Instructions for User access to Tartu University OpenAI Service.

General access to the OpenAl Service is granted through portal.azure.com. In there a user should have access to **TU-OpenAl-LLM-Department** Resource group through one of two possible roles:

- **Contributor** user with the ability to deploy OpenAI models and set configurations as well as fine tune models on 'training' data sets
- Reader user with the ability to only interact with OpenAI models by asking for completions or embeddings

Most users for a resource group should fall under the second category with only select **Contributor** users that manage the department workspace

To access the OpenAI user interface we need to navigate to **portal.azure.com** in our browser (log in with our university credentials) and from there we navigate to **Resource Groups**:

≡ Microsoft Azure	P Sear	:h resources, service	s, and docs (G+/)						
Azure se	rvices	4	ß	~		(4)		_	
Create	Resource groups	Subscriptions	Azure OpenAl	API Management	Azure Active Directory	Management groups	Key vaults	Storage accounts	More services
Resource	es								
Recent	Favorite								

Afterwards we select the resource group with our department name (e.g., **TU-OpenAI-LLM-CompSc**). In the resource group we have two resources:

- **OpenAl** which is our user interface for deploying, fine tuning and interacting with GPT models
- Key Vault which is used to store API keys to access the deployed OpenAI models for application development end experiments.

	<u>م</u>	Search resources, services, and docs (G+/)			
Home > Resource groups >					
Resource groups « Tartu Ülikool (ut.ee)	(intersection of the second se	A #			
🕂 Create 🍪 Manage view 🗸 \cdots	₽ Search «	🕂 Create 🛞 Manage view 🗸 📋 Delete resource group 🖒 Refresh 🞍 Export to CSV 😽 Ope	n query $\mid \ \oslash$ Assign tags $ ightarrow$ Move $ ightarrow$		
Filter for any field	(Overview	∧ Essentials			
Name 1	Activity log	Subscription (move) : Azure OpenAl	Deployments : 2 Succeeded		
LogAnalyticsDefaultResources	Access control (IAM)	Subscription ID : 50adf8b0-52d9-4365-8f54-8cc94f7dc3d8	Location : West Europe		
IU-OpenAI-ANALYTICS	🔷 Tags	Tags (edit) : Add tags			
🕑 TU-OpenAl-API	Resource visualizer	Resources Recommendations			
🕑 TU-OpenAl-DS	🗲 Events				
间 TU-OpenAl-KV	··· Settings	Filter for any field Type equals all \times Location equals all \times $^+\!$			
STU-OpenAl-LLM-D1	🛆 Deployments	Showing 1 to 2 of 2 records.			
	Security	Name 🗘	Туре ↑↓		
	Deployment stacks	Second Se	Azure OpenAl		
	Policies	TU-OpenAl-Keyvault-D1	Key vault		
	In Properties				
	Cost Management				
	South a last a last a last a last				
	Cost alerts (preview)				
	Advisor recommendations				

OpenAI Workspace environment allows the user to explore, develop and deploy models.



We can create a new deployment by selecting '**Model Deployment**' or '**Deploy**' actions. This is then followed by choosing '**Create new deployment**' option. The Deployment name can then be referenced in the API calls.

Azure Al Azure Al Studio					
«	Azure A	Al Studio > Deployments			
🟠 Azure OpenAl	Dep	loyments			
Playground	Deploy deploy	ments provide endpoints to the Azure ments, edit them, and create new deplo	OpenAl base models, or your fine-tuned models, configu pyments.	red with settings to mee	t your needs, incl
🖨 Chat	+ 0	Greate new deployment 🖉 Edit deplo	oyment 📋 Delete deployment 🧷 Column options	🕐 Refresh	in Playground
<mark>8</mark> 9 Completions		Deployment name \vee	Model name \vee	Model version \vee	Deployme
DALL-E (Preview)		<u>gpt-35-test</u>	gpt-35-turbo	0301	Standard
Management	0	ada-embedding-test	text-embedding-ada-002	2	Standard
T Deployments					
🕞 Models					
🖸 Data files					
🗟 Quotas					
Content filters (Preview)					

When creating new deployment, it is necessary to select advanced options (available once the model has been chosen) and reduce the **Tokens Per Minute** value to the lowest acceptable one. This value corresponds to the total amount of tokens that can be sent to the model each minute. Since this value counts towards Tartu University global quota, it is important to have a justification for having values above 40000.

