Expert Systems With Applications 202 (2022) 117166

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

A multi-input multi-label claims channeling system using insurance-based language models

Anuj Dimri, Arindam Paul, Deeptha Girish, Peng Lee, Sardar Afra *, Andrew Jakubowski

American Family Insurance, 821 E Washington Ave, Madison, WI 53703, USA

ARTICLE INFO

Keywords:
Insurance
Language models
BERT
Transfer learning
Claims classification
Fraud detection

ABSTRACT

Servicing claims, a time consuming and labor-intensive task, plays a pivotal role in how insurance companies serve their policyholders. Claims may not get routed early enough in the process to the correct team, leading to dissatisfied customers because of inefficient claim's management. Claims departments need to process substantial amount of structured and unstructured data to successfully route claims — a process referred to as channeling. The scope of the present work is limited to the auto insurance claims with a focus on four different downstream classification tasks including claims' fraud and bodily injuries. We propose a system that utilizes claims' notes and structured data to build machine learning models, which employ an insurance-based language model built by enhancing Google's BERT, to route claims to domain experts. The proposed channeling system successfully routes important claims to domain experts for additional review, which can substantially improve claims management and customer satisfaction.

1. Introduction

When an insurance policyholder suffers a loss, the process can be very stressful. Not only has the policyholder experienced a traumatic event, but they have to communicate about the event as well as continue following up with their insurance company to ensure their loss is covered and they can be properly indemnified. During this important and stressful time, how an insurance company manages policyholder claims is critical. According to J.D. Power & Associates, claims cycle time is a leading indicator of customer satisfaction (Effler, 2019). Claims cycle time is the time it takes to settle and close an insurance claim, from the day it is opened also referred to as the first notice of loss (FNOL). A primary motivation for insurance companies is to reduce claims cycle time and improve customer satisfaction. This can be achieved by developing an automated system that correctly routes claims to appropriate claims personnel.

Insurance companies offer services to their policyholders in a variety of domains like life, health, auto, home, commercial, etc. In this work, we focus on auto claims — which generally involve incidents related to vehicles and their occupants. When an auto policyholder suffers a loss, they report the incident to their insurance company providing a detailed account of the incident to a customer representative such as the date of the loss, loss location, what caused the loss, as well as what loss was suffered. Throughout the claims process, detailed notes are entered into an internal claims system by various groups such as the customer care center, claims adjusters, and special investigations unit (SIU). These detailed unstructured notes are referred to as claim notes.

Claims in their inherent nature contain diverse types of information covering a range of different situations. For instance, a claim could be a collision on a highway where the vehicle was severely damaged, a few people were injured and an attorney was hired. The same claim can have three different labels — total loss of the vehicle, bodily injury, and attorney retention respectively. Therefore, claims classification can be treated as a multi-label classification problem. Claims develop over a period of time and not all information is available on the day when the loss occurred. Especially, when multiple customers are involved the information can keep updating for a few days. For the majority of the time, only text information gets appended/updated. In some cases, when a claim has more than one potential label associated with it, domain experts (claim adjusters) from various areas may need to work together to quickly service the claim and help the customer.

This work is focused on four different claims problems. First, total loss — indicates whether a vehicle is to be deemed as total loss or it should be repaired. If a vehicle is considered a total loss, it is sent to salvage yard, otherwise if it is repairable, it is sent to body shops for repair. Claims that are incorrectly classified as total loss but the vehicle is actually repairable, can be very costly to the insurance company.

* Corresponding author.
E-mail addresses: adimri@amfam.com (A. Dimri), apaul@amfam.com (A. Paul), dgirish@amfam.com (D. Girish), plee@amfam.com (P. Lee), safra@amfam.com (S. Afra), ajakubow@amfam.com (A. Jakubowski).

https://doi.org/10.1016/j.eswa.2022.117166
Received 11 November 2020; Received in revised form 6 December 2021; Accepted 31 March 2022
Available online 14 April 2022

Second, bodily injury — indicates if there are any injuries associated with the claim or not. If yes, what is the severity of the injuries. Third, attorney retention — indicates if there will be any attorneys involved in the claim or not. Last, fraud investigation — indicates if a claim should be investigated for potential fraud or not. Based on the aforementioned problems, claims can be routed to appropriate domain experts (e.g., claim adjusters or SIU) who interact with the customers and help them throughout claims process.

To build predictive models both structured and text data (claim notes) are utilized. For text data, various deep learning techniques have been used by researchers. In this work, TFIDF (Ramos et al., 2003), word embeddings (Chalkidis & Kampas, 2019; Mikolov, Sutskever, Chen, Corrado, & Dean, 2013; Pennington, Socher, & Manning, 2014), LSTM based approaches and more recently developed language models have been used for experimentation. For the language model, BERT (Devlin, Chang, Lee, & Toutanova, 2019), a bi-directional language model built on general purpose corpus (Wikipedia and BooksCorpus (Zhu et al., 2015)) using transformer (Vaswani et al., 2017) based approach is used. BERT performs better than other word embedding based approaches for many different natural language processing (NLP) specific tasks. However, an insurance corpus is different from general purpose corpus. In our previous work (Dimri, Yerramilli, Lee, Afra, & Jakubowski, 2019), this issue was addressed by further pre-training the BERT model with enhanced insurance specific corpus and claim notes. It was observed that incorporating insurance-based domain knowledge into the language model performs better than using the language model out-of-the-box. We refer to this enhanced model as an Insurance-Based Language Model (IBLM). However, in Dimri et al. (2019) we explored a single-label (different models for different kinds of claims), single-input (text-based) approach. In this work, a multi-input strategy to incorporate both text and structured data is adopted to improve the efficacy of the models. As new text data gets appended in the first few weeks, models for day 1, 5 and 10 are built respectively. This ensures that a claim get scored as soon as it is opened (day 1) and also on further days when more up-to date information is available.

Ideally, claims where severe injuries or major accidents are reported should be addressed manually. After getting predictions from the model(s), claims are channeled by calculating a *claim score*. The *claim score* is calculated by weighing predictions based on the type of classification problem, model accuracy, days since the claim is opened, severity of the incident associated and other domain heuristics. For example, a claim which is a potential fraud or where attorneys are hired take a long time to get closed and can be costly to the insurance company, should have precedence over a claim which has minor injuries associated with it. Based on these domain heuristics, the claims are ranked. A select number are chosen for routing based on personnel capacity. Through the channeling system we envision to streamline the claims process and let the domain experts review a subset of claims which are important. This would not only help the adjusters to focus on important claims but also help the customers in a timely manner.

As previously noted, claims data changes with time. Models can be built off of data available at different days following the occurrence of the claim. We have observed that day 10 model performed better than day 5 and day 1 models. This is because day 10 model has more information available and the information is more up to date than previous days. The multi-input strategy (text + structured data) performed better than structured only and text only models for majority of problems. We refer to multi-input (text + structured) data model as combined model in the rest of the paper. When compared to their binary counterparts, the multi-label models performed equally well and in the case of fraud, multi-label outperformed the binary approach. The channeling approach introduced in this work calculates *claim score* using predictions, domain heuristics and injury severity information, has a high recall for claims which have multiple positive labels associated with them and are more important as compared to claims having single positive labels. A web application is also built which hosts the models, the channeling system and routes the claims to appropriate domain experts.

In summary, our contributions are as follows -

1. Trained multi-label models for different days using structured, text and combined data for four different insurance classification problems — Total loss, Bodily Injury, Attorney retention and Fraud investigation detection. Used insurance based language model to process the text data.
2. Compared the multi-label approach with an equivalent binary approach for combined model.
3. Trained a multi-class injury severity model using text data which helps in differentiating between bodily injury claims.
4. Designed a channeling approach which combines predictions from different models across days to calculate *claim score* and route the claims to domain experts.
5. Designed a web based user interface to serve the best model(s) and channeling approach to route claims to domain experts.

The rest of the paper is organized as follows - Section 2 talks about related work and Section 3 contains the overview of our complete channeling system. Details around data used in this work are presented in Section 4, followed by different approaches for modeling and channeling claims in Section 5. Results are presented in Section 6, followed by the Discussion Section. Section 8 contains details for our web based user interface. Conclusion is presented in Section 9.

