

Reporte de visita de Microsoft:

Beyond training models: How to build AI products

David Radames Gómez Chiu

A01334574

13 de septiembre de 2018

Abstract

En este documento se presenta un reporte de la conferencia *Beyond training models: How to build AI products* impartida por un especialista en inteligencia artificial de Microsoft. El contenido del reporte se estructura en consejos proporcionados por el presentador acerca de la implementación de modelos de inteligencia artificial. Al final del documento se encuentran la sección de anexos, con una lista de herramientas útiles para el desarrollo de AI, y un glosario con términos utilizados a lo largo del documento.

Introducción

Microsoft ha indagado mucho en el campo de la inteligencia artificial y ha logrado desarrollar productos que implementan inteligencia artificial en forma de reconocimiento de imágenes, de rostros, de voz, detección de emociones (en individuos y en públicos), *chatbots*, etc. Algunos ejemplos disponibles en línea del trabajo de la empresa se encuentran en el primer anexo. En términos generales, AI podría verse como programación inversa, ya que por lo general se le dan instrucciones a una computadora y posteriormente se espera que aplique dichas instrucciones a una serie de problemas o ejemplos. En cambio, AI comienza por darle ejemplos a una computadora y luego se entrena para que identifique características y patrones en los ejemplos.

Las aplicaciones de inteligencia artificial se basan en dos conceptos fundamentales: entrenamiento y predicción. El primero de estos se refiere al aprendizaje de patrones comunes y el segundo significa que la computadora intente, con base en el entrenamiento que haya recibido, adivinar qué sigue en el patrón.

Implementación de AI

Durante la conferencia, se mencionaron una serie de consejos para implementar inteligencia artificial. El primero de ellos es: “primero haga que funcione”. Con esto se hace

referencia a que lo más importante es que un producto de *software* pueda realizar la tarea que tiene asignada y después va lo demás. Se aconseja construir soluciones basadas en heurísticas y concentrarse en primer instante en la funcionalidad. Como segundo paso, se puede alterar el *software* para que esté hecho de manera correcta y optimizada.

El segundo consejo y un punto importante de énfasis durante el evento fue: “consigue datos tempranamente”. Esta tarea puede tomar más tiempo del que comúnmente se estima, ya que se deben proporcionar datos y etiquetas que describan los datos. Un ejemplo de esto es un conjunto de datos de diferentes especies de mariposas, en el que se tenga imágenes de varias mariposas y etiquetas con el nombre de la especie de cada una. El proceso de conseguir los datos consiste de:

1. Recolectar muestras
2. Etiquetar muestras
3. Verificar etiquetas
4. Organizar mediante control de versiones

Este proceso puede resultar mucho más complejo de lo que inicialmente aparenta, en especial en el paso de la verificación. Tomando como premisa que los humanos cometen errores, es posible que varias etiquetas (dependiendo del tamaño de la muestra) sean incorrectas o que dos personas no estén de acuerdo en una etiqueta. Existen varios métodos para resolver este problema: tener más de una revisión de etiquetas, establecer una verdad con base en estadísticas de las etiquetas más usadas para las muestras dudosas, utilizar otro modelo de AI que reconozca elementos de la muestra, tener un grupo de expertos cuya opinión se considere como la verdadera, entre otras. Adicionalmente se debe mantener un registro, preferiblemente mediante control de versiones, del conjunto de datos, por si se altera algo que causa una empeora en el modelo.

Existen muchas buenas prácticas de etiquetado de elementos del conjunto de datos. En primer lugar, se debe contar con herramientas de etiquetado que puedan dividir la información en grupos. Como se mencionó anteriormente, cuando las herramientas de etiquetado no concuerden, se puede considerar la ayuda de supervisores, sobreescritura y voto de mayoría. Asimismo, se deben elaborar reportes que incluyan el proceso de etiquetado por persona, la precisión del conjunto de datos (en porcentaje de elementos correctamente etiquetados) y los resultados intermedios (incluyendo sus entrenamientos, validaciones y considerando el *overfitting*).

