the (base) in the beginning of the line to run conda commands

## 2. Creating Python Environment by using Miniconda

```
conda create -n myenv python=3.6
```

```
conda activate myenv
```

## 3. Downloading the project from GitHub

```
git clone https://github.com/Quan25/flask-summary.git
```

```
final_project
```

```
(myenv) dabanoglu19588@gcloud:~$ git clone https://github.com/Quan25/flask-summary.git final_project  
Cloning into 'final_project'...  
remote: Enumerating objects: 191, done.  
remote: Counting objects: 100% (191/191), done.  
remote: Compressing objects: 100% (136/136), done.  
remote: Total 191 (delta 91), reused 148 (delta 51), pack-reused 0  
Receiving objects: 100% (191/191), 710.58 KiB | 6.64 MiB/s, done.  
Resolving deltas: 100% (91/91), done.
```

Go to final\_project folder

```
cd final_project
```

Type the following command to download rough.zip to your directory

```
wget --load-cookies /tmp/cookies.txt
```

```
"https://docs.google.com/uc?export=download&confirm=$(wget
```

```
--quiet --savecookies/tmp/cookies.txt --keep-session-cookies
```

```
--no-check-certificate
```

```
'https://drive.google.com/file/d/1RxzfZOYyNvzvCf37_vABfJMkohAsEZK
```

```
th/' -O- | sed -rn
```

```
's/.confirm=([0-9A-Za-z_]+)\\.\\1\\n/p')&id=1RxfZOYyNvzvCf37_vABfJM  
kohAsEZKtH" -O rough.zip && rm -rf /tmp/cookies.txt
```

```
(myenv) dabanoglu19588@gcloud:~/final_project$ wget --load-cookies /tmp/cookies.txt "https://docs.google.com/uc?export=download&confirm=$(wget  
--quiet --savecookies/tmp/cookies.txt --keep-session-cookies --no-check-certificate 'https://drive.google.com/file/d/1RxfZOYyNvzvCf37_vABfJM  
kohAsEZKtH/' -O- | sed -rn 's/.confirm=([0-9A-Za-z_]+)\\.\\1\\n/p')&id=1RxfZOYyNvzvCf37_vABfJMkohAsEZKtH" -O rough.zip && rm -rf /tmp/cookies.txt  
wget: unrecognized option '--savecookies/tmp/cookies.txt'  
Usage: wget [OPTION]... [URL]...  
  
Try 'wget --help' for more options.  
Cannot open cookies file '/tmp/cookies.txt': No such file or directory  
--2022-04-06 10:54:22-- https://docs.google.com/uc?export=download&confirm=$(wget  
Resolving docs.google.com (docs.google.com)... 142.250.152.101, 142.250.152.113, 142.250.152.102, ...  
Connecting to docs.google.com (docs.google.com)[142.250.152.101]:443... connected.  
HTTP request sent, awaiting response... 303 See Other  
location: https://doc-14-a0-docs.googleusercontent.com/docs/securesc/ha0ro937gcuc717defkksulhg5h7mbpl/fogn79i97gealeprfhdccl3p9lu3sqf/1649242425000/09591  
149636102722091/*1RxfZOYyNvzvCf37_vABfJMkohAsEZKtH?e=download [following]  
Warning: wildcards not supported in HTTP.  
--2022-04-06 10:54:27-- https://doc-14-a0-docs.googleusercontent.com/docs/securesc/ha0ro937gcuc717defkksulhg5h7mbpl/fogn79i97gealeprfhdccl3p9lu3sqf/1649  
242425000/09591049636102722091/*1RxfZOYyNvzvCf37_vABfJMkohAsEZKtH?e=download  
Resolving doc-14-a0-docs.googleusercontent.com (doc-14-a0-docs.googleusercontent.com)... 142.250.159.132, 2607:f8b0:4001:c58::84  
Connecting to doc-14-a0-docs.googleusercontent.com (doc-14-a0-docs.googleusercontent.com)[142.250.159.132]:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 3916745 (3.7M) [application/zip]  
Saving to: 'rough.zip'  
  
rough.zip 100%[=====] 3.73M --.-KB/s in 0.02s  
2022-04-06 10:54:28 (188 MB/s) - 'rough.zip' saved [3916745/3916745]
```

