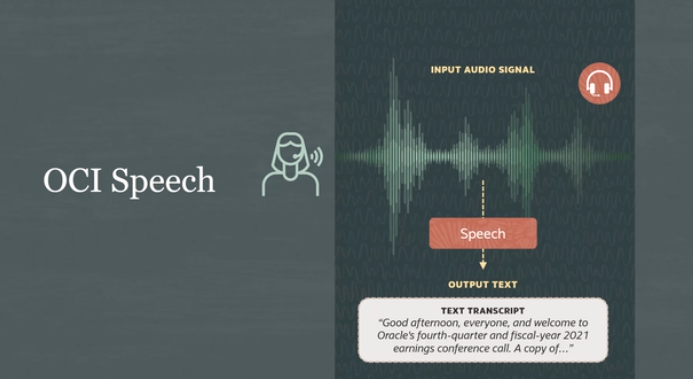


OCI Language analyzes unstructured text for you. It provides models trained on industry data to perform language analysis with no data science experience needed. It has five main capabilities. First, it detects the language of the text. It recognizes 75 languages, from Afrikaans to Welsh.

It identifies entities, things like names, places, dates emails, currency, organizations, phone numbers-- 14 types in all. It identifies the sentiment of the text, and not just one sentiment for the entire block of text, but the different sentiments for different aspects. So let's say you read a restaurant review that said, the food was great, but the service sucked. You'll get food with a positive sentiment and service with a negative sentiment. And it also analyzes the sentiment for every sentence.

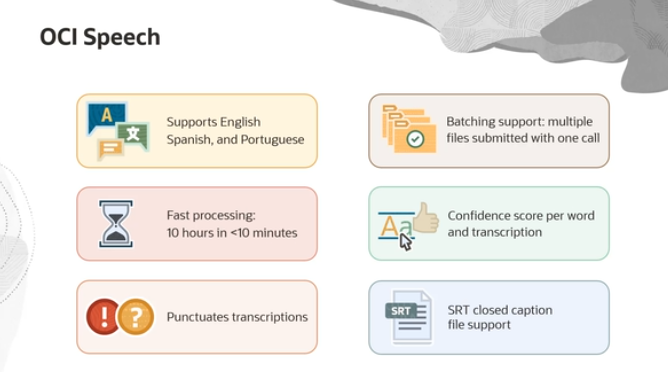
It identifies key phrases in the text that represent the important ideas or subjects. And it classifies the general topic of the text from a list of 600 categories and subcategories.

Now above content is telling us that These are the services that Oracle pre-trained models provide for language related task.



A diagram of a software development

Description automatically generated with medium confidence



Now, let's look at OCI Speech. OCI Speech is very straightforward, that it locks the data in audio tracks by converting speech to text. Developers can use Oracle's time-tested acoustic language models to provide highly accurate transcription for audio or video files across multiple languages. OCI Speech automatically transcribes audio and video files into text using advanced deep learning techniques. There's no data science experience required. It processes data directly in object storage. And it generates timestamped, grammatically accurate transcriptions.

OCI Speech supports multiple languages, specifically English, Spanish, and Portuguese, with more coming in the future. It has batching support where multiple files can be submitted with a single call. It has blazing fast processing. It can transcribe hours of audio in less than 10 minutes. It does this by chunking up your audio into smaller segments, and transcribing each segment, and then joining them all back together into a single file. It provides a confidence score, both per word and per transcription. It punctuates transcriptions to make the text more readable and to allow downstream systems to process the text with less friction.

And it has SRT file support. SRT is the most popular closed caption output file format. And with this SRT support, users can add closed captions to their video.

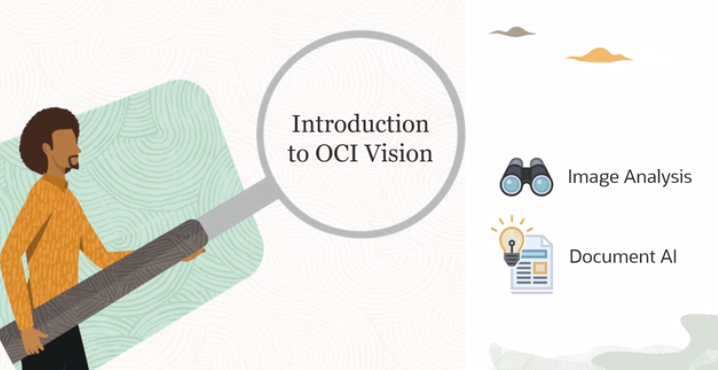
A screenshot of a computer

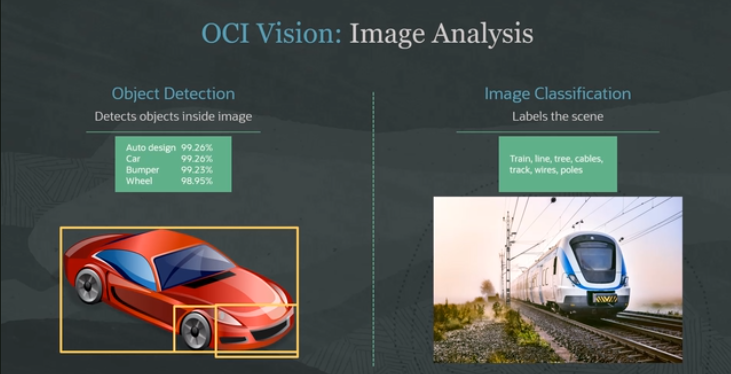
Description automatically generated

OCI Speech makes transcribed text more readable to resemble how humans write. This is called normalization. And the service will normalize things like addresses, times, numbers, URLs, and more. And you can see in the example here where you have a literally transcribed text. But on the right is the normalized text where the number is changed from words to numeric symbols.