🗰 Azure Al Azure Al Studio		
«	Azure Al Studio > Deployments	
🟠 Azure OpenAl	Deployments	
	Deployments provide endpoints to the Azure OpenAI base m	en e
Playground	deployments, ear them, and create new deployments.	Deploy model ×
ф Chat	+ Create new deployment 🖉 Edit deployment 🗎 D	Set up a deployment to make API calls against a provided base model or a custom
89 Completions		model. Finished deployments are available for use. Your deployment status will move to succeeded when the deployment is complete and ready for use.
DALL-E (Preview)		Select a model ()
Management		gpt-35-turbo
Deployments		Deployment name ()
6		*
C Models		
Data files		Content Filter ()
🔓 Quotas		Default V
Content filters (Preview)		
		200k tokens per minute quota available for your deployment
		Tokens per Minute Rate Limit (thousands) ①
		Corresponding requests per minute (RPM) = 240
		Create Cancel

You can also add a budget option by choosing your '**Resource group**' and navigating to '**Cost Management'** -> '**Budgets**'.

Home > TU-OpenAI-LLM-ITO									
TU-OpenAI-LLM-ITC Resource group) Budgets 🖉 ☆								
	+ Add 🕐 Refresh 🔞	Help \checkmark							
 Overview 	Scope : 💽 TU-OpenAl-LL	M-ITO	e All periods ~						
Activity log									
Access control (IAM)	Name	†⊥ Scope	1 Reset period	1 Creation date	↑: Evolution date ↑:	Budget	↑ Forecasted	↑ Evaluated snend	↑ Progr
🔷 Tags	You do not have any budget	, scope	1. Reset period	to creation date	rç expiration date i t	ç budget	1. Torecasted	10 Evaluated spend	iş nogi
🐥 Resource visualizer	Tou do not have any budget	3.							
🗲 Events									
Settings									
Deployments									
Security									
Deployment stacks									
Policies									
Properties									
🔒 Locks									
Cost Management									
🔍 Cost analysis									
Cost alerts (preview)									
③ Budgets									
Advisor recommendations									
Monitoring									
Insights (preview)									
Alerts									
ni Metrics									
Diagnostic settings									
🧬 Logs									
Advisor recommendations									
Workbooks									

From there you can name your budget, choose creation and expiration date and 'amount (in euros)' for your limit.

Home > TU-OpenAI-LLN	M-ITO Budgets >	
Create budget		
1 Create a budget	(2) Set alerts	
Create a budget and set a	alerts to help you monitor your costs.	VIEW OF MONTHLY COST DATA
Budget scoping		
The budget you create wi to have your budget mon	Il be assigned to the selected scope. Use additional filters like resource groups itor with more granularity as needed.	
Scope	TU-OpenAI-LLM-ITO	
Filters	⁺ ∀ Add filter	
Budget Details		
Give your budget a uniqu expiration date and the a	e name. Select the time window it analyzes during each evaluation period, its mount.	
* Name	Enter a unique name	
• Reset period 🛈	Monthly	No data to display
* Creation date 🔅	2023 ∨ October ∨ 1	
• Expiration date 🛈	2025 × September × 30 ×	
Budget Amount		
Give your budget amount	t threshold	
Amount *	0	

The '**Explore**' section in the OpenAl overview can be used to interact with the deployed models from the user interface.

Home > TU-OpenAl-LLM-D1 >		
Some State Stat	* …	
₽ Search «	🕝 Go to Azure OpenAl Studio 🔋 Delete	
Overview	↑ Essentials	
Activity log	Resource group (move) : TU-OpenAI-LLM-D1	API type : Azure OpenAl
Access control (IAM)	Status : Active	Pricing tier : Standard
Tags	Location : West Europe	Endpoint : https://tu-openai-d1.openai.azure.com/
X Diagnose and solve problems	Subscription (move) : Azure OpenAl Subscription ID : 50adf8b0-52d9+4365-8f54-8cc94f7dc3d8	Manage keys : <u>Click here to manage keys</u>
Resource Management	Tags (<u>edit</u>) : <u>Add tags</u>	
📍 Keys and Endpoint	Get Started Develop	
Ø Model deployments		the Asime Over Al Studie and estated welving ADI cells
Pricing tier	Deploy a model in	a the Azure OpenAl Studio and get started making API calls
Networking	task	with only a few examples or by specifying your task in English
💼 Identity		
S. Cost analysis	Ś	
Properties	Explore	Deploy
🔒 Locks	Get started now with the new Azure OpenAI Studio.	Learn the basics and check out our sample code Deploy a model in the Azure OpenAl Studio and get started making API calls
Monitoring	Explore	Develop Deploy
Alerts		
iii Metrics		
Diannostic settinos		

For large scale interactions for experimental or application purposes the preferred option is to access the API through a different endpoint that is managed by Tartu University directly. For Reader users this is the only way to interact with the OpenAI API outside of the User interface seen before.