Unzip the rough.zip:

```
unzip rough.zip
```

## 4. Installing libxml-parser-perl and perl modules.

This step is required for installing ROUGE-1.5.5

```
sudo apt-get install libxml-parser-perl
```

```
(myenv) dabanoglu19588@gcloud:~/final_project$ sudo apt-get install libxml-parser-perl  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following additional packages will be installed:  
  libauthen-sasl-perl libdata-dump-perl libencode-locale-perl libfile-listing-perl libfont-afm-perl libhtml-form-perl  
  libhtml-format-perl libhtml-parser-perl libhtml-tagset-perl libhtml-tree-perl libhttp-cookies-perl libhttp-daemon-pe  
  libhttp-date-perl libhttp-message-perl libhttp-negotiate-perl libio-html-perl libio-socket-ssl-perl liblwp-mediatype  
  liblwp-protocol-https-perl libmailtools-perl libnet-http-perl libnet-smtp-ssl-perl libnet-ssleay-perl libtimedate-pe  
  libtry-tiny-perl liburi-perl libwww-perl libwww-robotrules-perl perl-openssl-defaults  
Suggested packages:  
  libdigest-hmac-perl libgssapi-perl libcrypt-ssleay-perl libauthen-ntlm-perl  
The following NEW packages will be installed:  
  libauthen-sasl-perl libdata-dump-perl libencode-locale-perl libfile-listing-perl libfont-afm-perl libhtml-form-perl  
  libhtml-format-perl libhtml-parser-perl libhtml-tagset-perl libhtml-tree-perl libhttp-cookies-perl libhttp-daemon-pe  
  libhttp-date-perl libhttp-message-perl libhttp-negotiate-perl libio-html-perl libio-socket-ssl-perl liblwp-mediatype  
  liblwp-protocol-https-perl libmailtools-perl libnet-http-perl libnet-smtp-ssl-perl libnet-ssleay-perl libtimedate-pe  
  libtry-tiny-perl liburi-perl libwww-perl libwww-robotrules-perl libxml-parser-perl perl-openssl-defaults
```

```
sudo cpan install XML::Parser::PerlSAX
```

```
(myenv) dabanoglu19588@gcloud:~/final_project$ sudo cpan install XML::Parser::PerlSAX  
Loading internal logger. Log::Log4perl recommended for better logging  
  
CPAN.pm requires configuration, but most of it can be done automatically.  
If you answer 'no' below, you will enter an interactive dialog for each
```

```
sudo cpan install XML::RegExp
```

```
(myenv) dabanoglu19588@gcloud:~/final_project/RELEASE-1.5.5$ sudo cpan install XML::RegExp
Loading internal logger. Log::Log4perl recommended for better logging
Reading '/root/.cpan/Metadata'
  Database was generated on Wed, 06 Apr 2022 10:41:03 GMT
Running install for module 'XML::RegExp'
Fetching with LWP:
http://www.cpan.org/authors/id/T/TJ/TJMATHER/XML-RegExp-0.04.tar.gz
Fetching with LWP:
http://www.cpan.org/authors/id/T/TJ/TJMATHER/CHECKSUMS
Checksum for /root/.cpan/sources/authors/id/T/TJ/TJMATHER/XML-RegExp-0.04.tar.gz ok
'YAML' not installed, will not store persistent state
Configuring T/TJ/TJMATHER/XML-RegExp-0.04.tar.gz with Makefile.PL
Checking if your kit is complete...
Looks good
Generating a Unix-style Makefile
Writing Makefile for XML::RegExp
Writing MYMETA.yml and MYMETA.json
  TJMATHER/XML-RegExp-0.04.tar.gz
  /usr/bin/perl Makefile.PL INSTALLDIRS=site -- OK
Running make for T/TJ/TJMATHER/XML-ReqExp-0.04.tar.qz
```

## sudo cpan install XML::DOM

```
(myenv) dabanoglu19588@gcloud:~/final_project/RELEASE-1.5.5$ sudo cpan install XML::DOM
Loading internal logger. Log::Log4perl recommended for better logging
Reading '/root/.cpan/Metadata'
  Database was generated on Wed, 06 Apr 2022 10:41:03 GMT
Running install for module 'XML::DOM'
Fetching with LWP:
http://www.cpan.org/authors/id/T/TJ/TJMATHER/XML-DOM-1.46.tar.gz
Checksum for /root/.cpan/sources/authors/id/T/TJ/TJMATHER/XML-DOM-1.46.tar.gz ok
'YAML' not installed, will not store persistent state
Configuring T/TJ/TJMATHER/XML-DOM-1.46.tar.gz with Makefile.PL
Checking if your kit is complete...
Looks good
Running make for T/TJ/TJMATHER/XML-DOM-1.46.tar.gz
```