It also does profanity filtering, where it can either remove, mask, or tag profanity and output text, where removing replaces the word with asterisks, and masking does the same thing, but it retains the first letter, and tagging will leave the word in place, but it provides tagging in the output data.

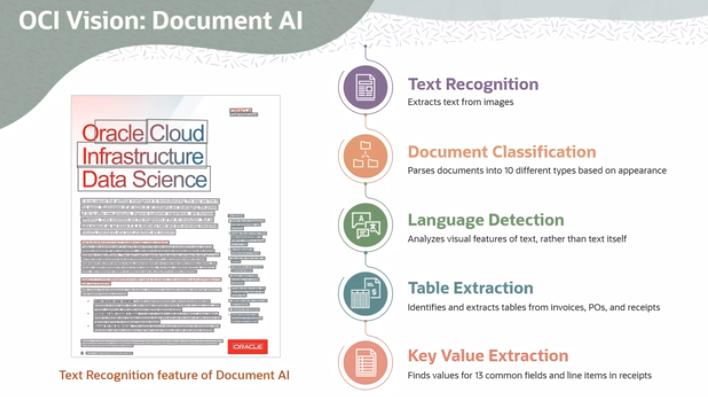
**Vision related service**

****



Next is OCI Vision. Vision is a computed vision service that works on images, and it provides two main capabilities-- image analysis and document AI. Image analysis analyzes photographic images. Object detection is the feature that detects objects inside an image using a bounding box and assigning a label to each object with an accuracy percentage. Object detection also locates and extracts text that appears in the scene, like on a sign.

Image classification will assign classification labels to the image by identifying the major features in the scene. One of the most powerful capabilities of image analysis is that, in addition to pretrained models, users can retrain the models with their own unique data to fit their specific needs.

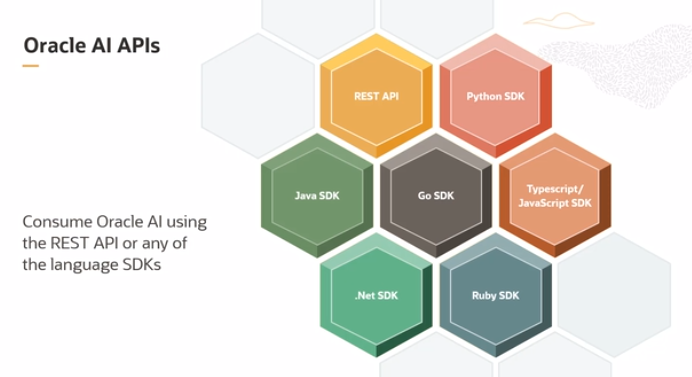


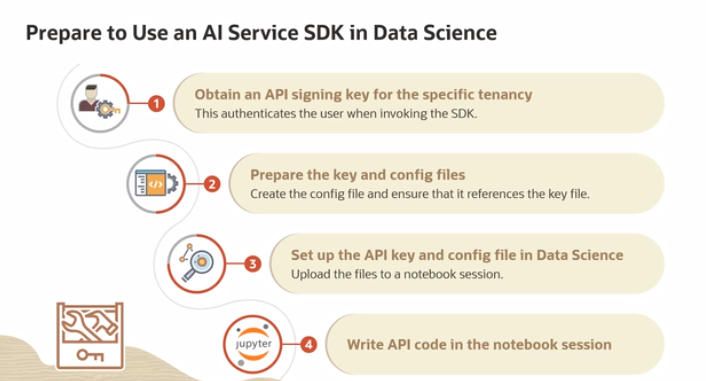
A second major capability of Vision is called document AI, and it's used for working with document images. You can use it to understand PDFs or document image types, like JPEG, PNG, and Tiff, or photographs containing textual information. The features of document AI are text recognition, also known as OCR or optical character recognition.

And this extracts text from images, including non-trivial scenarios, like handwritten texts, plus tilted, shaded, or rotated documents. Document classification classifies documents into 10 different types based on visual appearance, high level features, and extracted keywords. This is useful when you need to process a document, based on its classification, like an invoice, a receipt, or a resume.

Language detection analyzes the visual features of text to determine the language rather than relying on the text itself. Table extraction identifies tables in docs and extracts their content in tabular form. Key value extraction finds values for 13 common fields and line items in receipts, things like merchant name and transaction date.

**Oracle APIs and SDKs**





Now that you've seen what the services can do, let's look at how to access them in code. So how does a data scientist take advantage of these capabilities? Well, you can write code against the REST API or use any of the various language SDKs shown here.

But for data scientists working in OCI Data Science, it makes sense to use Python. Next, we'll look at what it takes to use the Python SDK in a notebook to be able to use the AI services. You can use a Notebook session in OCI Data Science to invoke the SDK for any of the AI services.

This might be useful to generate new features for a custom model or simply as a way to consume the service using a familiar Python interface. But before you can invoke the SDK, you have to prepare the data science notebook session by supplying it with an API Signing Key.

Signing Key is unique to a particular user and tenancy and authenticates that user to OCI when invoking the SDK. So therefore, you want to make sure you safeguard your Signing Key and never share it with another user. You can obtain an API Signing Key from your user profile in the OCI Console, then you save that key as a file to your local machine.

The API Signing Key also provides commands to be added to a config file that the SDK expects to find in the environment, where the SDK code is executing. The config file then references the key file. Once these files are prepared on your local machine, you can upload them to the Notebook session, where you will execute SDK code for the AI service.

You'll see a demo next that will show you these steps. The API Signing Key and config file can be reused with any of your notebook sessions, and the same files also work for all of the AI services. So the files only need to be created once for each user and tenancy combination.