The Tartu University managed API can be accessed using REST interface with POST method. The reference for Open AI REST API can be found here:

https://learn.microsoft.com/en-us/azure/ai-services/openai/reference

For current implementation only 'chat completions' (GPT-35-Turbo model) and 'Embeddings' (textembedding-ada-002 model) are supported. For the API REST access for chat completions, you will need 3 parameters to access the API:

- Your resource name the last part of your resource group name in azure, this should be provided via email together with these instructions
- Your deployment name this is the deployment name found in Azure OpenAl Studio, it is the deployment name a **Contributor** user creates when deploying a model (see image above)
- Your API Key this is found in the Key Vault resource in your Azure resource group (see access instructions bellow pages 6 7)
- (optional) Your model name when using Python openai library a model name has to be provided, it is the model name that is found under deployments in Azure OpenAI Studio (see image above)

These are three examples for accessing Tartu University managed OpenAI API

Using CURL:

```
curl --request POST \
  --url 'https://tu-openai-api-management.azure-api.net/<your resource
name>/openai/deployments/<your deployment name>/chat/completions?api-ver-
sion=2023-07-01-preview' \
  --header 'Content-Type: application/json' \
  --header 'User-Agent: Insomnia/2023.5.6' \
  --header 'api-key: <your key here>' \
  --data '{
    "messages": [
        {
            "role": "user",
            "content": "Does Azure OpenAI support customer managed keys?"
        }
    ],
    "max_tokens": 200,
    "temperature": 0.3
}'
```

Using Python requests package:

```
import requests
```

```
url = "https://tu-openai-api-management.azure-api.net/<your resource</pre>
name>/openai/deployments/<your deployment name>/chat/completions "
querystring = {"api-version":"2023-07-01-preview"}
payload = {
    "messages": [
        {
            "role": "user",
            "content": "Does Azure OpenAI support customer managed keys?"
        }
    ],
    "max_tokens": 200,
    "temperature": 0.3
}
headers = {
    "Content-Type": "application/json",
    "User-Agent": "Insomnia/2023.5.6",
    "api-key": "<your key here>"
}
response = requests.request("POST", url, json=payload, headers=headers,
params=querystring)
```

Using Python openai package:

```
import openai
openai.api_type = "azure"
openai.api_key = "<your key here>"
openai.api_base = "https://tu-openai-api-management.azure-api.net/<your
resource name>"
openai.api_version = "2023-07-01-preview"
# create a chat completion
chat_completion = openai.ChatCompletion.create(
    deployment_id = "<your deployment name>",
    model="<your model name>",
    messages=[{"role": "user","content": "Does Azure OpenAI support customer
managed keys?"}]
)
```

To access the value of your **API Key** for Tartu Managed OpenAI API we need to go to **Key Vault** in our resource group which hosts the two API keys that can be used for access. Both keys are valid and are

used to rotate access when changing keys.

		<u>م</u>	Search resources, services, and docs (G+/)	
Home > Resource groups > Resource groups Tartu Ülikool (ut-ee)	«	FU-OpenAI-LLM-D1	☆ ☆ …	
Tartu Ülikool (ut-ee) + Create ③ Manage view ∨ ···· Filter for any field Name ↑↓ [@ LogAnalyticsDefaultResources [@ TU-OpenAI-ANALYTICS [@ TU-OpenAI-API [@ TU-OpenAI-API [@ TU-OpenAI-KV [@ TU-OpenAI-KV [@ TU-OpenAI-LLM-D1		Resource group Earch Activity log Activity log Access control (IAM) Tags Resource visualizer Events Settings Deployments Security Deployment stacks Policies III Properties Locks	+ Create ③ Manage view ∨ Delete resource group ○ Refresh ↓ Export to CSV S Oper ∧ Essentials Subscription (move) : Azure OpenAI Subscription ID : 50adf8b0-52d9-4365-8f54-8cc94f7dc3d8 Tags (edit) : Add tags Resources Recommendations Filter for any field Type equals all × Location equals all × [†] √ Add filter Showing 1 to 2 of 2 records. Show hidden types ○ Name ↑↓ ③ TU-OpenAI-D1 ⑦ TU-OpenAI-Keyvault-D1	Type ↑↓ Assign ta Deployments : 2.5µ Location : West
		Cost Management \$. Cost analysis Cost alerts (preview)		

