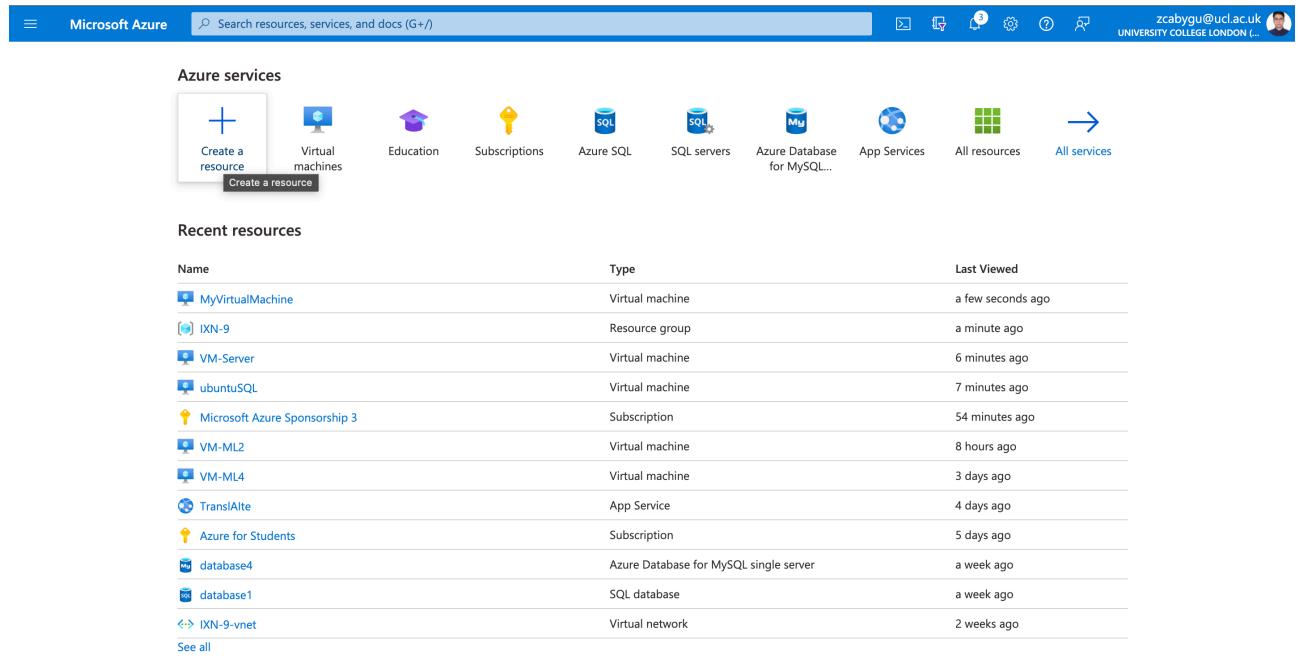


Deployment Guide of the Transl'AI'te Web Application

Step 1. Create an Azure Virtual Machine

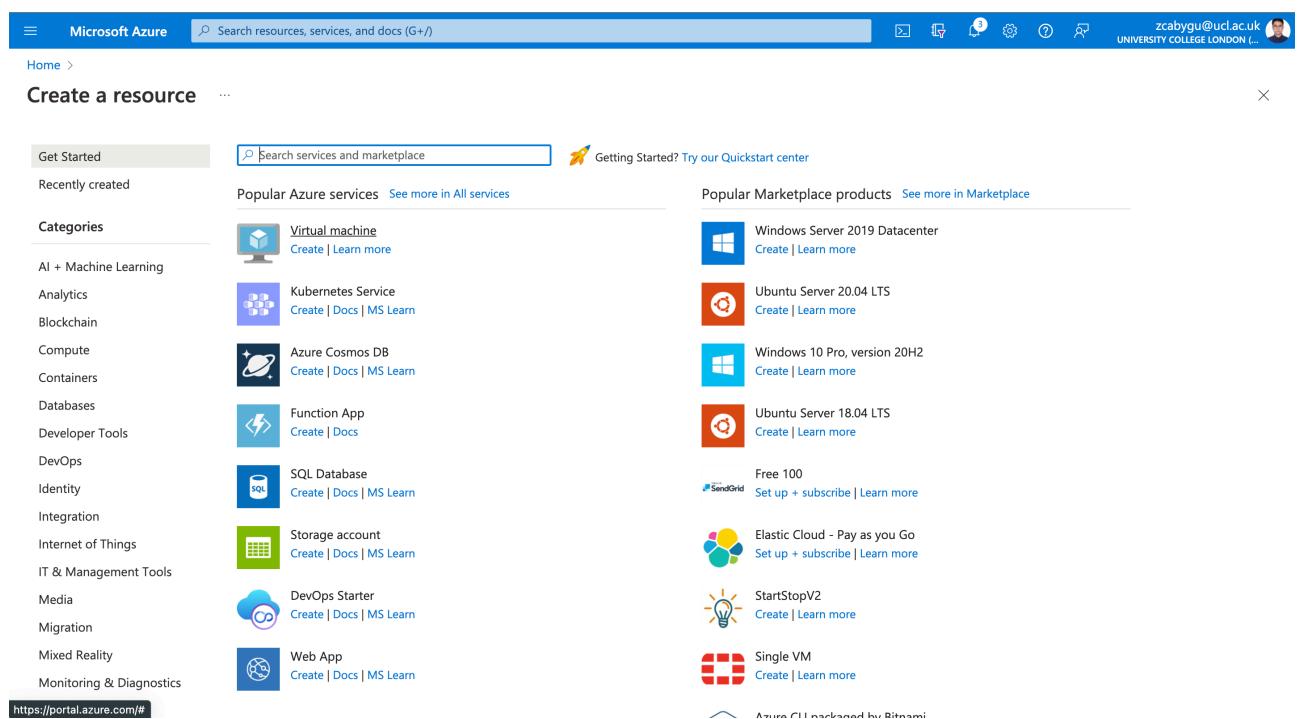
1. Login to your Microsoft Azure Portal.



The screenshot shows the Microsoft Azure portal interface. At the top, there's a search bar and a user profile. Below the header, the 'Azure services' section features a 'Create a resource' button, which is highlighted with a red box. Other service icons include Virtual machines, Education, Subscriptions, Azure SQL, SQL servers, Azure Database for MySQL, App Services, All resources, and All services. The 'Recent resources' section lists various Azure resources like MyVirtualMachine, IXN-9, VM-Server, etc., with their types, last viewed times, and links. A link to 'See all' is at the bottom of this section.

<https://portal.azure.com/#create/hub?subscriptionId=640a5079-21c8-4dc9-9db1-a3b7fc76afcd>

2. Click the 'Create a resource' option under Azure services.



The screenshot shows the 'Create a resource' page. On the left, there's a sidebar with 'Get Started' and 'Recently created' sections, followed by a 'Categories' tree view. The main area has a search bar and a 'Getting Started? Try our Quickstart center' link. It's divided into 'Popular Azure services' and 'Popular Marketplace products'. Under 'Popular Azure services', there are cards for Virtual machine, Kubernetes Service, Azure Cosmos DB, Function App, SQL Database, Storage account, DevOps Starter, and Web App. Under 'Popular Marketplace products', there are cards for Windows Server 2019 Datacenter, Ubuntu Server 20.04 LTS, Windows 10 Pro, Ubuntu Server 18.04 LTS, SendGrid, Elastic Cloud - Pay as you Go, StartStopV2, and Single VM. Each card includes a 'Create' or 'Get started' button and a 'Learn more' link. A note at the bottom says 'Azure CLI packaged by Bitnami'.

