

Machine Learning and Video Analysis on Microsoft Azure

Troubleshooting Guide



Welcome to the troubleshooting guide of VideoHacker: Index and Analyze Videos With Microsoft Azure! This guide will help you with troubleshooting any issues you may face while following the workshop materials.

Please choose your Operating System below to see the troubleshooting guide:

[Windows](#)

[OSX](#)

If curious, you can also read about further [background information](#) about this technology and potential [next steps](#) if you enjoyed this workshop.

This is the troubleshooting guide for Windows. If that is not your operating system, please go back to the first page to choose the right troubleshooting guide.

The following sections describe the possible problems that could occur or things to be aware of when following the workshop.

SECTION 1: CREATING AN AZURE ACCOUNT

- It is possible that there are some errors while creating an Azure account. Most of these errors disappear when you **create the account in an incognito tab**.
- During signup, you can verify your mobile number by providing an OTP. If you choose to receive the OTP via text message, note that it might take some time. Simply wait for a minute and if you do not receive an OTP by then, **click on the “I did not receive a code” button**.
- **Do not use the OTP via call option**, as this simply redirects you to the beginning of the form.
- If you have previously used Azure and have used your free trial credits, you will need to input your credit card information and **sign up for a Pay-As-You-Go Account**.
- If you do not have a student email address, you will need to input your credit card information and **sign up for a Pay-As-You-Go Account**.

SECTION 2: UPLOADING VIDEOS TO VIDEO INDEXER

- Login using your corporate account (university i.e. student accounts are considered corporate)
- Connecting it to Azure takes quite a bit of time so **be patient!**
- If you face the error which says “Try to **refresh the page**” then do exactly that. It takes a couple of tries before the videos upload successfully, especially since many people are simultaneously trying to upload videos right now.



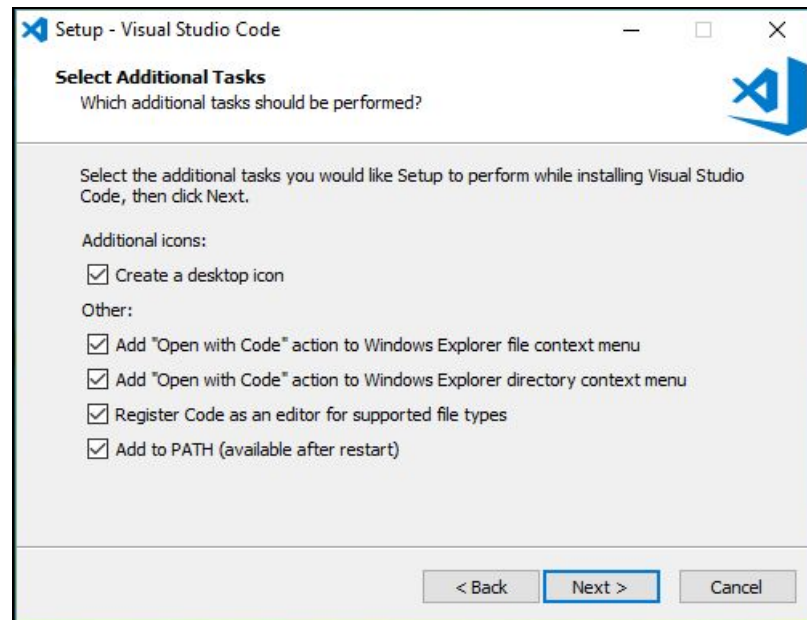
- Use the refresh button next to the upload button to check the status of your uploads.



SECTION 3: DOWNLOAD AND INSTALL VS CODE

- If the site does not show the correct version for your OS, **scroll down to the bottom** of the site and **manually select the version** that matches your OS.



- **Restart your PC** after installation to ensure the PATH variable is set (not essential, but may be useful).
- During install, **choose all the options** in this dialog box to make it easy for you to work with VS code (not essential, but may be useful).