The specific keys are located under 'Secrets' section in the resource:

Microsoft Azure	✓ Search resources, services, and docs (G+/)
Home >	
() IU-OpenAI-Keyvaul	t-D1
Overview	A Upcoming TLS 1.0, 1.1 deprecation: Please enable support for TLS 1.2 on clients (applications/platform) to avc
Activity log	
Access control (IAM)	Resource group (move) : TU-OpenAI-LLM-D1
🗳 Tags	Location : West Europe
🗙 Diagnose and solve problems	Subscription (move) : <u>Azure OpenAl</u>
≆ Access policies	Subscription ID : 50adf8b0-52d9-4365-8f54-8cc94f7dc3d8
🗲 Events	
Objects	
📍 Keys	Tags (edit) : Add tags
Secrets	Get started Properties Monitoring Tools + SDKs Tutorials
🔁 Certificates	
Settings	Ma
	Our recommendation is to use a vau not share :
Networking	
Ø Microsoft Defender for Cloud	≡ €
Properties	Control access to key vault
🔒 Locks	Assign access policy and determine whether a given servic
Monitoring	

Keys can be accessed by clicking on them to open the versioning and then clicking once more on the most recent version:

■ Microsoft Azure	∠P Search resources, services, and docs (G+/)
Home > TU-OpenAl-Keyvault-D1	
TU-OpenAl-Keyvaul	t-D1 Secrets 🛪 …
✓ Search «	+ Generate/Import 🕐 Refresh
(*) Overview	
Activity log	Name Type
Access control (IAM)	Azure-OpenAl-API-Key1
Tags	Azure-OpenAl-API-Key2
🗙 Diagnose and solve problems	
š⊟ Access policies	
🗲 Events	
Objects	
📍 Keys	
🖪 Secrets	
Certificates	
Settings	
š⊟ Access configuration	
I Networking	
• · · · · · · · · · ·	

Version:

■ Microsoft Azure		\mathcal{P}^{-} Search resources, services, and docs (G+/)
Home > TU-OpenAI-Keyvault-D1 Secret	s > Azure-OpenAl-API-Key1 > 733f089480af507	
Properties		
Created	8/28/2023, 2:37:44 PM	
Updated	8/28/2023, 2:37:44 PM	
Secret Identifier	https://tu-openai-keyvault-d1.vault.azure.net/	secrets/Azure-OpenAI-API-Key1/8d73d84dfce14078a733f0 🗈
Settings Set activation date 💿		
Set expiration date ①		
Enabled	Yes No	
Tags	0 tags	
Secret Content type (optional) Show Secret Value		
Secret value	•••••	

The key can be seen by clicking on the 'Show Secret Value'. After copying the key, we can insert it in the <your key here> section in one of the code examples above.

Additional information – access for embeddings models

For the API REST access for embeddings, you will need the same 3 parameters to access the API:

- Your resource name the last part of your resource group name in azure, this should be provided via email together with these instructions
- Your deployment name this is the deployment name found in Azure OpenAl Studio
- Your API Key this is found in the Key Vault resource in your Azure resource group (see access instructions in pages 6 - 7)

These are three examples for accessing Tartu University managed OpenAI API for embeddings

Using CURL:

```
curl --request POST \
    --url 'https://tu-openai-api-management.azure-api.net/<your resource
name>/openai/deployments/<your deployment name>/embeddings?api-version=2023-
07-01-preview' \
    --header 'Content-Type: application/json' \
    --header 'User-Agent: Insomnia/2023.5.6' \
    --header 'api-key: <your key here>' \
    --data '{"input": "Does Azure OpenAI support customer managed keys?"}'
```

Using Python requests package:

```
import requests
url = "https://tu-openai-api-management.azure-api.net/<your resource
name>/openai/deployments/<your deployment name>/embeddings"
querystring = {"api-version":"2023-07-01-preview"}
payload = {"input": "Does Azure OpenAI support customer managed keys?"}
headers = {
    "Content-Type": "application/json",
    "User-Agent": "Insomnia/2023.5.6",
    "api-key": "<your key here>"
}
response = requests.request("POST", url, json=payload, headers=headers,
params=querystring)
```

Using Python openai package:

```
import openai
openai.api_type = "azure"
openai.api_key = "<your key here>"
openai.api_base = "https://tu-openai-api-management.azure-api.net/<your
resource name>"
openai.api_version = "2023-07-01-preview"
# create a chat completion
chat_completion = openai.ChatCompletion.create(
    engine = "<your deployment name>",
    input = "Does Azure OpenAI support customer managed keys?"
)
```