2. Related work

Recent advances in deep learning have led to many state-of-the-art algorithms in various computer vision and natural language processing (NLP) tasks. Especially in text classification, counting techniques like TFIDF (Ramos et al., 2003), word embedding based approaches (Chalkidis & Kampas, 2019; Mikolov et al., 2013; Pennington et al., 2014) have been used as standard techniques but they lack contextual information. This was soon replaced by context-based embeddings (Peters et al., 2018) and then ultimately by language models (Devlin et al., 2019; Howard & Ruder, 2018). Language models involve a pre-training step which can be done on a large general-purpose domain corpus with one or more objectives and then using transfer learning, it can be further trained on context-specific target corpus for different downstream tasks. Further, pre-training the language models on target domain corpus has led to a further increase in the performance of several downstream tasks (Dimri et al., 2019; Lee et al., 2020).

There have been several applications of NLP and deep learning for claims handling problems. Kolyshkina and van Rooyen (2006) study the impact of textual information in claims cost prediction for an Australian insurance company. They utilized simple word count-based features in ensemble CART decision tree models and found that models that include textual information outperformed models without it. Popowich (2005) used a part-of-speech based NLP concept matcher on a corpus consisting of management notes, call center logs and patient records for identifying medical claims that require further investigation. Wang and Xu (2018) harnessed Latent Dirichlet Allocation (LDA) for extracting textual features, and then developed a multi-input model that utilized both text features as well as structured data for detecting claims fraud. Sabban, Lopez, and Mercuzot (2020) considered claims severity as a binary problem, where most claims are not severe. First, they utilized balanced bagging for oversampling rare severe claims to create a less imbalanced dataset. Then, they experimented with traditional ML techniques as well as Convolutional Neural Networks (CNNs) and LSTMs with FastText embedding for predicting if a claim is severe or not.

In our previous work (Dimri et al., 2019), we introduced IBLM which is a general domain language model (BERT (Devlin et al., 2019)) that is further pre-trained using insurance specific corpus with enhanced vocabulary. For downstream classification tasks, we used claim

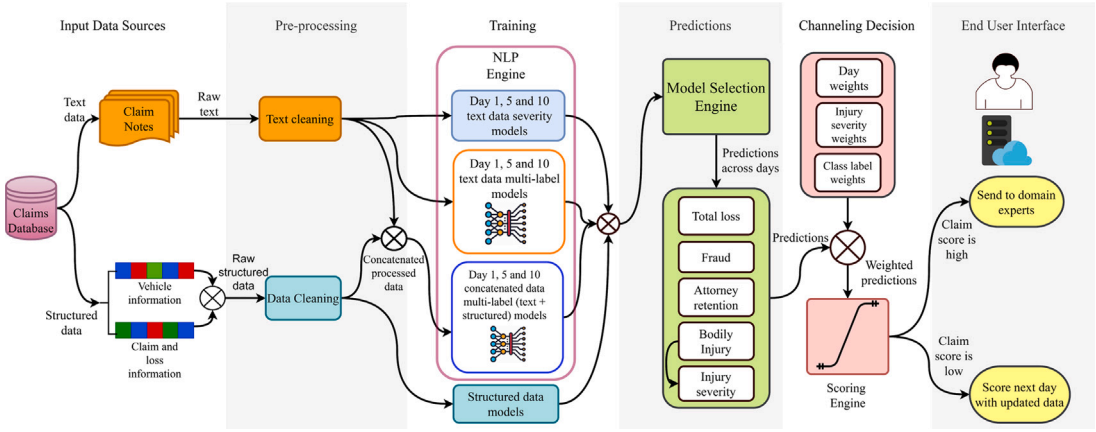


Fig. 1. Channeling System: Illustrates the flow of claims through the modules that constitute the claims channeling system. The best model is chosen across input types and modeling approaches. The predictions obtained from the best model is weighed using weights proposed by domain experts to generate the claim score in the scoring engine. Based on claim score, claims are routed to domain experts or scored next day with updated data.

notes only and considered different claims problems as separate binary classification tasks. As claim notes change over time, we built models for day 1 and day 10 only. In this work, we combine structured data with claim notes, treat claims as a multi-label classification problem, introduce total loss and claim severity as new tasks and increase the granularity of the models by training model for day 5 also. We also introduce claims channeling system which consumes the predictions from the different models as inputs which scores each claim based on which the claims are routed to the appropriate domain experts.

3. Channeling system overview

The information used as input to the channeling system comprises of both structured and unstructured data. Structured data includes vehicle, claim, and loss details. Unstructured data is in the form of claim notes which contain detailed information about the event that led to the filing of the claim and other details may continue to be added until the claim is closed. Using all the data mentioned, machine learning models are built and used to predict different labels associated with the claim and then route them to domain experts (claim adjusters) using our channeling approach, which would help the customers throughout the claim life cycle as shown in Fig. 1.

The input data is pre-processed before being fed into the channeling system. For claim notes, pre-processing includes removal of stop words, email addresses, unicode, and digits (phone numbers, numeric part of addresses, zip codes, etc.). For the structured data, categorical fields are converted into one-hot vectors. As mentioned previously, not all information is available on the day the claim is filed. Claims data evolves over time. This is especially true for claims notes. Therefore, different machine learning models are built for claims notes accumulated through day 1, day 5, and day 10. After day 10, the updates to claim notes are reduced drastically, so no models are built after day 10.

A claim can have multiple labels associated with it. As mentioned, in this work we limit our labels to total loss, bodily injury, attorney retention, and fraud investigation. Multi-label models are built for structured data, text data, and combined (text + structured) data for day 1, 5 and 10, respectively. If an injury is associated with a claim it is of prime importance to know the severity of the injury. The injury severity is a multi-class problem containing four classes — superficial, minor, moderate, and fatal. A multi-class classifier is built to predict the severity of the injury as well. The NLP engine shown in Fig. 1 is responsible for building all the text-based models.

After getting predictions from different models the best model across input types and modeling approaches is chosen using the model selection engine. The predictions from the best models across days are combined using domain-specific knowledge to get a holistic view of a claim. E.g., a claim having multiple labels associated with it is of more importance than a claim having a single label. Similarly, a claim having fatal injuries is more important than superficial injuries. Using domain knowledge, problem type, severity information, and days since the claim is opened, the scoring engine calculates a *claim score* for each claim. As the claims need to be reviewed manually by adjusters, only a small fraction of all the claims can be routed to them. *Claim score* is used to rank the claims in order of their importance. If it is above a certain threshold, they are routed to the adjusters. Else, the claim is scored the next day if it has updates in its inputs using the appropriate day model. The details of the complete system is shown in Fig. 1.

4. Data insights

In this section, we talk about the data used in this work in more detail. We focus on auto claims only which contains structured and unstructured data. The structured data mainly contains information related to the insured person and details around their insurance policy. The unstructured data is the raw body of text called claims notes, which contains information ranging from the description of the loss occurrence to information about various parties involved in the claim. It contains information about activities related to servicing the claim and much more. More importantly, claims notes hold rich information beyond what is captured in the structured data. In Section 4.1 we introduce Claims Life Cycle which describes detailed claims processing. Section 4.2 has insights around distribution of the data.

