

Sworn Script: AI powered legal document management and review

*Sworn affidavits, contracts, settlement agreements,
case briefs and more....*



**SWORN
SCRIPT**



Introduction

Efficient Automation: Sworn Script AI enables legal professionals to effectively handle extensive long form documents.



Challenges in Document Management

- Managing long legal documents can be time-consuming and error-prone
- Legal professionals face challenges like complex formatting requirements, version control issues, and tedious manual review
- These challenges can hinder productivity and increase the risk of errors in critical legal documents



Sworn Script AI: The Solution

Sworn Script is a software tool that automates document management and review. It enables users to extract important information from large documents and utilize it for various tasks including database storage, document querying, and document editing.

The Tech behind Sworn Script

OpenAI:

- *Works with GPT-3.5-Turbo*
- *Better results GPT-4*

AskMarvin:

- **AI Engineering Framework**
- **Utilizes Pydantic Library**
- **Makes it easy to structure your prompts with Class instances**

Marvin

- AI Models
- AI Functions
- AI Classifiers

AI Models

```
from marvin import ai_model
from pydantic import BaseModel, Field

@ai_model
class Location(BaseModel):
    """A representation of a US city and state"""

    city: str = Field(description="The city's proper name")
    state: str = Field(description="The state's two-letter abbreviation (e.g. NY)")

# We can now pass unstructured context to this model.
Location("The Big Apple")
```

Useful for enforcing structured data output

AI Functions

```
from marvin import ai_fn

@ai_fn
def generate_recipe(ingredients: list[str]) -> list[str]:
    """From a list of `ingredients`, generates a
    complete instruction set to cook a recipe.
    """

generate_recipe(["lemon", "chicken", "olives", "couscous"])
```

Excellent for generative tasks while still giving a desired output type.

AI Classifiers

```
from marvin import ai_classifier
from enum import Enum

@ai_classifier
class Sentiment(Enum):
    POSITIVE = "POSITIVE"
    NEGATIVE = "NEGATIVE"

Sentiment("That looks great!")
```

Used for classification tasks using a clever logits trick with the tiktoken



Areas of Improvement

Improved UI and UX

More robust query management system

- Ability to delete queries
- More control over base classes
-

Improved document exploration

Pipeline to ML Classification models

Conclusion

Efficient Automation: Empowering Legal Professionals with Sworn Script AI revolutionizes document management. AI boosts productivity and accuracy for legal professionals, streamlining the industry.