Run following to check if ROUGE is successfully installed

## ./runROUGE-test.pl

```
(myenv) dabanoglu19588@gcloud:~/final_project/RELEASE-1.5.5$ ./runROUGE-test.pl
../ROUGE-1.5.5.pl -e ../data -c 95 -2 -1 -U -r 1000 -n 4 -w 1.2 -a ROUGE-test.xml > ../sample-output/ROUGE-test-c95-2-1-U-r1000-n4-w1.2-a.out
../ROUGE-1.5.5.pl -e ../data -c 95 -2 -1 -U -r 1000 -n 4 -w 1.2 -a -m ROUGE-test.xml > ../sample-output/ROUGE-test-c95-2-1-U-r1000-n4-w1.2-a-m.out
../ROUGE-1.5.5.pl -e ../data -c 95 -2 -1 -U -r 1000 -n 4 -w 1.2 -a -m -s ROUGE-test.xml > ../sample-output/ROUGE-test-c95-2-1-U-r1000-n4-w1.2-a-m-s.out
../ROUGE-1.5.5.pl -e ../data -c 95 -2 -1 -U -r 1000 -n 4 -w 1.2 -l 10 -a ROUGE-test.xml > ../sample-output/ROUGE-test-c95-2-1-U-r1000-n4-w1.2-l10-a.out
../ROUGE-1.5.5.pl -e ../data -c 95 -2 -1 -U -r 1000 -n 4 -w 1.2 -l 10 -a -m ROUGE-test.xml > ../sample-output/ROUGE-test-c95-2-1-U-r1000-n4-w1.2-l10-a-m.out
../ROUGE-1.5.5.pl -e ../data -c 95 -2 -1 -U -r 1000 -n 4 -w 1.2 -l 10 -a -m -s ROUGE-test.xml > ../sample-output/ROUGE-test-c95-2-1-U-r1000-n4-w1.2-l10-a-m-s.out
../ROUGE-1.5.5.pl -e ../data -c 95 -2 -1 -U -r 1000 -n 4 -w 1.2 -b 75 -a ROUGE-test.xml > ../sample-output/ROUGE-test-c95-2-1-U-r1000-n4-w1.2-b75-a.out
../ROUGE-1.5.5.pl -e ../data -c 95 -2 -1 -U -r 1000 -n 4 -w 1.2 -b 75 -a -m ROUGE-test.xml > ../sample-output/ROUGE-test-c95-2-1-U-r1000-n4-w1.2-b75-a-m.out
../ROUGE-1.5.5.pl -e ../data -c 95 -2 -1 -U -r 1000 -n 4 -w 1.2 -b 75 -a -m -s ROUGE-test.xml > ../sample-output/ROUGE-test-c95-2-1-U-r1000-n4-w1.2-b75-a-m-s.out
../ROUGE-1.5.5.pl -e ../data -3 HM -z SIMPLE DUC2002-BE-F.in.26.lst 26 > ../sample-output/UC2002-BE-F.in.26.lst.out
../ROUGE-1.5.5.pl -e ../data -3 HM DUC2002-BE-F.in.26.simple.xml 26 > ../sample-output/UC2002-BE-F.in.26.simple.out
../ROUGE-1.5.5.pl -e ../data -3 HM -z SIMPLE DUC2002-BE-L.in.26.lst 26 > ../sample-output/UC2002-BE-L.in.26.lst.out
../ROUGE-1.5.5.pl -e ../data -3 HM DUC2002-BE-L.in.26.simple.xml 26 > ../sample-output/UC2002-BE-L.in.26.simple.out
../ROUGE-1.5.5.pl -e ../data -n 4 -z SPL DUC2002-ROUGE.in.26.spl.lst 26 > ../sample-output/UC2002-ROUGE.in.26.spl.lst.out
../ROUGE-1.5.5.pl -e ../data -n 4 DUC2002-ROUGE.in.26.spl.xml 26 > ../sample-output/UC2002-ROUGE.in.26.spl.out
```