4.1. Claims life cycle

Fig. 2 represents the life cycle of a particular claim highlighting what information is appended and when since an incident is reported till the claim is closed. The exact details might vary for every incident. An example of the progression of claims notes with time for a particular incident is shown in Fig. 2. Once an incident is reported, immediately you might have vehicle information like year, make, model, color, registration details of the vehicles involved, the driver and passenger details, location, date, and time of the accident. Further, information around insurance and contact information of the other party involved in the accident is also populated. Sometimes, a police complaint is also filed. In a few days following the accident more information like a detailed description of the accident, witness statements, the speed, weather, and road conditions are added. Once a claim is filed and a claim adjuster (domain expert) is assigned to the case, more information regarding claims processing is added. For example, pictures of the accident, policy details of the people involved, police report number, etc. Vehicle inspection is also done which provides detailed

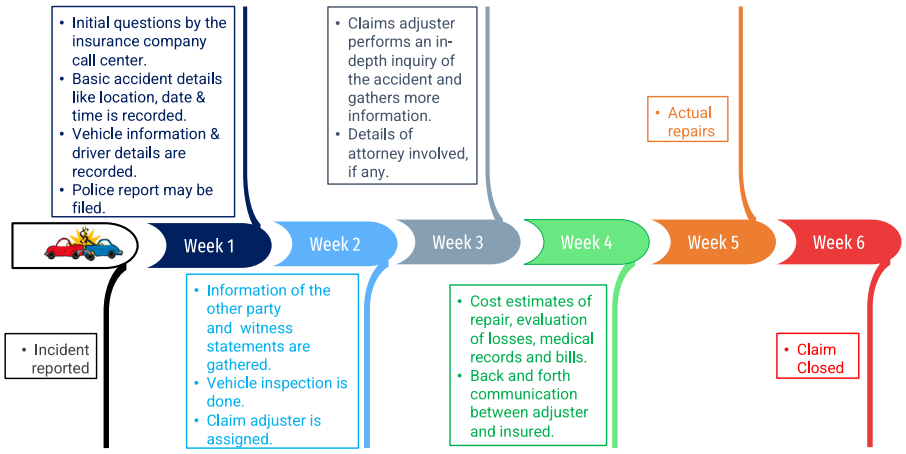


Fig. 2. Claims life cycle: An example of what data gets added during claim processing for one particular claim on a weekly basis.

Table 1
Binary label distribution for different claims classification problem.

Label	0	1
Total Loss	80.81%	19.19%
Bodily Injury (BI)	89.26%	10.74%
Attorney	93.28%	6.72%
Fraud	99.04%	0.96%

Table 2
Distribution of length of claim notes across days. As days increase, the length of notes also increases.

Quantile	Day1	Day5	Day10
5%	34	45	46
25%	151	185	189
50%	235	282	293
75%	340	485	573
95%	622	1104	1460
99%	959	1721	2266
100%	5603	7245	7245

Table 3
Class Distribution for different injury severity (BI Severity) claims.

Class	Percentage
Superficial	20.23%
Minor	62.78%
Moderate	13.91%
Complex/Fatal	3.08%

information on the damages done to the vehicle. Adjusters can ask for more details about the incident, leading to multiple back and forth communication between insured and adjusters. Claims notes are updated with this information making them more informative. As the processing progresses over days other information like cost estimates of repair, evaluation of the losses, medical records, and bills, details of the attorney involved if any might be added to the claims notes. Once actual repairs to the vehicle are done, all the payments are done and the claim is finally closed.

4.2. Data distribution

Insurance claims classification is a multi-label problem. Not only can a claim have multiple labels, but also having a positive class on any label makes it more likely to be positive for another. Therefore, for each claim, the claims handling process ascertains if a claim is a total loss or not, has injuries or not, requires attorney attention or not, and if it needs to be sent for fraud investigation, with 1 signifying if it is positive for that claim. Table 1 presents the class distribution of all the four labels. It is observed that the total loss label is less imbalanced than bodily injury (BI), which in turn is less imbalanced than attorney and fraud. As modern automobile design is geared towards passenger safety, during major crashes, the automobile may have total loss but there may not be any bodily injury to the passengers. Further, by their very nature, claims that require attorneys to be involved or require attention from the Fraud Special Investigation Unit (SIU) unit are less common but are very important for insurance companies. Moreover, a claim that requires an attorney likely involves a bodily injury, and a claim deemed to be handled by Fraud SIU is very likely involving some attorney investigation or BI or both.

All the datasets presented in this paper are for a 6 month period from July 2018 to December 2018. The dataset comprises of structured/tabular data and unstructured data in the form of claim notes. As claim notes evolve over time their lengths keeps on increasing. Table 2 depicts the quantiles of the number of tokens per claim across different days. We see that as days pass by the length of the notes keep increasing. However, the increase in length of claim notes is more from day 1 to day 5 as compared to day 5 to day 10. We observe that the distribution is very right-skewed as the length quantiles for 95% is very different from that of 99% which in turn is very different from the 100% - number of tokens in the longest claim.

As observed in Table 1, less than 11% claims have a positive BI label. Once injuries are associated with a claim, getting to know the severity of the injuries is of prime importance. This is because a claim involving a serious bodily injury or injuries involving hospitalization has to be handled with urgency compared to a claim with a superficial injury. The BI severity consists of four different classes and their distribution in the ascending order of severity of injuries is presented in Table 3.

5. Methodology

This section is divided into two parts — Modeling approach and Channeling approach. Modeling approach contains the approaches taken to build different machine learning models based on structured, text and both of them combined. The methods used to combine predictions from models to get a *claim score* to rank the claims based on their importance is discussed in the channeling approach subsection.

5.1. Modeling approach

As claims data comprises of both structured (tabular) and text data, they are modeled separately as well as together (combined). For

structured only data, the performance of the XGBoost model (Chen & Guestrin, 2016) and a two-layer dense network was compared. For text-only models, TFIDF, LSTM, BiLSTM, and language models were built and compared. Different configurations, architectures of the above models were evaluated and the hyperparameters were tuned before finalizing them. We tried random search for tuning different set of hyperparameters for different approaches presented in this work. We chose the best set of hyperparameters to perform our final set of experiments. Details are in 6.

The first approach considered for structured data is XGBoost (Chen & Guestrin, 2016). It is an ensemble learning approach in which the resultant strong classifier aggregates the outputs from multiple weak classifiers. XGBoost implements the gradient boosting algorithm for decision trees. The second approach used to model structured data is a two-layer dense network.

In this paper, for text data, logistic regression is used as the machine learning algorithm and TFIDF statistics are used as input features. TFIDF (Term Frequency Inverse Document Frequency) (Ramos et al., 2003) calculates the product of the frequency of a given word in a document with the inverse document frequency. GloVe (Global Vectors) (Pennington et al., 2014), a popular word embedding technique along with LSTM and BiLSTM is also used in this study and the results are compared. GloVe incorporates global statistics (word co-occurrences) along with local statistics to generate word vector representations. LSTM and BiLSTM are types of Recurrent neural networks (RNNs) (Hochreiter & Schmidhuber, 1997) have connections that have loops or recurrent connections which allow for feedback and memory to the networks over time. This makes them effective for sequence prediction problems such as text classification.

More recently, language models have been used in many NLP related tasks and have achieved state of the art performance. Language models captures complex features of the corpus and take context into account when generating embeddings. In this work, BERT (Devlin et al., 2019) is used, which is a bidirectional language model based on Transformers (Vaswani et al., 2017) with two pre-training objectives. First, masked language model (MLM) which randomly predicts masked words in a sequence. Second, next sentence prediction (NSP) which predicts whether one sentence follows another in the corpus. BERT is trained on a general purpose corpus using English Wikipedia and BooksCorpus (Zhu et al., 2015).

As insurance corpus is very different from general purpose corpus, BERT is further pre-trained using insurance corpus with enhanced vocabulary in our previous work (Dimri et al., 2019), and referred to as IBLM. Once the language model is pre-trained, it can be used for different insurance classification tasks with slight architecture changes. IBLM (Dimri et al., 2019) is fine-tuned in a multi-label fashion for our problem across different days.

Another model is trained that combines text and structured data together. The detailed model architecture is shown in Fig. 3. The text branch is comprised of an IBLM, followed by a linear layer which outputs a 500 dimensions vector. The structured data branch comprises of a two layer dense network. The outputs from the text branch (500 dimensions) and structured data branch (100 dimensions) are concatenated together and passed through a linear layer (500 + 100 = 600 dimensions). The final output consists of four neurons, each corresponding to a different label (total loss, bodily injury, attorney retention and fraud). Each layer is followed by a *tanh* activation layer. Different number of layers, activation functions, number of neurons were considered before finalizing the above architecture. Further, there is an option of training the complete model as shown in Fig. 3 or to treat the text branch as a fixed encoder and not train it by freezing its weights.