SECTION 4: INSTALL NODE

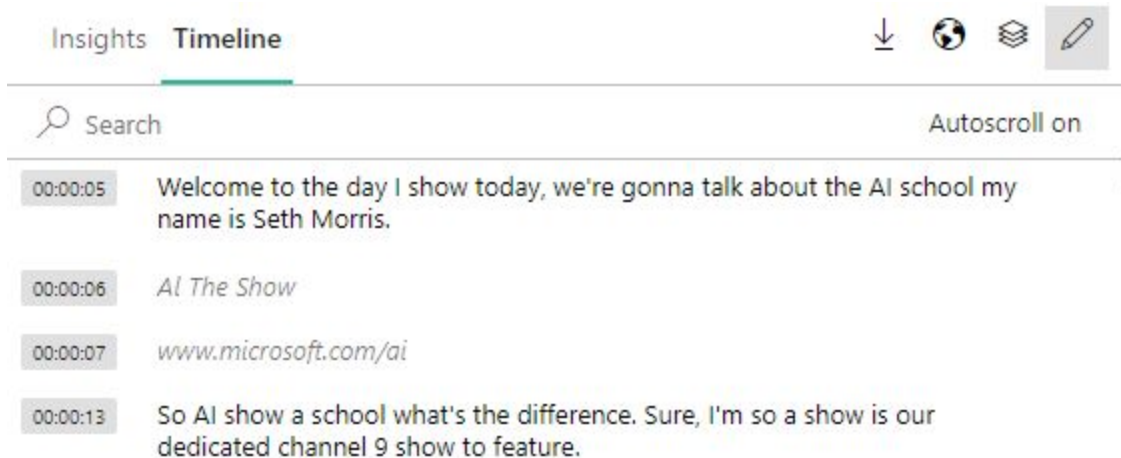
- After installing node, if your VS Code is open, **first close it**. Reopen VS code, create a new terminal and try to run the npm install (ensure that you are in the proper directory first). If that fails (shows “npm is not recognized”), **kill the terminal**, create another terminal and try the command again.

```
npm : The term 'npm' is not recognized as the name of a cmdlet, function, script file, or operable program. Check the spelling of the
name, or if a path was included, verify that the path is correct and try again.
At line:1 char:1
+ npm install
+ ~~~~~
+ CategoryInfo          : ObjectNotFound: (npm:String) [], CommandNotFoundException
+ FullyQualifiedErrorId : CommandNotFoundException
```

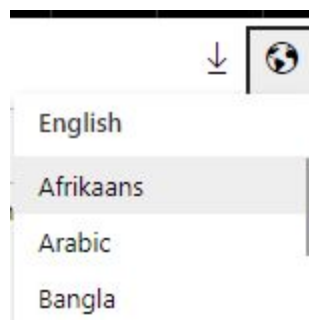
- To kill the terminal, click on the icon that looks like this -  and to add a new terminal, click on the icon that looks like this - 

SECTION 5: EXPLORE AN INDEXED VIDEO

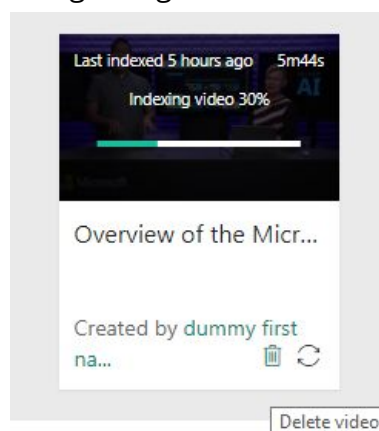
- You need to **play the video** in order to see the insights tab.
- The transcript tab is now called the timeline tab. And the phrase to change is “So AI show a school”.



- **After editing the name and text, press enter** to ensure the change is captured. If you see a progress icon next to that paragraph, you are doing it right.
- To translate the video, simply **press on the globe icon** and choose a language from the dropdown.



- If the video does not get indexed, **delete it and reupload**. The delete button can be found by hovering over the right edge of the video box as shown below -



SECTION 6: CREATE VIDEO INDEXER API SUBSCRIPTION

- Check your API key under My Subscriptions in the HOME TAB. Click show to see the primary key and copy it over.

Profile

Email

First name

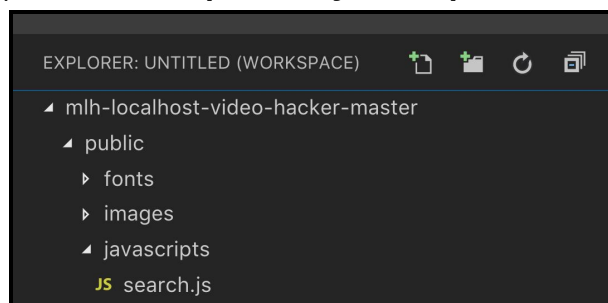
Last name

Your subscriptions

Subscription details			Product	State
Subscription name	Product Authorization subscription	Rename	Authorization	Active
Primary key	xx	Show Regenerate		
Secondary key	xx	Show Regenerate		

SECTION 7: ADDING CODE

