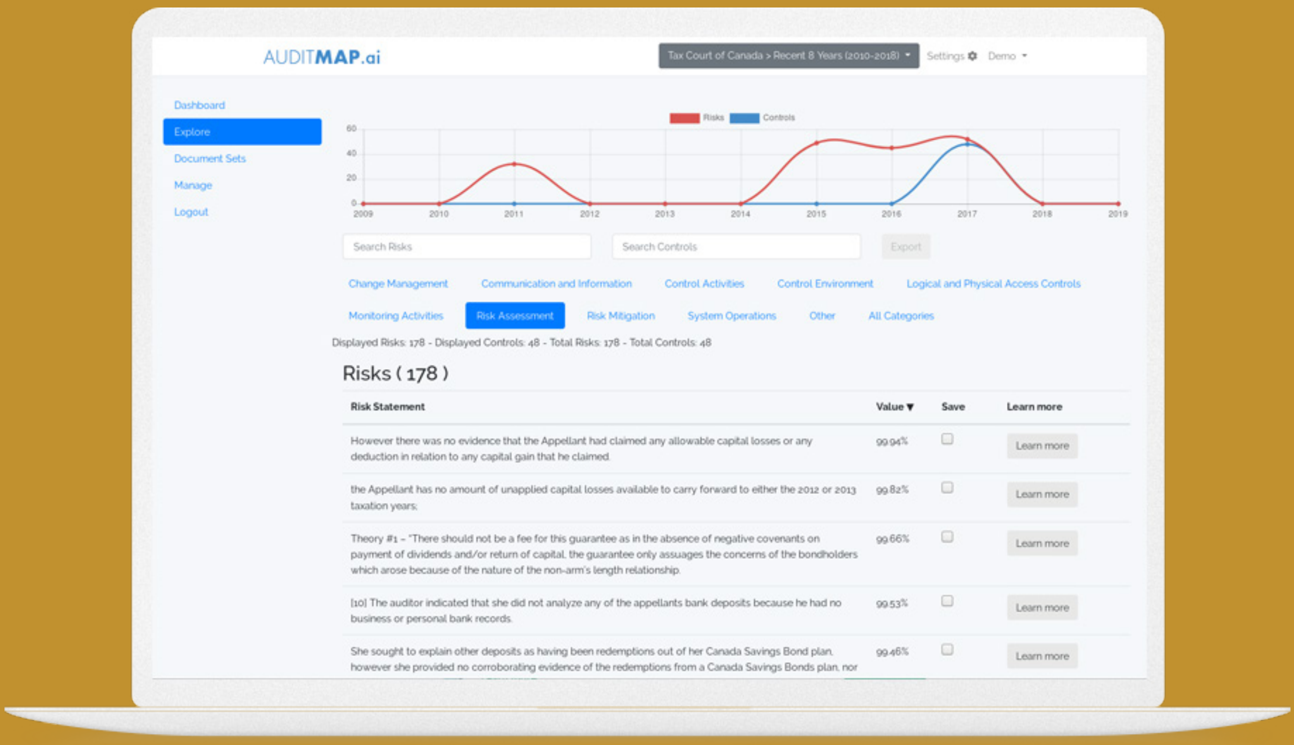
STALLION.AI

ENTERPRISE RISK INSIGHTS FOR MANAGEMENT

stallion.ai



Stallion.ai

The Problem

The Solution

How it works

Internal Audit - What is it?

Internal audits are mandatory reviews of corporate/ gov't functions, that produce actionable reports.

Human auditors must review a large corpus of unstructured data to perform their tasks.

Internal audit functions have a structured planning phase which is sample-based.

Identify Risk

Measure Risk

Prioritize Risk

Select & Develop Audits

The Problem

Too much data

Knowledge transfer from audit reports is ineffective & inconvenient

Audit coverage completeness is not measurable or quantifiable

Insights from past documentation require human revie

Challenges

01	02	03	04	05
Spotty audit coverage	**Human bias**	**Lack of quantification**	**Report consolidation**	**Poor knowledge transfer**
Corpus is generally sampled, as there is too much paper to analyze manually.	Missed risks and controls because of reviewers' experience and education background.	Quantified overview is not available. Numbers are hard to generate, and so the 50,000 foot view is often misleading.	Lack of clarity on where to look for problems, or how to correlate disparate information sources.	Poor knowledge transfer, leading to silos of knowledge and fragile reliance on key staf.