## 4. Installing pyrouge

```
git clone https://github.com/bheinzerling/pyrouge.git
```

```
cd pyrouge
```



```
(myenv) dabanoglu19588@gcloud:~/final_project/RELEASE-1.5.5$ git clone https://github.com/bheinzerling/pyrouge.git
Cloning into 'pyrouge'...
remote: Enumerating objects: 551, done.
remote: Total 551 (delta 0), reused 0 (delta 0), pack-reused 551
Receiving objects: 100% (551/551), 123.17 KiB | 1.64 MiB/s, done.
Resolving deltas: 100% (198/198), done.
(myenv) dabanoglu19588@gcloud:~/final_project/RELEASE-1.5.5$ cd pyrouge
(myenv) dabanoglu19588@gcloud:~/final_project/RELEASE-1.5.5/pyrouge$
```

**pip install -e .**

```
(myenv) dabanoglu19588@gcloud:~/final_project/RELEASE-1.5.5/pyrouge$ pip install -e .
Obtaining file:///home/dabanoglu19588/final_project/RELEASE-1.5.5/pyrouge
Installing collected packages: pyrouge
  Running setup.py develop for pyrouge
Successfully installed pyrouge-0.1.3
(myenv) dabanoglu19588@gcloud:~/final_project/RELEASE-1.5.5/pyrouge$
```

## 5. Installing pytorch

**conda install pytorch-cpu==1.1.0 torchvision-cpu==0.3.0 cpuonly -c pytorch**

```
Proceed ([y]/n)? y

Downloading and Extracting Packages
blas-1.0 | 6 KB | ##### | 100%
pillow-8.3.1 | 637 KB | ##### | 100%
openjpeg-2.4.0 | 331 KB | ##### | 100%
numpy-1.19.2 | 22 KB | ##### | 100%
libtiff-4.2.0 | 502 KB | ##### | 100%
lz4-c-1.9.3 | 185 KB | ##### | 100%
ninja-1.10.2 | 1.5 MB | ##### | 100%
certifi-2021.5.30 | 139 KB | ##### | 100%
libwebp-base-1.2.2 | 440 KB | ##### | 100%
torchvision-cpu-0.3.0 | 3.8 MB | ##### | 100%
cpuonly-2.0 | 2 KB | ##### | 100%
mkl-service-2.3.0 | 52 KB | ##### | 100%
mkl-2020.2 | 138.3 MB | ##### | 100%
jpeg-9d | 232 KB | ##### | 100%
mkl_fft-1.3.0 | 170 KB | ##### | 100%
mkl_random-1.1.1 | 327 KB | ##### | 100%
intel-openmp-2022.0 | 4.2 MB | ##### | 100%
cffi-1.14.6 | 220 KB | ##### | 100%
freetype-2.11.0 | 618 KB | ##### | 100%
six-1.16.0 | 18 KB | ##### | 100%
zstd-1.4.9 | 480 KB | ##### | 100%
lcms2-2.12 | 312 KB | ##### | 100%
pytorch-cpu-1.1.0 | 53.5 MB | ##### | 100%
libpng-1.6.37 | 278 KB | ##### | 100%
numpy-base-1.19.2 | 4.1 MB | ##### | 100%
olefile-0.46 | 48 KB | ##### | 100%
pytorch-mutex-1.0 | 3 KB | ##### | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
```

## 5. Downloading the pretrained-bert-model into final\_project folder

**wget**

**<https://s3.amazonaws.com/models.huggingface.co/bert/bert-large-uncased.tar.gz>**

```
(myenv) dabanoglu19588@gcloud:~/final_project/RELEASE-1.5.5$ cd ..
(myenv) dabanoglu19588@gcloud:~/final_project$ wget https://s3.amazonaws.com/models.huggingface.co/bert/bert-large-uncased.tar.gz
--2022-04-06 11:16:26-- https://s3.amazonaws.com/models.huggingface.co/bert/bert-large-uncased.tar.gz
Resolving s3.amazonaws.com (s3.amazonaws.com)... 52.217.44.102
Connecting to s3.amazonaws.com (s3.amazonaws.com)|52.217.44.102|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1248501532 (1.2G) [application/x-tar]
Saving to: 'bert-large-uncased.tar.gz'

bert-large-uncased.tar.gz 100%[=====>] 1.16G 39.4MB/s in 29s

2022-04-06 11:16:56 (41.4 MB/s) - 'bert-large-uncased.tar.gz' saved [1248501532/1248501532]
```