En tercer lugar, se aconseja administrar los conjuntos de datos. Esta necesidad surge del hecho de que cambiar alguna característica de un conjunto de datos, por más mínimo que sea el cambio, ocasiona que se esté tratando con un conjunto totalmente diferente. Por ejemplo, si se tiene un conjunto de imágenes y se les cambia la resolución, se considera un *dataset* diferente. Para el manejo adecuado de conjuntos de datos, se recomienda tener separaciones entre los datos de entrenamiento y de validación, para cada modelo.

El siguiente consejo es: “obtenga un ciclo de retroalimentación mediante métricas”. Esto se refiere a establecer métricas que ayuden a monitorear el progreso de la red neuronal en su proceso de entrenamiento y validación. Una característica de las buenas métricas es que no son cosas abstractas (satisfacción, felicidad, etc.), sino medidas tangibles y cuantificables.

El quinto consejo es: “haga *debugging* primero”. Este consejo no solamente aplica para implementación de inteligencia artificial, sino que también para el desarrollo de *software* en general. Se debe refactorizar, obtener resultados intermedios, usar control de versiones (como los mencionados en el anexo dos) y tomar instantáneas del modelo. Con *software* que puede llegar a ser muy complejo, el orden se vuelve un factor de éxito.

El sexto consejo, “comience simple”, comparte el objetivo del consejo anterior: mantener la organización con el fin de que el producto de *software* sea mantenible. La complejidad de la red aumentará con el tiempo, por lo que no se debe intentar hacer todo de una sola vez. Se debe establecer un objetivo que indique para qué se quiere optimizar la AI. Asimismo, se deben describir las características de los conjuntos de datos para saber qué se busca aprender de ellos.

En séptimo lugar viene “optimice metódicamente”. Para saber qué ajustes se le debe hacer a un modelo, se deben hacer tareas como monitorear la delta entre precisiones. Es posible obtener nuevas características a partir de errores. Se pueden utilizar métodos como la matriz de confusión y herramientas como las del anexo dos.

El octavo consejo es “mantenga versiones”. Se ha repetido múltiples veces este punto a lo largo del documento, debido a su importancia. Trabajar con redes neuronales puede ser una tarea compleja por la gran cantidad de información con la que se trabaja, así que es necesario cumplir con este punto.

Por último, el noveno consejo indica que se debe enriquecer más la red. Esta tarea incluye agregar más datos, con todo el proceso que esto conlleva; agregar más funcionalidades (establecer métricas, etc.); y hacer la red más profunda (esto significa agregar más capas a la red neuronal).

Conclusión

La inteligencia artificial es una tendencia importante de la actualidad. Tiene muchos casos de aplicación en los que puede ser sumamente útil, sin embargo, es necesario saber implementarla de una manera ordenada. Tener un procedimiento estructurado para construir una red neuronal es imperativo, ya que de esta manera es más fácil que el producto final sea mantenible y escalable para proporcionar una mejor experiencia de usuario.

Glosario

AI: inteligencia artificial (por sus siglas en inglés).

Chatbot: forma de inteligencia artificial que es capaz de mantener conversaciones.

Entrenamiento: proceso de aprendizaje de patrones.

Predicción: capacidad de deducir el siguiente paso de un patrón.

Overfitting: cuando un modelo representa demasiado bien los datos. Es decir, modela el tanto los datos útiles, como el ruido.

Dataset/conjunto de datos: colección de información (elementos de muestras y etiquetas) que se le proporciona a una red neuronal como parte de su entrenamiento y pruebas.

Debugging: corrección de errores de un producto de *software*.

Anexos

1. Productos de AI

- Captionbot.ai
- Customvision.ai

- How-old.net
- What-dog.net
- Celebslike.me
- Pix iOS app

2. Control de versiones

- Git
- Elephant (para archivos pesados)

3. Herramientas para AI

- studio.azureml.net
- insights.timeseries.azure.com
- powerbi.com
- azure.microsoft.com/en-us/services
- gallery.azure.ai

4. Cursos para aprender AI

- fast.ai