- Ensure you **delete the mlh-localhost-videohacker-master part in the extract all dialog box**, or else there will be a nesting of folders.
- If you do not see search.js (or any other file) in VS Code explorer, **delete the extracted folder, extract again**, reopen VS Code and add the new folder to its workspace.
- The file search.js is present under **public -> javascripts** folder.



- Replace the phrases ACCOUNT_ID and SUBSCRIPTION_KEY with your account id and subscription key. Ensure that both are **surrounded by double quotes ""**.

SECTION 8: TEST APP

- To open a new terminal, go to the **Terminal menu -> Create a new terminal**.
- Ensure you are in the correct directory in the terminal. i.e. you should see something like `\Downloads\mlh-localhost-hackerlog-master>` and the cursor waiting there. If that doesn't happen, **kill the terminal and create a new one**. If that still doesn't help, **restart VS code**. Otherwise you might get an error like "ENOENT: no such file or directory, open '..\package.json'"
- Wait till **"App listening on port 3000!"** is shown on the console before trying to open localhost:3000 in your browser or it will say "This site can't be reached"



This site can't be reached

localhost refused to connect.

- Did you mean <http://localhost3000.org/>?
- Search Google for **localhost 3000**

ERR_CONNECTION_REFUSED

- Type **clear** in your terminal to delete all the text shown, so that there is less clutter (not necessary, but recommended).
- If you cancel the extraction midway (or if the extraction doesn't complete successfully for some reason) it will still create a folder but it might not have all the files so it may throw errors when you run npm start. If this happens to you, close VS code, **delete the extracted folder, extract again** and repeat all the steps from there. One common error is "failed to lookup..."
- If you get the error "listen EADDRINUSE 0.0.0.0:3000", ensure there are no other terminals where the node index.js command is still running. If there are, **kill those terminals** and **re-run the command in a new terminal**.