Go to summarizer folder and change BertParent.py file like below:

```
self.model =  
BertModel.from_pretrained('YOUR_PROJECT_Directory/bert-largeuncased.ta  
r.gz')
```

```
cd summarizer
```

```
vim BertParent.py
```

```
def __init__(self, model_type: str, size: str):  
    #self.model = self.model_handler[model_type].from_pretrained(self.size_handler[size][model_type])  
    self.model = BertModel.from_pretrained('final_project/bert-large-uncased.tar.gz')  
    self.tokenizer = self.token_handler[model_type].from_pretrained(self.size_handler[size][model_type])  
    self.vector_size = self.vector_handler[size][model_type]  
    self.model_type = model_type  
    self.model.eval()
```

## 6. Installing required packages

```
pip3 install flask
```

```
pip3 install pandas
```

```
pip3 install sklearn
```

```
pip3 install nltk
```

```
pip3 install gensim==3.8.3
```

```
pip3 install pytorch-pretrained-bert
```

```
pip3 install matplotlib==3.0.0
```

## Download punkt package with nltk

```
python3
```

```
import nltk  
nltk.download('punkt')  
exit()
```

```
(myenv) dabanoglu19588@gcloud:~/final_project/summarizer$ python3  
Python 3.6.13 |Anaconda, Inc.| (default, Jun  4 2021, 14:25:59)  
[GCC 7.5.0] on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>> import nltk  
>>> nltk.download('punkt')  
[nltk_data] Downloading package punkt to  
[nltk_data]      /home/dabanoglu19588/nltk_data...  
[nltk_data]   Unzipping tokenizers/punkt.zip.  
True  
>>> exit()  
(myenv) dabanoglu19588@gcloud:~/final_project/summarizer$
```

## 7. Running the application

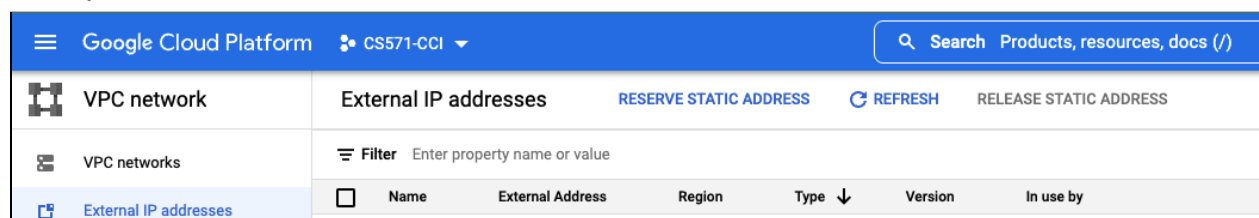
**python3 app.py**

```
(myenv) dabanoglu19588@gcloud:~/final_project$ python3 app.py
* Serving Flask app 'app' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
WARNING:werkzeug: * Running on all addresses.
  WARNING: This is a development server. Do not use it in a production deployment.
INFO:werkzeug: * Running on http://10.128.0.28:5000/ (Press CTRL+C to quit)
INFO:werkzeug: * Restarting with stat
WARNING:werkzeug: * Debugger is active!
```

As you can see it is running. Now we need to access this service from everywhere. We need to reserve a static IP to this virtual machine and open a port.

## 8. Reserving an Static IP address and Opening a Port

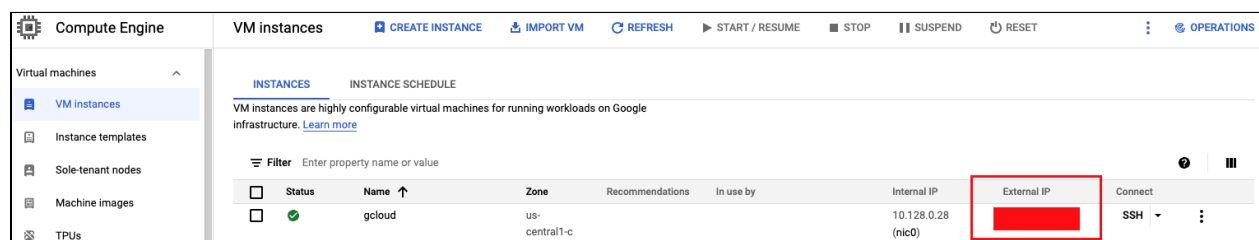
Go to VPC network -> External IP addresses -> click RESERVE STATIC ADDRESS on the top menu