The solution

Aggregate

Navigate

Accelerate

	Challenges	AuditMap Solution
01	**Audit Coverage Completeness**	Complete document set analysis
02	**Consolidated Report Investigation**	Search and AI-based analysis
03	**Ensure Best Practices**	Dynamic Risk & Control Matrices

AuditMap Functionality

Summarize documents

Organize statements and documents per audit program/risk area

Generate RCMs

Map control objectives to ERM Frameworks

Example User Capabilities

Map out risks and controls, per year, per program/risk area, and investigate

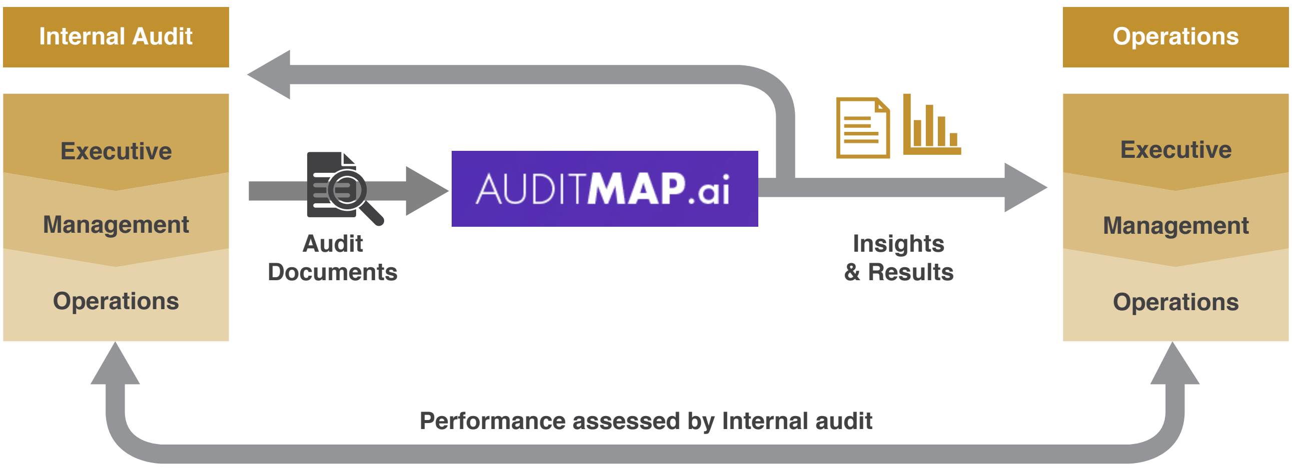
Infer list of previous controls; assess risk repetition and similarities

Integrating Risk Management Frameworks

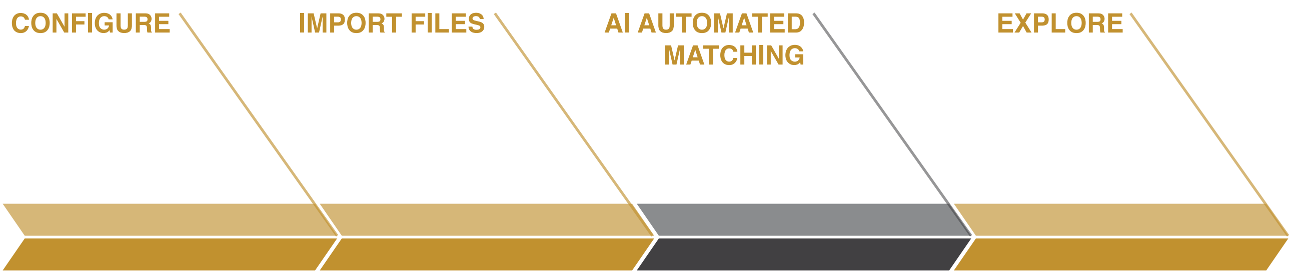
Changing frameworks quickly, enables users to view their reports in a new setting.

Novel avenues of risk can quickly be explored

The AuditMap Value Cycle



Getting Started - Workflow Overview



Define Audit Universe
Select a predefined audit universe to customize, or start from scratch.

Ingest and Analyze Corpus
AuditMap crawls a directory tree to ingest documents. Simply place the thousands of audit reports in one spot, and it does the rest.

Maps Key Statements to Audit Universe
Link risks to entities within audit universe, creating risk context. This step is AUTOMATED.