```
Error: listen EADDRINUSE 0.0.0.0:3000
    at Object._errnoException (util.js:1022:11)
    at _exceptionWithHostPort (util.js:1044:20)
    at Server.setupListenHandle [as _listen2] (net.js:1367:14)
    at listenInCluster (net.js:1408:12)
    at doListen (net.js:1517:7)
    at _combinedTickCallback (internal/process/next_tick.js:141:11)
    at process._tickCallback (internal/process/next_tick.js:180:9)
    at Function.Module.runMain (module.js:695:11)
    at startup (bootstrap_node.js:188:16)
    at bootstrap_node.js:609:3
```

- If you see an "unable to fetch" pop up in your webpage, go back to the **VS Code terminal** to see the error message displayed.
- If you see the error "StatusCodeError: 401 - USER_NOT_ALLOWED"; go back to your VideoIndexer site and **ensure you are connected to Azure**. You should not be seeing a message like this -

⊗ Video Indexer can't connect to your Azure Media Services account. [Contact support](#)

Ensure that you are in the correct account (VideoIndexer sometimes creates extra accounts with the names of your resource group, ensure you are in the account that has your Azure username on it).

- If you see the error "StatusCodeError: 500 - Internal server error" ensure that your **ACCOUNT_ID** and **SUBSCRIPTION_KEY** are the right value and retry.

This is the troubleshooting guide for Mac. If that is not your operating system, please go back to the first page to choose the right troubleshooting guide.

The following sections describe the possible problems that could occur or things to be aware of when following the workshop.

SECTION 1: CREATING AN AZURE ACCOUNT

- It is possible that there are some errors while creating an Azure account. Most of these errors disappear when you **create the account in an incognito tab**.
- During signup, you can verify your mobile number by providing an OTP. If you choose to receive the OTP via text message, note that it might take some time. Simply wait for a minute and if you do not receive an OTP by then, **click on the “I did not receive a code” button**.
- **Do not use the OTP via call option**, as this simply redirects you to the beginning of the form.
- If you have previously used Azure and have used your free trial credits, you will need to input your credit card information and **sign up for a Pay-As-You-Go Account**.
- If you do not have a student email address, you will need to input your credit card information and **sign up for a Pay-As-You-Go Account**.

SECTION 2: UPLOADING VIDEOS TO VIDEO INDEXER

- Login using your corporate account (university i.e. student accounts are considered corporate)
