Potential of Large Language Models For Insurance Document Data Extraction


Austin Ader Ahsan Imran

Faculty Advisor: Professor Retsef Levi

Capstone Sponsor: Sai Raman

Cognisure Team: Ankita Ranjan, Rama Pallelapati

CogniSure

MIT MANAGEMENT SLOAN SCHOOL

Problem Overview

Background

The issue of unstructured documents is the biggest barrier for digitization of the underwriting process for a $250+ billion industry. Over 90% of submissions are unstructured.

Cognisure has a large set of proprietary extraction algorithms that produce accurate results (95%+ accuracy) and are trained on thousands of document types.

Despite these developments, Cognisure needs to invest significant manual effort in analysing documents that the algorithms are not trained on.

Motivation

		2023	2024	2025
Number of Documents per Year		100,000	1,000,000	10,000,000
Cognisure AI Automation		70%	75%	80%
Manual Exceptions		30%	25%	20%
		30,000	250,000	2,000,000
Manual Effort (Hrs.)	0.3	9,000	75,000	600,000
Cost (Per Hr.)	$60	**$540,000**	**$4,500,000**	**$36,000,000**

Exponential rise expected in number of manually analysed documents

Development of generic models (capable of understanding unseen documents) can reduce the manual effort needed

Scope

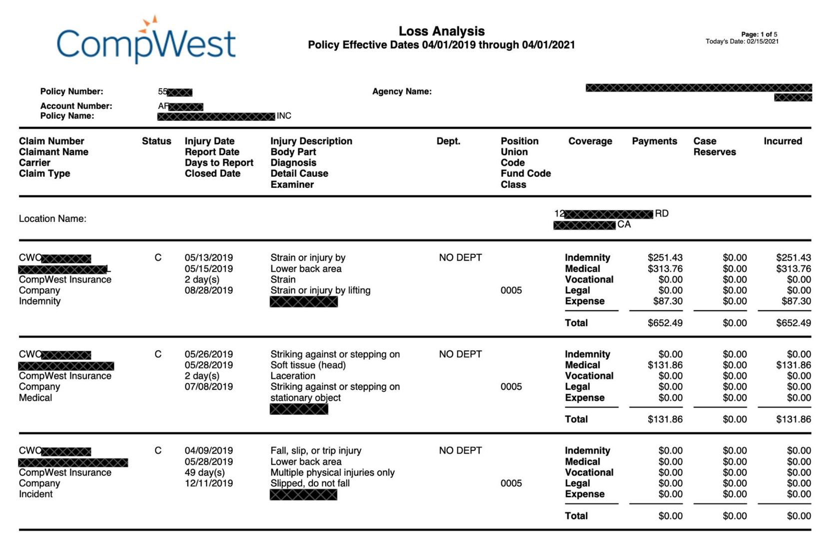
Loss runs are insurance documents that provide claims history of an insurance policy

Our goal is to extract 4 fields using machine learning approaches:

Carrier Name **Losses as of Date**
Name Insured **Run Date**

Accuracy of the model is critical as Loss Runs are used by underwriters for policy decisions

The Challenge



Complex formats and structures

Variation in formats between different carrier

Variation in formats for the same carrier

Methodology

Layout Language Model (LayoutLM)

LayoutLM is a transformer based model that relies on features such as text, images and spatial elements (coordinates/location of text).

Amazon Textract

OpenAI

1. We use Amazon Textract as our OCR engine
2. Employing an embeddings approach and develop a similarity score to find the most similar output from Textract for each field
3. BIO tagging to label all the text on the document
4. Feeding text, coordinates, and images as features to LayoutLM for field prediction

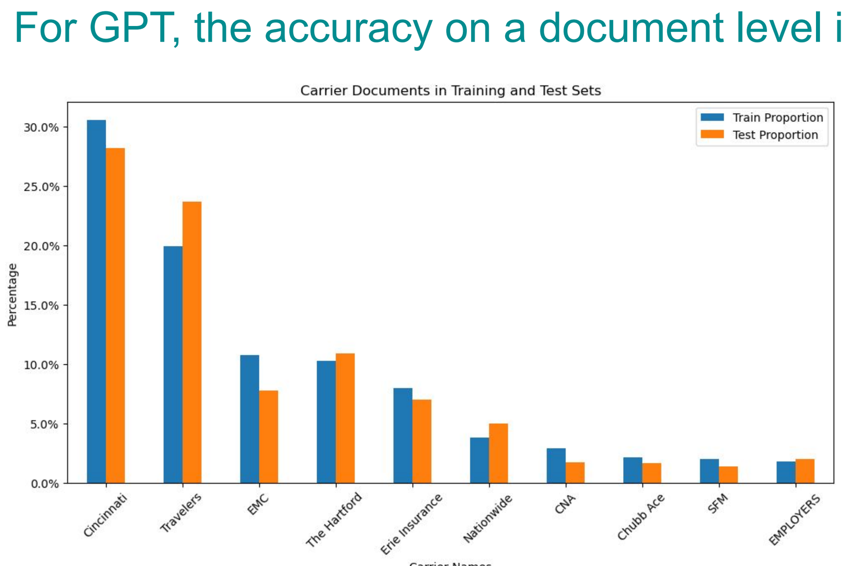
GPT

1. Using Python to extract all the text from loss runs
2. Developing prompts and feeding the prompts and the text from loss runs into GPT
3. Cleaning output from GPT and obtaining predictions for fields

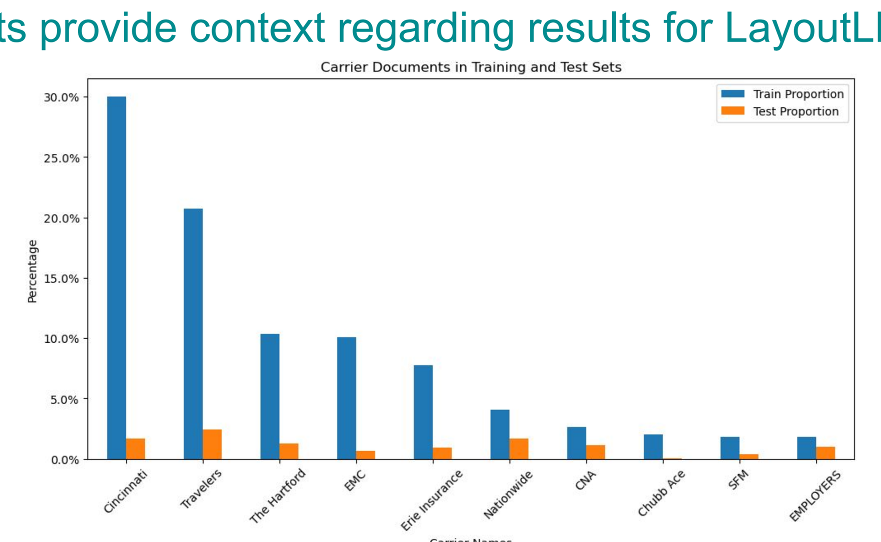
Results and Business Impact

Results

For GPT, the accuracy on a document level is **51%**; The following plots provide context regarding results for LayoutLM.



Consistency in carriers across the training and test sets allows LayoutLM to perform well with **80% accuracy** on document level



A variation in carriers across the training and test sets causes the performance of LayoutLM to drop significantly to **30% accuracy** on a document level

Impact

10-15% reduction in time spent on manual analysis

$50,000 saved per month in 2024 with an enhanced model pipeline