Wriker CoPilot for Pitney Bowes Shipping apps

Wriker, named for the copilot of the Starship Enterprise in Star Trek, the Next Generation, brings a shipping copilot to the next generation of Pitney Bowes shipping apps.

Wriker integrates Azure Computer Vision AI services to make it easier to generate shipping labels for people using online stores to sell their items.

Wriker will be trained using Azure Computer Vision to estimate weight and size information based on the images in the order forms, saving sellers the time and trouble of filling out online forms after their customers have already done the work for them by creating an order.

Options

We had three customer concepts originally

1. Small in-person business owners
2. Online web store owners, such as Shopify sellers
3. Office shipping department personnel

1. Small in-person business owners: Rate Shop and Get Single Rate
    a. Data input form – use document intelligence to train model
    b. Shipping info retrieval process – watch for data leakage
    c. Pros
        i. Could be very specific, so less costly to set up the copilot
        ii. Supports small businesses
        iii. Corner the small business market with copilot, even regionally
    d. Cons
        i. Less use case because of difficulty marketing, smaller amount shipping than corporate
        ii. Targeted marketing could be difficult
        iii. Data usage and storage could compromise PII
2. Online seller with webstore: Rate Shop and Get Single Rate; Manifest
    a. Data input form from customers is processed with the copilot using Doc intel
    b. Shipping info retrieval process
    c. Camera app used for inventory and packaging entries
    d. Can match website logo/design/theme
    e. Saves info automatically, can print manifests or daily/weekly/monthly usage
    f. Pros
        i. Streamlines the shipping process
        ii. Copilot helps to answer questions about measuring or weighing
        iii. Adds level of professionalism with 24/7 support for seller
        iv. Seamless integration with PitneyShip Cube

v. Can use Azure Vision for dimensions using camera images, so no forms
vi. Once it is working, anyone can do the shipping

g. Cons
i. Added expense to webstore owner, using data from copilot storage
ii. Takes time to enter the information to set up the automation
iii. Must check inventory to make sure copilot is accurate, can't just go to autopilot
iv. Data usage could compromise PII

3. Office shipping department: Multipiece; Manifest
a. Data input form with programmable interface
b. Shipping info setup page for frequent addresses
c. Able to be programmed locally with data storage
d. Multi-piece rates available
e. Pros
i. Larger customer base with fewer sales making it easier to get better ROI
ii. Greater data pool being generated to keep model trained/updated
iii. More regularity of shipments, cheaper to run
iv. Don't need to train staff to use the copilot
v. Daily tracking of expenses and packaging inventory use
f. Con
i. Might not be as useful because less need for new info; same shipments regularly
ii. Exhaustion of resources for the business due to using third party resource
iii. Data safety and compliance issues need to be addressed

Proof of Concept:

Chose option 2; Benjamin's cousin uses Shopify to sell online.

Pain points:

Create and save a new file each time – takes time and memory space

Fill out the forms for each shipment – takes time and is repetitive

Create and save file separately for label information – duplicates information and wastes time

We will let the shop owners know the expected capabilities of Wriker and what information it will use to be as transparent as possible and define the expected use.

We are not using the data to impact a person, so fairness biases will mainly be to make sure all shipping options are researched so that no company is overlooked just because another is often used.

The phone camera option aids in inclusiveness helping those with writing troubles

Reliability and safety are wrapped up in the training models and setting up Wriker to pull information only from the specific areas of the documents it is trained to pull from

Privacy and security are part of the development of code and choosing how to store or present the responses

The development team will need to accept accountability for how the app works, and the shop owner will be made aware that their images need to be their own property.

Project Goals:

Our goal is for the Azure Computer Vision services to use camera images to learn what her inventory is and measure shipping cartons using the Computer Vision image analysis feature.

Wriker is taught which packaging materials are best used to send each item, and how much each product weighs.

The packaging materials and box/envelope sizes would be manually entered, or Wriker Copilot could come with a preset group of box/envelope options with option to add more.

Wriker Copilot is trained to recognize the inventory, match it to the shipping materials necessary and suggest packing options and box sizes.

Wriker Copilot is also integrated to the shipping information captured by the seller, on their site, so owner does not need to re-type, but rather Wriker pulls that information from the site, checks the Pitney Bowes API for shipping costs based on the address and suggested box/envelope size and weight, and returns a reply with shipping costs.

Using the information provided by the seller, Wriker copilot also lets her know when her box/envelope/product inventory is getting low and suggests purchase quantity options based on the sales rate over a given period of time.

Explain how the interface works
Explain how the AI agents are working
Explain buttons or input fields

The Azure services used are:

Document intelligence
Bot services
Custom Vision
Computer Vision
AI Search

Key learning:

Copilot is a tool that needs to be verified and updated and kept trained

It makes things easier, but is not mean to allow autopilot mode for user

Connecting and integrating AI with apps is not very complicated, but does take some know-how and practice

Reliability and safety are performed by regular checks on Wriker's scores. Our code is designed to be used in a secure environment using keys, secrets and specific connections to the particular API being called for the specific data being sought. Each agent will be connected separately and then the information is integrated by Wriker.