- Connecting it to Azure takes quite a bit of time so **be patient!**
- If you face the error which says “Try to **refresh the page**” then do exactly that. It takes a couple of tries before the videos upload successfully, especially since many people are simultaneously trying to upload videos right now.

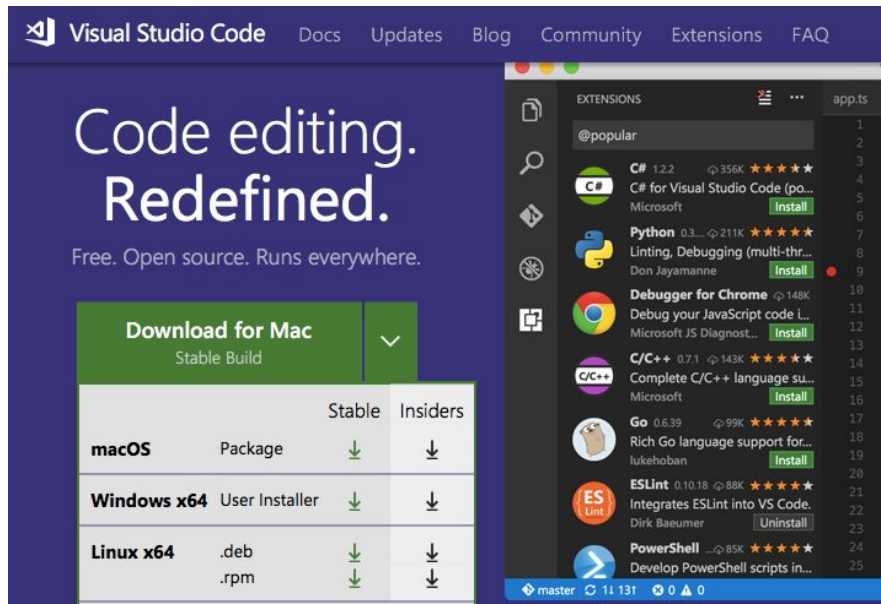


- Use the refresh button next to the upload button to check the status of your uploads.

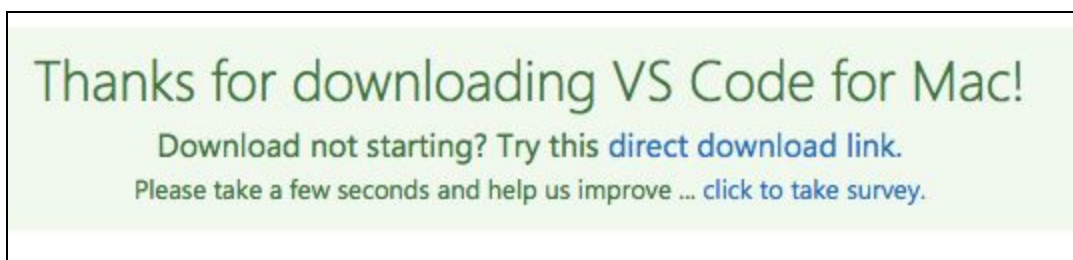


SECTION 3: DOWNLOAD AND INSTALL VS CODE

- Select **Mac** as your operating system on Visual Studio Code's website and the Stable version as the package you want to download.



- If the Visual Studio Code package does not start automatically downloading, you can select **Direct Download**.



- **Restart your PC** after installation to ensure the PATH variable is set (not essential, but may be useful).
- If you have problems with \$PATH, you can open your terminal and run the command **sudo nano /etc/paths**. Enter your password when prompted. Then add the desired path. Hit "Y" to save the modified path. This option should not be used before simply restarting the computer, as this can have negative effects later.

SECTION 4: INSTALL NODE

- Node demands two different packages before you can install it. Many developers already have these packages on their computer, but if Node is having a hard time running, it may be because these packages are not installed. You can go to NodeJS.org and **download the Node installer for Mac**.

Node.js® is a JavaScript runtime built on [Chrome's V8 JavaScript engine](#).

Download for macOS (x64)

10.13.0 LTS
Recommended For Most Users

11.1.0 Current
Latest Features

[Other Downloads](#) | [Changelog](#) | [API Docs](#) [Other Downloads](#) | [Changelog](#) | [API Docs](#)

Or have a look at the [Long Term Support \(LTS\) schedule](#).

Sign up for [Node.js Everywhere](#), the official Node.js Monthly Newsletter.



- Or, if you are worried about any packages you download wreaking havoc on your computer, you can **download Homebrew, a package manager for Mac**. To install Homebrew, open your Terminal and type in:

`ruby -e "$(curl -fsSL`

`https://raw.githubusercontent.com/Homebrew/install/master/install)"`.

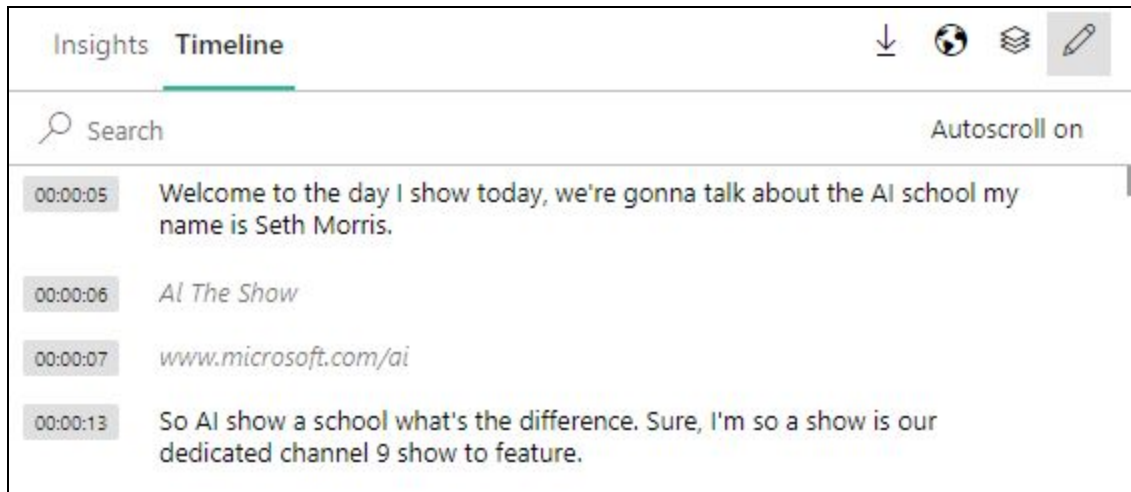
- After installing node, if your VS Code is open, **first close it**. Reopen VS code, create a new terminal and try to run the npm install (ensure that you are in the proper directory first). If that fails (shows “npm is not recognized”), **kill the terminal**, create another terminal and try the command again.