<https://portal.azure.com/#>

3. Type **virtual machines** in the search. Under Services, select Virtual machines.

4. In the Virtual machines page, select Create and then Virtual machine. The Create a virtual machine page opens.

The screenshot shows the 'Create a virtual machine' page in the Microsoft Azure portal. The 'Basics' tab is selected. In the 'Project details' section, the subscription is set to 'Microsoft Azure Sponsorship 3' and the resource group is 'IXN-9'. In the 'Instance details' section, the virtual machine name is 'MyVirtualMachine', the region is '(Europe) UK South', and the image is 'Ubuntu Server 20.04 LTS - Gen2'. The 'Review + create' button is at the bottom left, and navigation buttons '< Previous' and 'Next : Disks >' are at the bottom right.

5. In the **Basics** tab, under **Project details**, make sure the correct subscription is selected and then choose to **Create new** resource group.

Type *myResourceGroup* for the name.*.

6. Under **Instance details**, type *myVM* for the **Virtual machine name**, and choose *Ubuntu 18.04 LTS - Gen2* for your **Image**. Leave the other defaults. The default size and pricing is only shown as an example. Size availability and pricing are dependent on your region and subscription.

7. Under **Administrator account**, select **Password**.

8. Enter your desired **Username** and **Password** for the virtual machines. If portal rejects your username and password, retry according to its rules until it is accepted. Remember the username and the password you entered.

9. Under Inbound port rules > Public inbound ports, choose Allow selected ports and then select SSH (22), HTTP (80), and HTTPS (443) from the drop-down.

Microsoft Azure Search resources, services, and docs (G+)

Home > Virtual machines >

Create a virtual machine

Size * Standard_DS1_v2 - 1 vcpu, 3.5 GiB memory (£47.77/month) [See all sizes](#)

Administrator account

Authentication type Password

Username * User_1

Password * [View](#)

Confirm password * [View](#)

Inbound port rules

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

Public inbound ports * Allow selected ports

Select inbound ports * HTTP (80), HTTPS (443), SSH (22)

Warning: This will allow all IP addresses to access your virtual machine. This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.

Review + create < Previous Next : Disks >

10. Leave the remaining defaults and then select the **Review + create** button at the bottom of the page.

11. On the **Create a virtual machine** page, you can see the details about the VM you are about to create. When you are ready, select **Create**.

Microsoft Azure Search resources, services, and docs (G+)

Home > Virtual machines >

Create a virtual machine

Validation passed

Basics **Disks** **Networking** **Management** **Advanced** **Tags** **Review + create**

PRODUCT DETAILS

1 X Standard DS1 v2 by Microsoft [Subscription credits apply](#) 0.0654 GBP/hr [Pricing for other VM sizes](#)

TERMS

By clicking "Create", I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace offering(s) listed above; (b) authorize Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. See the [Azure Marketplace Terms](#) for additional details.

Name Yufei Gu

Preferred e-mail address * yufei.gu.20@ucl.ac.uk

Preferred phone number * 07529968601

Create < Previous Next > Download a template for automation

12. Wait for the deployment of your virtual machine.

Deployment name: CreateVm-canonical.0001-com-ubuntu-server-focal-2-20220329212135 | Overview

Deployment details:

Resource	Type	Status	Operation details
MyVirtualMachine	Microsoft.Compute/virtualMachines	Created	Operation details
myvirtualmachine19	Microsoft.Network/networkInterfaces	Created	Operation details
MyVirtualMachine-ip	Microsoft.Network/publicIPAddresses	OK	Operation details
MyVirtualMachine-nsg	Microsoft.Network/networkSecurityGroups	OK	Operation details

13. When the deployment is finished, select **Go to resource**.

Your deployment is complete

Deployment details:

Resource	Type	Status	Operation details
MyVirtualMachine	Microsoft.Compute/virtualMachines	Created	Operation details
myvirtualmachine19	Microsoft.Network/networkInterfaces	Created	Operation details
MyVirtualMachine-ip	Microsoft.Network/publicIPAddresses	OK	Operation details
MyVirtualMachine-nsg	Microsoft.Network/networkSecurityGroups	OK	Operation details

Next steps:

- Setup auto-shutdown Recommended
- Monitor VM health, performance and network dependencies Recommended
- Run a script inside the virtual machine Recommended

[Go to resource](#) [Create another VM](#)

Step 2. Connect to your virtual machine

1. Open the Cloud Shell in the top-right corner in your Azure Portal.

MyVirtualMachine

Cloud Shell

2. Connect to your virtual machine via SSH with client.

Type the following command in the Cloud Sheel:

'ssh <username>@<Public IP address>'

Replace the <username> with the username you entered when you create the virtual machine.