We stick with a two layer network in the structured branch because our main aim is to find out whether combining text and structured data together will have better performance on different insurance classification tasks than having separate text model and structured data model. Further from an application point of view, deeper networks will increase latency of the model at prediction time. As our combined model already has a language model with a 12 layered transformer network, adding more layers both to the structured branch as well as to the concatenated combined model will make the model deeper thereby further increasing latency.

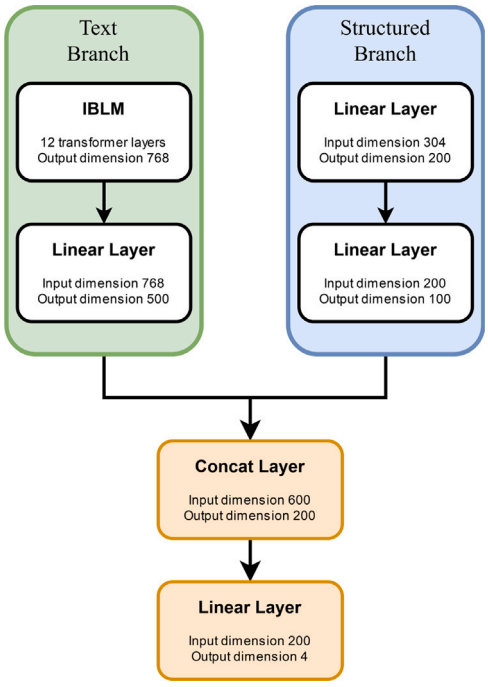


Fig. 3. Combined model architecture : combines the text data branch and structured data branch. The text branch consists of Insurance Based Language Model (IBLM) followed by a linear layer. The structured data branch consists of a two layer network.

The implementation was done in pytorch (Paszke et al., 2019). *BCEwithLogitsLoss* was used as the loss function. Pytorch combines BCELoss (Binary Cross Entropy Loss) and Sigmoid Layer into a single class called BCEwithLogitsLoss (Binary Cross Entropy with Logits Loss) (Stevens, Antiga, & Viehmann, 2020). BCEwithLogitsLoss has been utilized in several recent publications (Hande, Puranik, Priyadharshini, Thavareesan, & Chakravarthi, 2021; Lewis, Mahmoodi, Zhou, Coffee, & Sizikova, 2021; Melekhov et al., 2019).

During testing/inference sigmoid function is applied to the logits (outputs from the model) to convert them to probabilities. For comparison, binary models are also trained and compared to their multi-label counterparts following the same architecture but changing the corresponding loss functions.

If a claim has an injury associated with it (i.e., the claim is BI-positive), getting to know the severity of the injury associated with the claim is very important. To achieve this, an injury severity model is built. A BI-positive claim can have one of the four injury classes — superficial, minor, moderate and complex/fatal, therefore the injury severity problem is treated as a multi-class classification. The structured data does not contain much signal about the severity of the injury. However, the text data (claim notes) contains the details about the loss incident and has signal about the severity of the incident and corresponding injuries. So claims notes only are used to build the injury severity model. It is observed in Table 1, only 10.74% of all claims are BI-positive claims. For claims which are not BI-positive, there is no notion of severity of injuries. Therefore, instead of building a severity model using all claims data where the severity class labels would be rare, a hierarchical approach was used. First, it is predicted if a claim is BI-positive or not using the models described above, and then for

claims that are predicted to be BI-positive, the multi-class BI-severity model is used to predict the severity class for that claim.

As claims data changes with time, the same model architectures are used to build models for day 1, day 5 and day 10 using the data which is available till the day when model is trained.

5.2. Channeling approach

After getting predictions from the models based on different days only a subset of claims are selected, which can be reviewed manually by the adjusters. Each day new claims are opened and old claims are updated. On any given day, the idea is to select claims across different days and different classification problems. This ensures that importance is given not only to new claims but also to old claims where updates have been made for each problem type. With time, as more information gets associated with the claim the models make better decisions. To select a subset of claims, they are ranked. A *claim score* approach is proposed in which the score of a claim is calculated based on model predictions across days and domain specific heuristics.

$$claim\ score = f(Prediction, label\_type, day, severity)$$

1. Approach I — In this approach, for each claim the sum of the probabilities of the four problem types is computed and then they are sorted in decreasing order. For different days, a simple average across all days is used.
2. Approach II — The problem with previous approach is that all four problems have different distributions and taking a simple average would have negative effects. To solve this, for each problem type, the probabilities are changed to percentiles and then the average of the percentiles is used.
3. Approach III — In Approach II, all the different problems and different days models are treated equally. This is not true in the real world, so domain specific weights are assigned to each problem type. For e.g., fraud and attorney are assigned a higher weight than total loss. Similarly, as data gets updated with time, day 10 model makes predictions on more data as compared to day 5 and day 1. So day 10 model is assigned with highest weight followed by day 5 and then day 1.
4. Approach IV — The previous approach, considers all the bodily injury claims equally but not all injury claims are of same severity. So multi-class severity model predictions are used and different weights are assigned to all injury claims based on their severity probabilities. The severity weights are as follows — fatal injuries are assigned the highest weight followed by moderate, minor and superficial injuries.

Ideally, claims which have multiple positive labels should be reviewed manually. For e.g., a claim which has both attorney and bodily injury associated with it should have a higher *claim score* than a claim which has only attorney as the positive label. Further, a claim with fatal bodily injury should have a higher score than minor bodily injury.

6. Results

In this section, the results for all the experiments are discussed. This section is divided into two parts — modeling results and channeling results. In the modeling results subsection, results for the different machine learning modeling approaches are discussed. In the channeling results subsection, the performance metrics of the different channeling schemes proposed in this work are compared.

6.1. Modeling results

For all experiments six months worth of claims are used. The first four months are used for training, the next half month is used for validation and the last one and a half months are used as out of time (OOT) test data. We used a total of 170K claims. Our training set, validation set and OOT test set have 115K, 15K and 40K claims respectively. All results presented are on the OOT test data.

Table 4
Comparison of AUC scores of 2 layer dense neural network and XGBoost model. XGBoost outperforms the neural network. The best results have been highlighted.

Model	Day 1		
	Macro	Micro	Weighted
2 layer dense network	0.858	0.919	0.893
XGBoost	**0.881**	**0.924**	**0.907**

6.1.1. Multi-label classification

Four different claims problems are considered — total loss, bodily injury, attorney retention and fraud investigation detection, where each classification problem has a label 1 or 0. For multi-label classification, we present area under the receiving operating characteristic curve (AUC) with three different aggregation methods: micro, macro and weighted. Micro calculates AUC globally by considering each element of the label indicator matrix as a label. Macro calculates AUC for each label, and finds their unweighted mean without taking class imbalance into consideration. Weighted calculates AUC for each label, and finds their weighted average (Pedregosa et al., 2011).

For structured data, XGBoost and a 2-layer dense neural network are trained and their performance is compared. Random search was used for hyperparameter tuning (Bergstra & Bengio, 2012). For XGBoost, the hyperparameters were number of estimators, maximum depth and lambda (L2 regularization term) which were found to be 100, 6 and 1 respectively. The 2-layer neural network has the following parameters: 256 dense units in each layer, batch size 128, gradient descent with learning rate 0.001 and momentum 0.9 and dropout of 0.2, which were also tuned using random search. The results are presented in Table 4. It is seen that the XGBoost model performs better than the 2-layer network in all the above mentioned AUC aggregation methods. The structured data remains the same for day 1, day 5 and day 10, therefore, the results do not vary with time in this case.

For text data, TFIDF with logistic regression, LSTM, BiLSTM and IBLM are used for our experiments. In the first method, text data is vectorized using the TFIDF vectorizer and classification is performed using logistic regression. Hyperparameter tuning is done jointly for TFIDF and logistic regression using random search. The main hyperparameters were maximum number of features, document frequency, n-gram range, maximum iterations and L2 regularization parameter. The LSTM model consists of an embedding layer, followed by LSTM layer, global average pooling, a dense layer, dropout and an output layer with sigmoid activation. Adam optimizer with learning rate 0.001 and batch size of 128 is used. The BiLSTM model has the same structure as the LSTM model but it has two stacked BiLSTM layers. For the language model, IBLM is used and a linear layer is added with four output neurons at the end corresponding to four labels. Learning rates in the range 1e-2 to 2e-5, warmup steps, weight decay are some of the hyperparameters used. The batch size was fixed to 32 and Adam optimizer was used. The results of the different models trained on text data only are presented in Table 5. As text changes with time, and more information is appended after day 1, so we train models for day 5 and 10 also. It is seen that IBLM outperforms all the models followed by LSTM, BiLSTM and TFIDF. These results support the fact that language models give the best performance. It is observed that AUC are better at a later date (day 10 is better than day 5 and day 5 is better than day 1) and this pattern is consistent across all the models. This implies that using text from a later day is better for decision making as it is more complete, relevant and up to date.