```
npm : The term 'npm' is not recognized as the name of a cmdlet, function, script file, or operable program. Check the spelling of the
name, or if a path was included, verify that the path is correct and try again.
At line:1 char:1
+ npm install
+ ~~~~~
+ CategoryInfo          : ObjectNotFound: (npm:String) [], CommandNotFoundException
+ FullyQualifiedErrorId : CommandNotFoundException
```

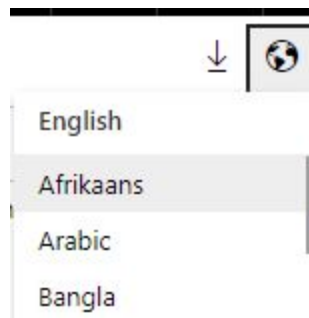
- To kill the terminal, click on the icon that looks like this -  and to add a new terminal, click on the icon that looks like this - 

SECTION 5: EXPLORE AN INDEXED VIDEO

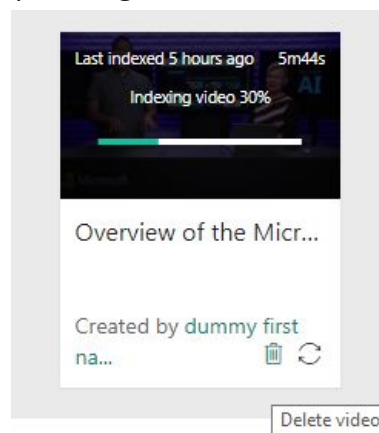
- You need to **play the video** in order to see the insights tab.
- The transcript tab is now called the timeline tab. And the phrase to change is “So AI show a school”.
- If the video does not play, check to make sure you are using the same email address for the OAuth and API key so that the video and the indexing capabilities are pointing to the same location.



- **After editing the name and text, press enter** to ensure the change is captured. If you see a progress icon next to that paragraph, you are doing it right.
- To translate the video, simply **press on the globe icon** and choose a language from the dropdown.



- If the video does not get indexed, **delete it and reupload**. The delete button can be found by hovering over the right edge of the video box as shown below. Sometimes large files can get stuck uploading if there is a weak wifi connection.



SECTION 6: CREATE VIDEO INDEXER API SUBSCRIPTION

- Check your API key under My Subscriptions in the HOME TAB. Click show to see the primary key and copy it over.

Profile

Email

First name

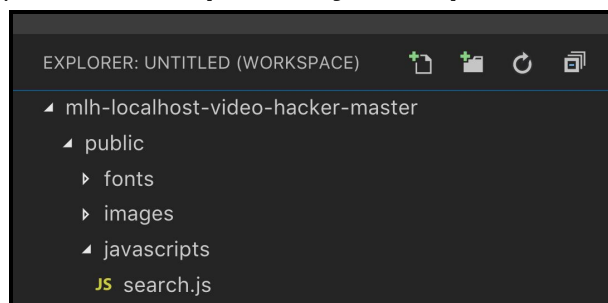
Last name

Your subscriptions

Subscription details		Product	State
Subscription name	Product Authorization subscription	Authorization	Active
Primary key	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	Show Regenerate	
Secondary key	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	Show Regenerate	

SECTION 7: ADDING CODE

- If you do not see search.js (or any other file) in VS Code explorer, **delete the extracted folder, extract again**, reopen VS Code and add the new folder to its workspace.
- The file search.js is present under **public -> javascripts** folder.



- Replace the phrases ACCOUNT_ID and SUBSCRIPTION_KEY with your account id and subscription key. Ensure that both are **surrounded by double quotes ""**.

SECTION 8: TEST APP

- To open a new terminal, go to the **Terminal menu -> Create a new terminal**.
- Ensure you are in the correct directory in the terminal. i.e. you should see something like `\Downloads\mlh-localhost-hackerlog-master>` and the cursor waiting there. If that doesn't happen, **kill the terminal and create a new one**. If that still doesn't help, **restart VS code**. Otherwise you might get an error like "ENOENT: no such file or directory, open '..\package.json'"
 - Wait till **"App listening on port 3000!"** is shown on the console before trying to open localhost:3000 in your browser or it will say "This site can't be reached"



This site can't be reached

localhost refused to connect.