Replace the <Public IP address> with the Public IP address of your azure virtual machine. You can find it in the position as the picture shows below: The Public IP address of this sample virtual machine is 20.117.101.114, the third row of the second column. Did you find it?

3. After you typed the correct ssh command, it will ask you for the password of your virtual machine. (If you are first connecting to the virtual machine, it will ask you if you really want to connect to it with a (yes/no) option. Type yes to continue)

After you entered the correct password, you can connect to your virtual machine. The head of your command will change to <username>@<name of your virtual machine> as shown below. This is a proof of you successfully connecting to your virtual machine. Congratulations!

Step 2.5. Set inbound port networking rule

4. Enter the Networking page of your virtual machine on azure portal by clicking the left catalog.

Microsoft Azure Search resources, services, and docs (G+)

Home > MyVirtualMachine

MyVirtualMachine | Networking

Virtual machine

Search (Cmd+)/ Attach network interface Detach network interface Feedback

myvirtualmachine19

IP configuration ipconfig1 (Primary)

Network Interface: myvirtualmachine19 Effective security rules Troubleshoot VM connection issues Topology

Virtual network/subnet: IXN-9-vnet/default NIC Public IP: 20.117.101.114 NIC Private IP: 10.1.0.5 Accelerated networking: Enabled

Inbound port rules Outbound port rules Application security groups Load balancing

Network security group MyVirtualMachine-nsg (attached to network interface: myvirtualmachine19)
Impacts 0 subnets, 1 network interfaces Add inbound port rule

Priority	Name	Port	Protocol	Source	Destination	Action	...
300	SSH	22	TCP	Any	Any	Allow	...
320	HTTPS	443	TCP	Any	Any	Allow	...
340	HTTP	80	TCP	Any	Any	Allow	...
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow	...
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	Allow	...
65500	DenyAllInBound	Any	Any	Any	Any	Deny	...

5. Click the '**Add inbound port rule**' blue button on the left hand side.
6. Set the Destination port ranges to 8080 (usually it is the default) and left other options unchanged. Click **Add** and finished.

The screenshot shows the Microsoft Azure portal interface. On the left, there's a sidebar with navigation links like Home, MyVirtualMachine, Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Networking, Connect, Disks, Size, Security, Advisor recommendations, Extensions + applications, Continuous delivery, Availability + scaling, Configuration, and Identity. The 'Networking' option is selected. In the main content area, under 'MyVirtualMachine | Networking', there's a table for 'Inbound port rules'. The table has columns: Priority, Name, Port, Protocol, and Action. It lists several rules, including SSH (Priority 300, Port 22, TCP, Allow), HTTPS (Priority 320, Port 443, TCP, Allow), HTTP (Priority 340, Port 80, TCP, Allow), and three system-defined rules (Priority 65000-65500, Action Allow). To the right of the table, a modal window titled 'Add inbound security rule' is open. It has fields for Source (set to 'Any'), Source port ranges (set to '*'), Destination (set to 'Any'), Destination port ranges (set to '8080'), Protocol (set to 'Any'), and Action (set to 'Allow'). At the bottom of the modal are 'Add' and 'Cancel' buttons.