Further, structured and text data are combined and a combined model is trained as shown in Fig. 3. Two different set of experiments are performed. First, the text branch of the combined model is treated as a fixed encoder by not training it. Second, the whole combined model is trained end to end. This is done for different learning rates. The rest of the hyperparameters are same as text only models for IBLM. The results

Table 5
AUC scores for models trained on text data only in a multi-label classification approach. We observe that IBLM performs better than other deep learning approaches across all the days. The best scores are highlighted.

Model	Day 1			Day 5			Day 10		
	Macro	Micro	Weighted	Macro	Micro	Weighted	Macro	Micro	Weighted
TFIDF	0.849	0.856	0.838	0.907	0.913	0.903	0.932	0.941	0.903
LSTM	0.870	0.909	0.868	0.909	0.940	0.916	0.930	0.956	0.941
BiLSTM	0.847	0.894	0.851	0.888	0.931	0.907	0.915	0.951	0.936
IBLM	**0.897**	**0.927**	**0.895**	**0.931**	**0.955**	**0.937**	**0.946**	**0.968**	**0.956**

Table 6
AUC scores for different neural network configurations for multi-label combined model. When IBLM is treated as a fixed encoder and not trained further we see higher scores for all the days. The best scores are highlighted.

Model	IBLM trained	Learning rate	Day 1			Day 5			Day 10		
			Macro	Micro	Weighted	Macro	Micro	Weighted	Macro	Micro	Weighted
IBLM + 2 layer n/w	No	1e−3	**0.902**	**0.938**	**0.919**	**0.926**	**0.956**	**0.945**	**0.942**	**0.967**	**0.959**
IBLM + 2 layer n/w	Yes	1e−3	0.875	0.926	0.903	0.875	0.924	0.902	0.874	0.925	0.903
IBLM + 2 layer n/w	No	2e−5	0.881	0.931	0.909	0.913	0.948	0.931	0.927	0.956	0.944
IBLM + 2 layer n/w	Yes	2e−5	0.883	0.925	0.914	0.913	0.950	0.944	0.932	0.959	0.959

Table 7
AUC scores across different labels for the best multi-label models for structured only, text only and combined (structured + text) data. For instance, combined model for Total Loss performs best for all the days, but for BI and Attorney text models perform better. However, for Fraud, on Day 1, combined model performs better but as claim notes get updated, text only model performs better for later days.

	Day 1			
Model	TL	BI	Attorney	Fraud
Structured	0.932	0.878	0.879	0.836
Text	0.875	**0.924**	**0.921**	0.868
Combined	**0.934**	0.900	0.907	**0.868**
	Day 5			
Structured	0.932	0.878	0.879	0.836
Text	0.926	**0.957**	**0.948**	**0.894**
Combined	**0.954**	0.942	0.934	0.874
	Day 10			
Structured	0.932	0.878	0.879	0.836
Text	0.948	**0.972**	**0.964**	**0.902**
Combined	**0.965**	0.956	0.952	0.893

Table 8
Comparison of AUC scores between binary and multi-label models for each of the four claims classification problems (TL, BI, Attorney and Fraud). For each input type, only a single multi-label model is trained to predict all four claims classes, whereas four distinct binary models (one for each claims type) are trained. It is observed that Fraud has higher AUC score when trained in a multi-label way as compared to binary as it is able to learn from other labels which is not possible in binary classification. The best models for both binary and multi-label classification are highlighted.

	Day 1				
Classification	Model	TL	BI	Attorney	Fraud
Binary	Structured	0.927	0.877	0.873	0.812
Binary	Text (LSTM)	0.852	0.898	0.890	0.807
Binary	Text (IBLM)	0.874	**0.922**	**0.916**	0.852
Binary	Combined	**0.936**	0.902	0.905	**0.853**
Multi-label	Structured	0.932	0.878	0.879	0.836
Multi-label	Text (IBLM)	0.875	**0.924**	**0.921**	0.868
Multi-label	Combined	**0.934**	0.900	0.907	**0.868**

of the different combined model configurations are in Table 6. The best results have been highlighted. It is observed that the best results are for the configuration when the text branch is not further fine-tuned and is treated as a fixed encoder. The learning rate used is 1e-3. which was used for structured only experiments and gave the best results. Another advantage of treating the text branch as a fixed encoder is that training is also faster as the weights of the language model (IBLM) are not further updated. Given structured data chosen is mostly static across time, the trend of day 10 performing the best, then day 5 and then day 1 holds in the combined model as well.

By comparing the best models for structured, text and combined, it is seen that combined model performs the best, followed by text models and then structured models. However, it is insightful to view how the multi-label combined model performs on the four individual problem types: (total loss (TL), bodily injury (BI), attorney retention (Attorney) and Fraud). Table 7 contains the AUC of the multi-label combined model applied to each individual problem types for day 1, day 5, and day 10. It is noted that the multi-label combined model for total loss outperform the multi-label text models. However, for the other labels especially attorney and BI, that is not the case. There are two factors influencing this. First, the features for training total loss structured model include detailed vehicle information including vehicle damage indicators that describe if a certain part of the vehicle was damaged or not. As a vehicle may be recommended for total loss when there is a damage to multiple parts of it, these features are strongly correlated to total loss indicator. This is not the case for the other labels as the features for structured models are not as directly related to the final label and majority of signal is embedded in the claim notes. Hence adding structured features can have a subtractive influence on the performance of the combined model especially for bodily injury and attorney. Second, the text models improve greatly as more notes are incorporated, especially for labels such as attorney and fraud that are difficult to predict based on FNOL (or day 1) information, and as we incorporate more recent details, performance of the models improve. As the structured models are based on FNOL information, they are unable to improve.

Fig. 4 contains the precision–recall curves for all days for each problem type. The improvement is evident across days, however, improvement from day 5 to 10 is less as compared to day 5 from day 1. This is because a large portion of the claim notes are entered within first week so the amount of text data available for training and prediction increases more from day 1 to day 5, than day 5 to day 10. The updates are even less after day 10, so no models are built after day 10, instead day 10 models are used for all days after 10. For fraud, day 1 tends to not have enough information to determine whether a claim should be investigated by the Special Investigation Units (SIU). However, by day 5, a large percentage of suspicious claims will have been flagged leaving a smaller number to be flagged, so the performance of day 5 and day 10 models are similar in that regard.