Expose Risk & Explore
Information presented to operator based upon the current activity, exposing insights.

How it works

The Multi-model Approach

Stallion.ai Process

Inputs

- Audit Universe
- Audit Reports

+

Stallion.ai Engine

- Live Indexing in NoSQL
- Tensorflow-based classifiers (and more)

↓

Audit Process Improvements

- Explore risks + controls
- Mapping risks + controls to documents + audit universe
- Explore Entity Relationships (Knowledge Graph)
- Fast Search (keyword, regex, etc.)
- Risks Classification
- Risk Similarity
- Entailment of risks and controls (future)

Book Your Live Software Demonstration

Intellectual Property

Augmenting Word Embedding Models With Expert Knowledge (DRAFT)

Mathieu Lemay[†], Daniel Shapiro[*†], Hamza Qassoud[*], Mary Kate MacPherson[†], Kieran Yee[†], and Miodrag Bolic[*]
[*]School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, Ontario, Canada
Email: {hqass076, mbolic}@eecs.uottawa.ca
[†] Lemay Solutions Consulting Inc., Ottawa, Ontario, Canada.
Email: {matt, daniel, marykate, kieran}@lemay.ai

***Abstract*—Pre-trained word embedding models transfer meaning to new text based upon the generality of original corpus. However, industry-specific terminology and nuances are lost to the general use cases of natural language. In this paper, we propose an approach for enriching existing word embeddings with domain-specific knowledge without re-training the embedding using a small custom word embedding.**

***Index Terms*—WORD EMBEDDING, EXPERTISE, KNOWLEDGE**

I. Introduction

WORD embedding is a technique for representing words using numeric vectors, which allows calculations to be performed on a "word" in comparison to all other words. In a good embedding, those calculations will map to semantic relationships; the canonical example being "king - man + woman = queen" [1].

Unfortunately, processing a large enough corpus to generate useful vectors is computationally expensive. Therefore, common practice for practical applications is to use a pre-trained word embedding model trained on a very large corpora such as Google News. However, these pre-trained embedding models do not generalize well to text from more specialized fields and industries, as the meanings of words can change in the context of a particular domain. For example, the word "collision" is usually closely associated with "car", "vehicle", "accident", and "insurance". In the context of computer networking, on the other hand, words such as "packet", "transmission", and "Ethernet" should be considered more relevant. In other words, a jack-of-all-trades embedding is master of none; but for many practical applications, such as real estate, financing, accounting, or computer science, we wish to have an expert solving the problem.

in this work, is to "enhance" the embedding generated by the general-purpose model by combining it with a second embedding generated by a separate, smaller, domain-specific model.

Our previous work investigated visual deep learning recommender systems, generating recommendations for a user based on what could be seen on the screen [2]. For example, if a user saw a picture of a dog and then used a search engine to search for "dog", the recommender system would learn to connect those two events [3]. The recommender system grouped events into contexts, such as "animals" or "colors". This work made use of a pre-trained Word2Vec model trained on the Google News corpus. By determining semantic distance between keywords, the system could classify previously unseen keywords into known contexts, or create a new context for the keyword. A problem arose when observing a keyword not recognized by the pre-trained model. The typical approach is to simply delete words from the corpus where they are not found in the pre-trained word embedding model. However, in this application that would mean failing to classify the onscreen text. In our previous work [4], an attempt was made to solve this issue by generating a new word embedding based around the keywords. Groups of the keywords were entered into a search engine programmatically, and the first 30 resulting documents were scraped into a corpus. This corpus was then utilized to train a new Word2Vec model with a much better understanding of the keywords. Although the results were promising, generating the new Word2Vec model using this method was computationally expensive, requiring 30 minutes of scraping and computation for only 9 keywords.

This work examines three widely-used word embedding models and explores the use of a secondary, smaller domain-specific embedding to improve their understanding of domain-

STALLION.AI

Contact Us:
hello@stallion.ai
+971 44 270774

Stallion Tech Inc.
900 Lady Ellen Place, Suit 114,
Ottawa, ON, KIZ 5L5, Canada
Contact: +1 (604) 781 9481

Office # 1104, Executive Tower D
(Aspect Tower), Al Abraj Street,
Business Bay, Dubai - UAE
Contact: +971 44 270774

hello@stallion.ai   www.stallion.ai