Step 3. Download the web application package

1. Clone the web application package
 - Enter command : '**git clone https://github.com/Yufei-Gu-451/IXN-Team-9.git**'
2. Clone the Bert module into the app directory
 - Enter command : '**git clone https://github.com/google-research/bert.git IXN-Team-9/app/bert**'
3. Download a pre-trained BioBERT model from <https://github.com/naver/biobert-pretrained.git> to your computer. The first model is used as a sample **BioBERT-Base v1.1 (+ PubMed 1M)** - based on BERT-base-Cased (same vocabulary).
4. Decompress the downloaded pre-trained BioBERT model file on your computer.

BioBERT Pre-trained Weights

This repository provides pre-trained weights of BioBERT, a language representation model for biomedical domain, especially designed for biomedical text mining tasks such as biomedical named entity recognition, relation extraction, question answering, etc. Please refer to our paper [BioBERT: a pre-trained biomedical language representation model for biomedical text mining](#) for more details.

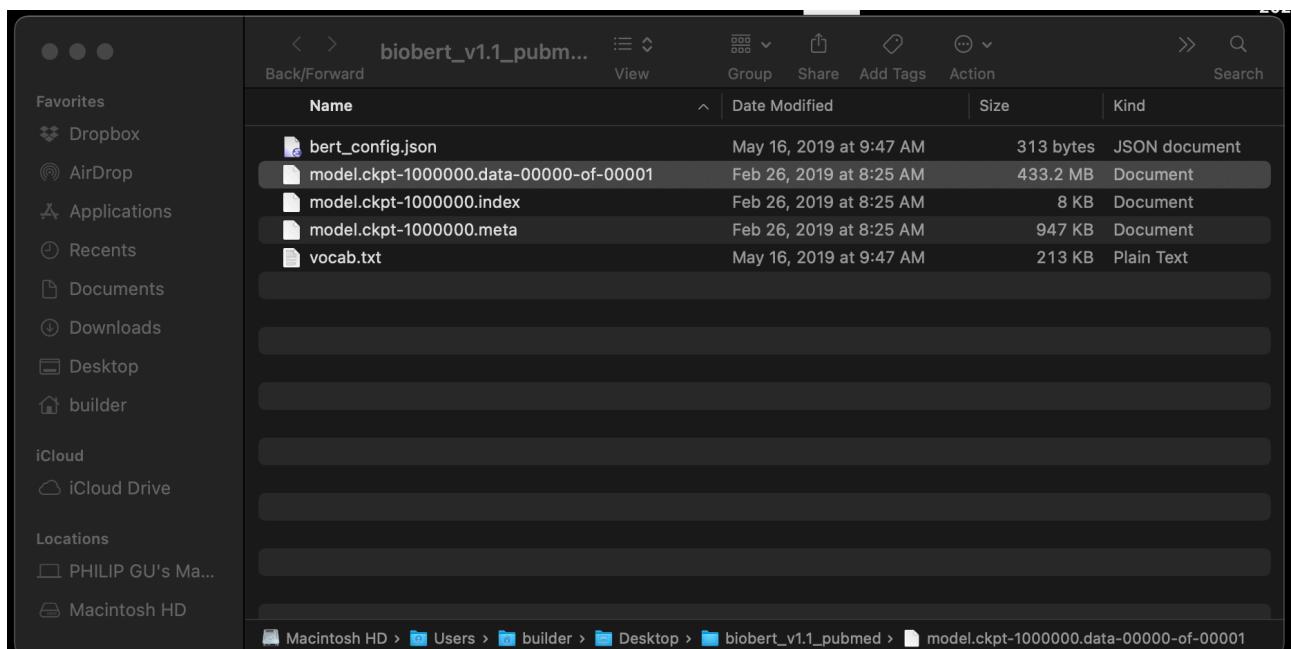
Downloading pre-trained weights

Go to [releases](#) section of this repository or click links below to download pre-trained weights of BioBERT. We provide three combinations of pre-trained weights: BioBERT (+ PubMed), BioBERT (+ PMC), and BioBERT (+ PubMed + PMC). Pre-training was based on the [original BERT code](#) provided by Google, and training details are described in our paper. Currently available versions of pre-trained weights are as follows:

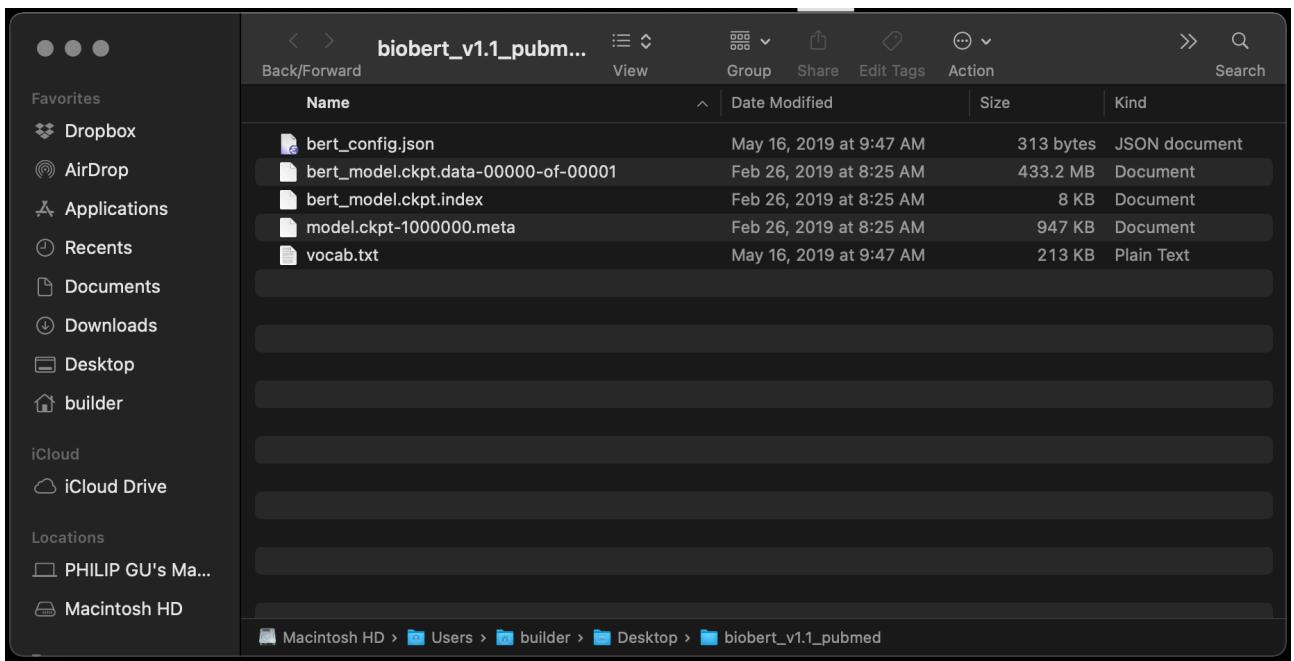
- [BioBERT-Base v1.1 \(+ PubMed 1M\)](#) - based on BERT-base-Cased (same vocabulary)
- [BioBERT-Large v1.1 \(+ PubMed 1M\)](#) - based on BERT-large-Cased (custom 30k vocabulary), [NER/QA Results](#)
- [BioBERT-Base v1.0 \(+ PubMed 200K\)](#) - based on BERT-base-Cased (same vocabulary)
- [BioBERT-Base v1.0 \(+ PMC 270K\)](#) - based on BERT-base-Cased (same vocabulary)
- [BioBERT-Base v1.0 \(+ PubMed 200K + PMC 270K\)](#) - based on BERT-base-Cased (same vocabulary)

Make sure to specify the versions of pre-trained weights used in your works. If you have difficulty choosing which one to use, we recommend using [BioBERT-Base v1.1 \(+ PubMed 1M\)](#) or [BioBERT-Large v1.1 \(+ PubMed 1M\)](#) depending on your GPU resources. Note that for BioBERT-Base, we are using WordPiece vocabulary ([vocab.txt](#)) provided by Google as any new words in biomedical corpus can be represented with subwords (for instance, Leukemia => Leu + ##ke + ##mia). More details are in the closed [issue #1](#).