6.1.2. *Multi-label vs Binary label classification*

In this section, instead of a single multi-label classifier, four different binary classification models are trained. Table 8 presents this comparison for day 1 only. We limit the comparison to day 1 only as we are interested in knowing which modeling technique (single multi-label

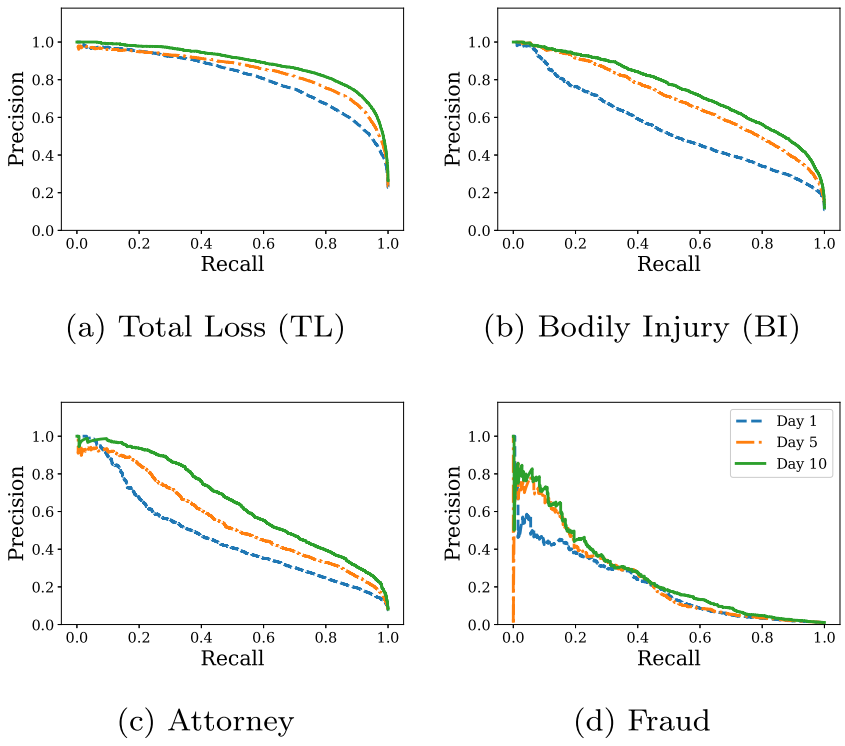


Fig. 4. Precision–recall curves for multi-label models when trained using combined data across days for each problem type. As the claims get updated with more recent notes over time, the performance of the models improve.

classifier or four different binary classifiers) is better at predicting the insurance classification tasks and assigning the claims to the adjuster on day 1 itself (early in the life cycle of the claim). We could follow the same process with day 5 and day 10 models but in that case we would need to wait for five and ten days respectively to get model predictions and assign the claim to the claim adjuster which could cause delays in claim processing.

For the binary classifiers, the model architecture is the same except for changes in loss function and number of neurons in the output layer. In terms of AUC, it is observed that total loss and bodily injury combined binary models do better than their multi-label counterpart. For attorney and fraud it is opposite. The decrease in fraud when trained in a binary classification way, is higher as compared to all the three different problems. This indicates that when trained in a multi-label way the fraud labels take advantage of other labels as well leading to a higher AUC.

Fig. 5 contains precision–recall curves for multi-label and binary classification models for each problem type for day 1. The values are similar to each other but improvement is observed in fraud. This supports the fact that fraud labels indeed take advantage of data from other problem types when trained in a multi-label way which is not possible when trained in a binary fashion. Also, just by using one multi-label model similar results are obtained and they are better in case of fraud than four different binary classification models. Similar trend is expected for day 5 and day 10 also, as with more data available the multi-label approach would benefit even more than the binary approach.

6.1.3. Multi-class BI severity classification

The structured data does not contain much signal with regard to the bodily injury and their corresponding severity. The majority of the injury related details are present in the claim notes. Recall in Table 7, it is observed that text-only model outperformed structured and combined models for bodily injury (BI) claims. Due to these reasons severity models are built using text data only. Similar to the aforementioned text-based models, TFIDF with Logistic regression, LSTM, BiLSTM and IBLM is used for our experiments. In addition, a 1-dimensional convolutional neural networks (CNN) is also built as they often outperform LSTMs on smaller datasets. It is also noted that the dataset for BI severity classification is a subset of the dataset used for previous experiments as BI-positive claims are 10.74% of all claims, and therefore it is much smaller in size. For TFIDF with logistic regression, joint hyperparameter search is performed to get the optimal set of hyperparameters. The 1-D CNN model consists of an embedding layer, followed by two sets of successive 1-D CNN layer and 1-D max pooling layer, a flatten layer (for flattening the input), a dense layer and an output layer with softmax activation. The LSTM, BiLSTM and IBLM models are similar to the models used in text-based models for binary classification other than changing the activation from sigmoid to softmax.

Table 9 describes the results of the multi-class BI severity experiments. It is observed that IBLM outperforms the other methods for day 1, and logistic regression using TFIDF performs second-best. However, for day 5 and day 10, the logistic regression using TFIDF outperform the other models including IBLM models. The reason for this could be that severity classification is strongly linked to certain words used in the claims — claims that have severity as superficial would have very different words used than ones used in complex/fatal claims. Also, as more claim notes are added after the first day, it is likely that the newly added words further differentiate the severity of the claim as the claims adjuster is closer to ascertaining the true severity of the claim. Further as the dataset is significantly smaller, it is unsurprising that a simpler method such as logistic regression would outperform more complex deep learning methods. It is noteworthy that even on a smaller dataset, IBLM models are able to outperform other deep learning models across all days which proves the overall efficacy of domain-specific language models.

6.2. Channeling results

In this subsection, the results of different approaches discussed in Section 5.2 are presented. The aim is to route claims that have multiple

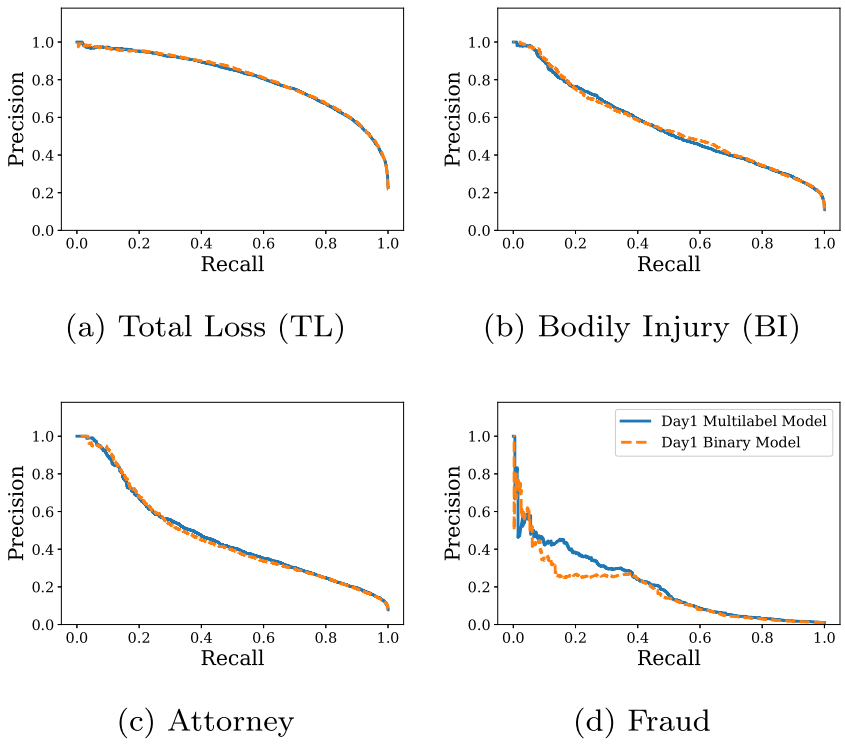


Fig. 5. Comparison of Precision–Recall values for Combined model for Day 1 only when trained in multi-label method versus binary-label method. The performance is similar for Total Loss, Bodily Injury and Attorney classification. However, for Fraud, the multi-label approach has a better performance as it is able to jointly learn from other labels which is not possible in binary classification.

Table 9
Comparison of AUC scores for different approaches for BI Severity (using only text data). On Day 1, IBLM performs the best whereas, for Days 5 and 10, TFIDF outperform other models. The best scores are highlighted.

Model	Day 1 auc		Day 5 auc		Day 10 auc	
	Macro	wt	Macro	wt	Macro	wt
TFIDF	0.658	0.613	**0.708**	**0.652**	**0.717**	**0.658**
CNN	0.628	0.584	0.658	0.609	0.645	0.601
LSTM	0.529	0.516	0.510	0.527	0.528	0.513
BiLSTM	0.545	0.531	0.536	0.531	0.525	0.521
IBLM	**0.684**	**0.634**	0.682	0.633	0.674	0.623

positive labels to domain experts as they are more important than single positive label claims. Let $\mathbf{S}_i$ define the set of all claims with i-positive labels and $\mathbf{S}_i^j$ represents an event in which j claims with i-positive labels has been routed to domain experts. Then the probability of such event, denoted as $\mathbf{Pr}(\mathbf{S}_i^j)$ and referred to as recall, is given by $\mathbb{E}[\mathbf{I}_{\mathbf{S}_i^j}]$ where $\mathbb{E}$ is the expectation and $\mathbf{I}_{S_i^j}$ is an indicator function, which is one if event S_i^j occurs and zero otherwise. Note that j is a function of volume threshold based on domain expert capacity. Ideally, at a certain volume threshold, a high recall for claims with multiple positive labels ($i \in \{2,3,4\}$) and a low recall for claims with single and no positive labels ($i \in \{1,0\}$) are desired.