- Did you mean <http://localhost3000.org/>?
- Search Google for **localhost 3000**

ERR_CONNECTION_REFUSED

- Type **clear** in your terminal to delete all the text shown, so that there is less clutter (not necessary, but recommended).
- If you get the error "listen EADDRINUSE 0.0.0.0:3000", ensure there are no other terminals where the node index.js command is still running. If there are, **kill those terminals** and **re-run the command in a new terminal**.

```
Error: listen EADDRINUSE 0.0.0.0:3000
    at Object._errnoException (util.js:1022:11)
    at _exceptionWithHostPort (util.js:1044:20)
    at Server.setupListenHandle [as _listen2] (net.js:1367:14)
    at listenInCluster (net.js:1408:12)
    at doListen (net.js:1517:7)
    at _combinedTickCallback (internal/process/next_tick.js:141:11)
    at process._tickCallback (internal/process/next_tick.js:180:9)
    at Function.Module.runMain (module.js:695:11)
    at startup (bootstrap_node.js:188:16)
    at bootstrap_node.js:609:3
```

- If you see an "unable to fetch" pop up in your webpage, go back to the **VS Code terminal** to see the error message displayed.
- If you see the error "StatusCodeError: 401 - USER_NOT_ALLOWED"; go back to your VideoIndexer site and **ensure you are connected to Azure**. You should not be seeing a message like this -

⊗ Video Indexer can't connect to your Azure Media Services account. [Contact support](#)

Ensure that you are in the correct account (VideoIndexer sometimes creates extra accounts with the names of your resource group, ensure you are in the account that has your Azure username on it).

- If you see the error "StatusCodeError: 500 - Internal server error" ensure that your **ACCOUNT_ID** and **SUBSCRIPTION_KEY** are the right value and retry.

BACKGROUND INFORMATION

This section provides context about the Video Analyzer application for workshop organizers to better understand the context of the workshop and answer deeper questions if workshop participants are curious to know more about the technology.

Humans recognize emotions through a wide variety of different modes, from facial expression, to voice tonality, to speech content. Computers have to analyze emotions differently, through a process called **machine learning**. Machine learning is a method of data analysis that automates the models that machines use to analyze data. Instead of a person needing to lay out for a machine exactly what happiness looks like, they can feed in a large dataset of faces that are happy, and then the computer itself building a model of what a happy face looks like. But this model is not full-proof, but simply gets statistically more likely to accurately recognize what a face is over time. A good guide on the basics of machine learning can be found [here](#).

Microsoft Azure's Video Analyzer goes further than traditional models by not simply categorizing emotions on "negative" or "positive" but a more fine-grained four different states: anger, fear, joy, and sadness. It does this by analyzing speech content (words) as well as voice tonality (audio). The computer uses both of these streams of data to understand the semantic differences between these two sentences:

	The lion escaped one month ago, but recently he was captured.
	The lion was captured one month ago, but recently he escaped.

Azure's Video Indexer has also taken a step forward in terms of how it can process images as well as audio. What is happening in the video is actually *not* a computer running a computer vision algorithm on each frame in a video. If this were the case, the frames below would read "icrosoft" "Mi" "rosoft" and "Micros" instead of the full word.



Instead, using machine learning the Indexer identifies which strings of texts likely come from the same visual source and when multiple strings need to be combined into one output string. As well, the Indexer also takes auditory input, such as a speaker saying “Microsoft” which increases the algorithm’s confidence that Microsoft is a real word by putting more **weight** on words that appear in the audio transcription as well.

NEXT STEPS

Many workshop participants want to know more about how companies can employ the technology they have just worked on, and how they individually can move individually forward with working on more in-depth projects.

Companies can use the Azure Video Indexer in a variety of ways:

- Broadcasters can use the Video Indexer to better understand the emotional impact of their advertisements, or when might be the most effective time to place a pop-up ad.
- Video platforms like Youtube or Twitch can automatically detect racy, violent, or sexually explicit content on their platforms. This can make sure that children are not seeing content that is too mature for their age.
- International news broadcasts and international summits can have a wider viewership as video can be translated near simultaneously into many different languages
- People with auditory impairments can access a wider range of shows as the application can automatically translate speech to text.
- Can create inferences of the main topics of a video which might be used to create automatically generated study guides for university lectures.

Encourage participants to brainstorm what other uses they might use video analysis technology. If participants want to move forward with deploying Video Indexer in an app, you can send them to Microsoft’s tutorial on how to [Embed Video Indexer Into An Application](#).