5. Change the name of 'model.ckpt_100001.*' to 'bert_model.ckpt.*', and copy the files to the BERT directory.



The original file you get from download



The files after renamed

6. Sent the pre-trained BioBERT model to the virtual machine you just created.

- Open the terminal on your computer
- Type in command ‘`scp <Default Download Directory>/biobert_v1.1_pubmed/bert_config.json <VM username>@<VM Public IP address>:IXN-Team-9/app/bert`’
- Type in command ‘`scp <Default Download Directory>/biobert_v1.1_pubmed/bert_model.ckpt.data-00000-of-00001 <VM username>@<VM Public IP address>:IXN-Team-9/app/bert`’
- Type in command ‘`scp <Default Download Directory>/biobert_v1.1_pubmed/bert_model.ckpt.index <VM username>@<VM Public IP address>:IXN-Team-9/app/bert`’
- Type in command ‘`scp <Default Download Directory>/biobert_v1.1_pubmed/bert_model.ckpt.meta <VM username>@<VM Public IP address>:IXN-Team-9/app/bert`’
- Type in command ‘`scp <Default Download Directory>/biobert_v1.1_pubmed/vocab.txt <VM username>@<VM Public IP address>:IXN-Team-9/app/bert`’

Hints to the top five commands:

- (When it asks for password, enter the password of your virtual machine)
- Replace **<Default Download Directory>** to the directory of your BioBERT model package.
- Replace **<VM username>** with the username you set when you create the virtual machine.

- Replace <VM Public IP address> with the public IP address of your virtual machine.
- If you have downloaded a different BioBERT model, replace <**biobert_v1.1_pubmed**> to the exact file name of the BioBERT model.

Step 4. Set up the python environment and deploy the web app

1. Download the python3.7 dev package on your virtual machine.

- Enter command : ‘**sudo apt install software-properties-common**’
- Enter command : ‘**sudo add-apt-repository ppa:deadsnakes/ppa**’ (Press enter as hinted during installation)
- Enter command : ‘**sudo apt-get install python3.7-dev**’

2. Install pip for python 3.7 package

- Enter command : ‘**sudo apt install python3.7 python3-pip**’ (Press Y as hinted during installation)

3. Install the mysql server and the unixodbc driver

- Enter command : ‘**sudo apt-get install mysql-client**’ (Press Y as hinted during installation)
- Enter command : ‘**sudo apt install unixodbc-dev**’ (Press Y as hinted during installation)
- Enter command : ‘**sudo apt install libmysqlclient-dev**’ (Press Y as hinted during installation)

4. Install all required package for the web application

- Enter command : ‘**sudo python3.7 -m pip install -r IXN-Team-9/requirements.txt --no-cache-dir**’

5. Download the required NLTK package

- Enter command : ‘**python3.7 IXN-Team-9/nltkConfig.py**’

6. Run the application

- Enter command : ‘**python3.7 IXN-Team-9/application.py**’

If everything is successful, you will see the following text, representing the deployment of your web application is now successful!

```
User_1@MyVirtualMachine:~/IXN-Team-9$ python3.7 application.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
 * Running on all addresses.
   WARNING: This is a development server. Do not use it in a production deployment.
 * Running on http://10.1.0.5:8080/ (Press CTRL+C to quit)
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 169-088-158
144.82.9.234 - - [29/Mar/2022 22:18:38] "GET / HTTP/1.1" 200 -
144.82.9.234 - - [29/Mar/2022 22:18:39] "GET /static/favicon.ico HTTP/1.1" 200 -
144.82.9.234 - - [29/Mar/2022 22:18:39] "GET /static/favicon.ico HTTP/1.1" 304 -
144.82.9.234 - - [29/Mar/2022 22:18:43] "GET /auth/login HTTP/1.1" 200 -
```

You can now use the IP address of your virtual machine and the port number <Public IP address>:8080 to access the web application from any computer!