Table 10 contains the results for different approaches used to calculate *claim score*. Based on *claim score*, top 10% of claims only (volume threshold) dependent on capacity of domain experts is selected. Predictions for combined models day 1, day 5 and day 10, together are used to calculate the *claim score*.

Each column in the table represents the recall for number of positive labels. $recall_0$ column presents recall of claims with no positive labels, $recall_1$ presents recall of claims with one positive labels and so on. It is desired to have lower values for $recall_0$ and $recall_1$ as they are less important and higher values for $recall_2$, $recall_3$ and $recall_4$ for a particular volume threshold. It is observed that Approach IV which takes into

Table 10
Recall metrics for different channeling approaches for all predictions till day 10 at 10% volume threshold. Approach IV outperforms all other approaches for claims which have multiple positive labels ($recall_2$, $recall_3$ and $recall_4$) which is desired. The best scores are highlighted.

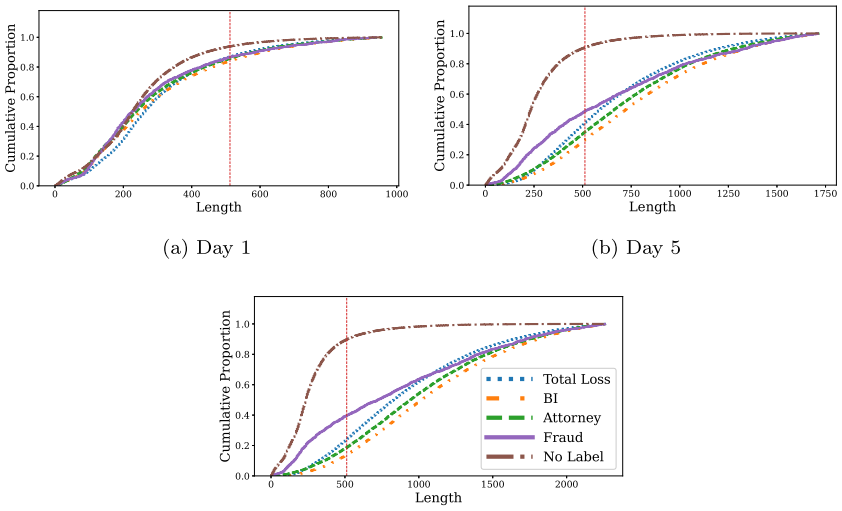
	$recall_0$	$recall_1$	$recall_2$	$recall_3$	$recall_4$
Approach I	0.004	0.219	0.584	0.756	0.730
Approach II	0.010	0.202	0.567	0.775	0.730
Approach III	0.013	0.189	0.592	0.710	0.800
Approach IV	0.009	0.179	**0.663**	**0.783**	**0.870**

Table 11
Recall metrics for channeling approaches for all predictions of Day 1 only at 10% volume threshold. Approach IV performs best for claims that have 2 or 3 positive labels but not for all positive labels (4). On Day 1, the claims data is not complete and hence a drop in performance. As claims get updated with time Approach IV performs best as shown in Table 10. The best scores are highlighted.

	$recall_0$	$recall_1$	$recall_2$	$recall_3$	$recall_4$
Approach I	0.013	0.227	0.488	0.680	**0.600**
Approach II	0.018	0.219	0.450	0.670	0.530
Approach III	0.026	0.194	0.470	0.610	0.400
Approach IV	0.020	0.185	**0.543**	**0.688**	0.530

consideration day weights, label weights, severity information based on ranking of predictions gives best results. Recall for multiple positive labels ($recall_2$, $recall_3$ and $recall_4$) are highest for Approach IV, whereas $recall_0$ and $recall_1$ are on the lower side. This ensures that the claims routed to domain experts for manual review would contain multiple positive labels and are deemed to be important.

It is observed based on the results of the previous subsection that day 10 model performs best as the claim notes are more complete compared to day 1 or day 5. However, it is important to understand and contrast the performance of different channeling approaches on the day the claim is opened compared to the results of the model incorporating data from all the days (Table 10). For this, day 1 combined model is



(a) Day 1 (b) Day 5

(c) Day 10

Fig. 6. Cumulative histograms for sequence length of claims with positive Total Loss, BI, Attorney, Fraud labels and no positive labels for day 1, 5 and 10 respectively. The red dashed line at 512 represents the maximum sequence length of the language model. Claims with no positive labels are shorter in length than claims with positive labels. Positive Fraud claims are shorter in length than other positive labeled claims.

used. It was noted earlier that day 1 performance is lower than day 10 as day 1 notes are not complete. Table 11 depicts the recall metrics for 10% volume threshold for day 1 combined model. As expected, the recall metrics are lower than Table 10. For $recall_2$ and $recall_3$, Approach IV performs the best. However, for $recall_4$, Approach IV does not performs the best. This is because claims needs to be updated with more information. Tables 10 and 11 gives a range of recall values to expect from different models based on claim age.

7. Discussion

7.1. Length analysis of claims notes

Figs. 6(a)–6(c) show the cumulative histogram plots of the sequence lengths of the claims on day 1, 5 and 10 respectively for all the claims belonging to one of the four classes as well as claims with no positive labels. Only up to the 99th percentile is considered to avoid large anomalies in the last percentile as observed in Table 2.

It is observed that there is no observable difference for the cumulative histograms across the different labels for day 1. However, it is also observed that more than 80% of claims with no labels have sequence lengths less than 512 for days 5 and 10 while claims with positive labels have a much smaller percentage of claims shorter than 512. This demonstrates that there is a distinct difference between claims with positive labels as compared to claims with no labels, and this is essentially because more notes are added on average in claims with positive labels compared to claims with no labels.

Moreover, among the different claim types with positive labels it is observed that claims which have a positive label for fraud have more than 40% of the claims with sequence lengths shorter than 512 as compared while the other positive classes have a much smaller % of claims shorter than 512. This is because fraud claims are often less detailed than other positive claims.

7.2. Volume thresholds

All the claims cannot be reviewed manually and only a small volume of claims can be routed to adjusters for manual review. Historically, these volumes have been 10% for total loss, 8% for BI and attorney and 2% for fraud. To get an idea of how well the models perform in terms of precision and recall the volume thresholds are set around historical averages. Fig. 7 contains the precision and recall values for different volume thresholds for each problem type. The model used here is day 10 combined model. The general trend is, as volume threshold is increased the precision drops and recall increases. It is desired to have a high recall (do not want to miss on positive labeled claims) but have to limit to a small volume. For total loss, the volume thresholds are 8%, 10% and 12%. In general, the precision is high for total loss for all volume thresholds and recall increases with increase in volume. For BI and attorney, volume rates are 6%, 8% and 10%, for fraud, volume rates are 1%, 2% and 4%. For total loss, high precision but low recall is obtained as there are a large amount of total loss claims and we limit the volume threshold around 10% only. The same is true for bodily injury claims as well. However, for attorney and fraud, the precision is on the lower side and recall values are higher. These two classification problems take longer to process as they require additional details like law court details and police reports, both of these are not a part of claim notes. For attorney, only 31% of claims had the final label assigned within 10 days, the corresponding number for bodily injury is 61.8%. Fraud labels also take a long time to get assigned because of investigation details. This makes attorney and fraud difficult to predict within first 10 days than total loss and bodily injury claims, leading to lower precision and recall values.