I reserved an address with configuration below

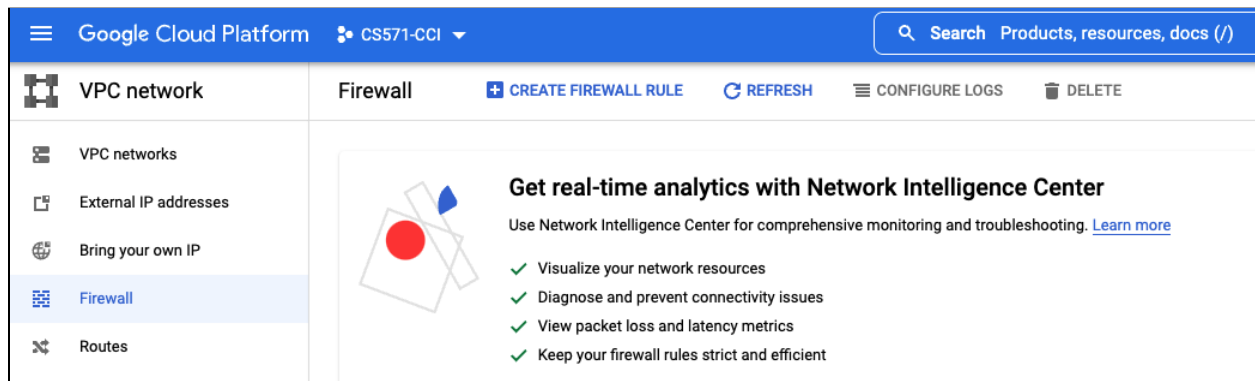
1. Enter a name for the address
2. Network Service Tier : Standart
3. IP version : IPv4
4. Type: Regional

After these steps GCP will assign your device a static IP. You can see that on VM instances page



Second step is open a port by Creating a firewall rule

Go to VPC network -> Firewall -> click CREATE FIREWALL RULE on the top menu



The application is running on port number 5000 so I use following configuration:

### Create a firewall rule

Firewall rules control incoming or outgoing traffic to an instance. By default, incoming traffic from outside your network is blocked. [Learn more](#)

Name \*  
nlp  
Lowercase letters, numbers, hyphens allowed

Description

Logs  
Turning on firewall logs can generate a large number of logs which can increase costs in Cloud Logging. [Learn more](#)  
☐ On  
☒ Off

Network \*  
default

Priority \*  
1000 [CHECK PRIORITY OF OTHER FIREWALL RULES](#)  
Priority can be 0 - 65535

Direction of traffic  
☒ Ingress  
☐ Egress

Action on match  
☒ Allow  
☐ Deny

Targets  
Specified service account

Service account scope  
☒ In this project  
☐ In another project

Target service account  
Compute Engine default service account

Source filter  
IPv4 ranges

Source IPv4 ranges \*  
0.0.0.0/0 for example, 0.0.0.0/0, 192.168.2.0/24

Second source filter  
None

Protocols and ports  
☐ Allow all  
☒ Specified protocols and ports

☒ tcp : 5000  
☐ udp : all  
☐ Other protocols  
protocols, comma separated, e.g. ah, sctp

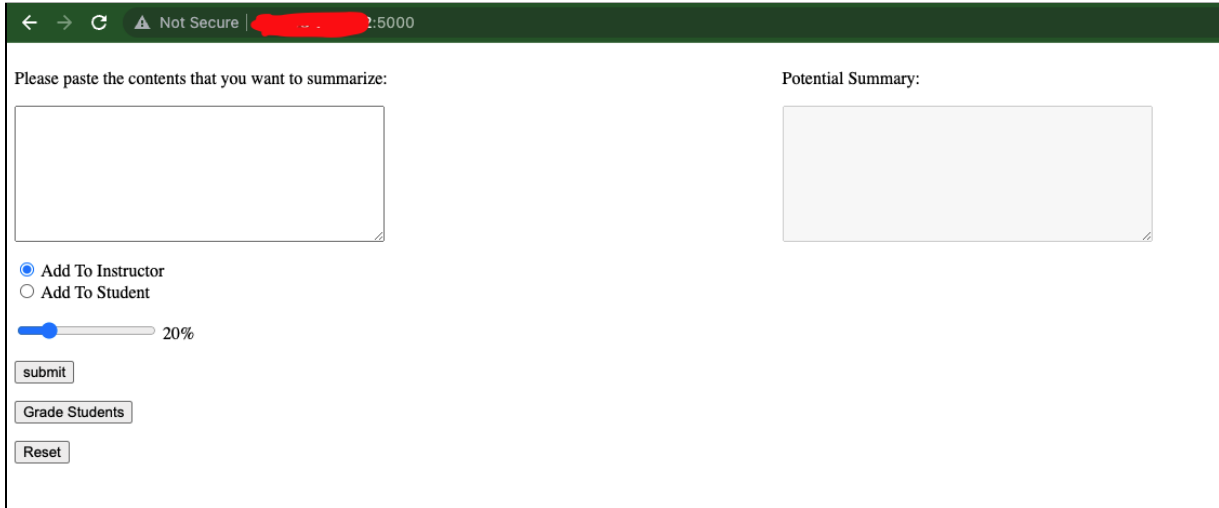
[DISABLE RULE](#)

[CREATE](#) [CANCEL](#)

Click create to create a firewall rule.

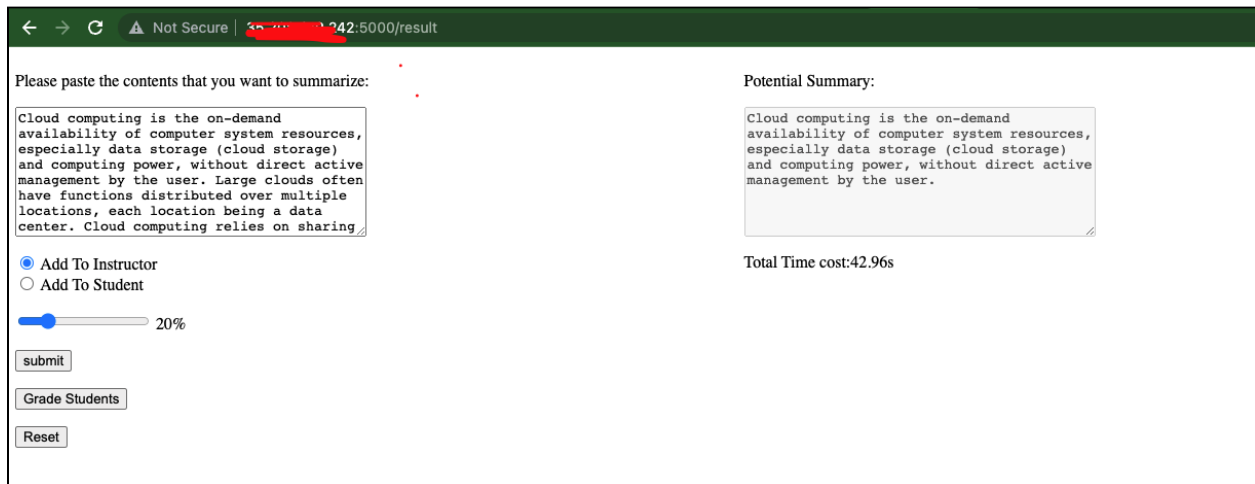
## 9. Accessing the service.

Go to another browser and enter following address : <Your-VM-StaticIP:5000->



The screenshot shows a web browser window with the address bar displaying a redacted IP address followed by :5000. The page has a dark green header. The main content area is divided into two columns. The left column contains a text input field with the placeholder text "Please paste the contents that you want to summarize:". Below the input field are two radio buttons: "Add To Instructor" (selected) and "Add To Student". A slider control is set to 20%. Below the slider are three buttons: "submit", "Grade Students", and "Reset". The right column contains a text area with the placeholder text "Potential Summary:".

Enter an article to summarize:



The screenshot shows the same web application interface, but now the text input field contains a paragraph about cloud computing. The text area on the right now displays a summarized version of the text. The address bar shows the URL as a redacted IP address followed by :5000/result. The "Total Time cost:42.96s" is displayed below the text area. The "submit" button is now disabled.

If everything goes well, the app is running on the VM.