8. Web based user interface

A web application is built as a front end to the proposed channeling system. The application enables end users (claims adjusters, SIU investigators) to retrieve the claims that meet the *claim score* thresholds based on the channeling mechanism. In addition, the application gives flexibility to the end user to write in the experience of the customer from claim notes and look at the predictions in real time. Further, some of the structured data comes from third party data sources and can take time to get updated. As the text specific model results and combined model results are similar in Table 7, both text only and combined models are deployed in the web application. The two modes are described in more detail as follows-

1. Batch — In this mode, all the new and updated claims get scored by the combined model (as both text and structured data are available) on a daily basis. Then the channeling mechanism would generate the *claim score* for each claim and route the claims above a certain threshold to the appropriate adjusters for manual review.

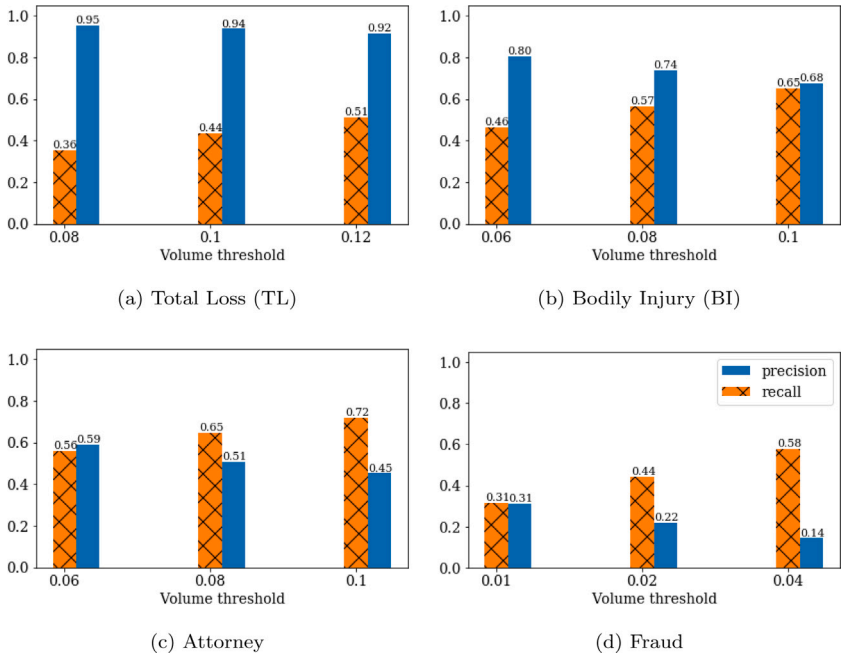


Fig. 7. Precision and Recall values for different volume thresholds around historical averages based on domain experts availability for each claims classification type for day 10 combined model. As volume thresholds increase, the precision decreases and recall increases.

Table 12
Macro AUC scores for different models across days.

Macro auc	Day 1 model	Day 5 model	Day 10 model
Day 1 data	0.902	0.871	0.869
Day 5 data	0.911	0.926	0.926
Day 10 data	0.912	0.938	0.942

2. Interactive — In this mode, the adjusters can just provide the claim notes and see the predictions. Further, the *claim score* is also calculated and based on the score the adjuster can know whether this claim would have been above the threshold or not. This feature is really helpful as sometimes multiple text updates are made in the same day so the adjuster can have access to results in real time and not wait for next day to get the results from batch mode.

It is important to route distinct types of claims to corresponding adjusters with specific expertise. In order to achieve this, the contribution of each classification problem in *claim score* is calculated and the claim is routed to the adjuster who has expertise corresponding to the problem with maximum contribution. However, due to multi-label nature of the claim, more than one classification problems might have significant contribution in calculating the *claim score*. In such cases, the claim is routed to different adjusters having relevant expertise. The adjusters can then keep in sync, share details and insights related to the claim with each other and service the claim quickly.

Considering models are built for day 1, 5 and 10 only, a good question is which model should we use on day 2, 3, 7, 8, etc. Building a different model for each day is not feasible, so only the above three models are used. To choose which day model to choose for intermediate days (2–5), (6–9) and (>10) the data from different days is scored using the three models. The macro AUC metrics is shown in Table 12. It is evident from Table 12 that day 1 model performs best on day 1 data, day 5 model on day 5 data and day 10 model on day 10 data. However, the question here is — which model should be chosen? day 1 or day 5 model for day 3? The trend in the tables is as follows — for a particular model we can move ahead in terms of days but not backwards. The upper triangular table is worse than lower triangular table. Day 1 model can be used for day 5, 10 but not vice-versa. Day 10 model cannot be used on day 1. Therefore, for the intermediate days 2–5 we propose to use day 1 model, day 6–9 use day 5 model and day 10 models for days after 10. Using this approach, claims on all days can be scored by using just three models and getting minimum performance drop.

9. Conclusion

In this work, text and structured data is combined together to train multi-label models for four different claims classification problems. IBLM that is trained with insurance specific corpus with enhanced vocabulary is utilized. As claims data changes with time, distinct models for different days are built. It is observed that combined multi-input model, text and structured data combined, performs better than structured-only and text-only models for majority of the classification tasks. When compared with their binary counterpart, the multi-label models perform equally well and outperforms in case of fraud. Further, to ascertain the level of severity for bodily injury claims, a multi-class classification model is developed. A channeling approach is proposed in which the predictions from the models for different days are combined to generate a *claim score* using domain specific heuristics to weigh different labels and day information. Severity of injury information is also included when calculating the *claim score*. It is observed that a high recall for claims which have more than one positive label associated with them is obtained. Further, a web application is built which combines the overall approach of making predictions using combined models, calculating the *claim score* and routing the claims to domain experts based on volume thresholds.

CRediT authorship contribution statement

Anuj Dimri: Conceptualization, Methodology, System design. **Arindam Paul:** Data curation, Writing – original draft. **Deeptha Girish:** Visualization, Investigation. **Peng Lee:** Web interface design, Writing – review & editing. **Sardar Afra:** Writing – review & editing, Validation, Supervision. **Andrew Jakubowski:** Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, *13*(Feb), 281–305.

Chalkidis, I., & Kampas, D. (2019). Deep learning in law: early adaptation and legal word embeddings trained on large corpora. *Artificial Intelligence and Law*, *27*(2), 171–198.

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785–794).

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: PRe-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*.

Dimri, A., Yerramilli, S., Lee, P., Afra, S., & Jakubowski, A. (2019). Enhancing claims handling processes with insurance based language models. In *2019 18th IEEE international conference on machine learning and applications* (pp. 1750–1755). IEEE.

Effler, G. (2019). U.S. Auto claims satisfaction study. https://www.jdpower.com/business/press-releases/2019-us-auto-claims-satisfaction-study. (Accessed 26 June 2020).

Hande, A., Puranik, K., Priyadharshini, R., Thavareesan, S., & Chakravarthi, B. R. (2021). Evaluating pretrained transformer-based models for COVID-19 fake news detection. In *2021 5th International conference on computing methodologies and communication* (pp. 766–772). IEEE.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Howard, J., & Ruder, S. (2018). Universal language model fine-tuning for text classification. In *Proceedings of the 56th annual meeting of the association for computational linguistics (Volume 1: long papers)* (pp. 328–339).

Kolyshkina, I., & van Rooyen, M. (2006). Text mining for insurance claim cost prediction. In *Data mining* (pp. 192–202). Springer.

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., et al. (2020). BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, *36*(4), 1234–1240.

Lewis, A., Mahmoodi, E., Zhou, Y., Coffee, M., & Sizikova, E. (2021). Improving tuberculosis (TB) prediction using synthetically generated computed tomography (CT) images. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 3265–3273).

Melekhov, I., Tiulpin, A., Sattler, T., Pollefeys, M., Rahtu, E., & Kannala, J. (2019). Dgc-net: Dense geometric correspondence network. In *2019 IEEE winter conference on applications of computer vision* (pp. 1034–1042). IEEE.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (pp. 3111–3119).

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems* (pp. 8026–8037).

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global vectors for word representation. In *Empirical methods in natural language processing* (pp. 1532–1543). URL http://www.aclweb.org/anthology/D14-1162.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., et al. (2018). Deep contextualized word representations. In *Proceedings of NAACL-HLT* (pp. 2227–2237).

Popowich, F. (2005). Using text mining and natural language processing for health care claims processing. *ACM SIGKDD Explorations Newsletter*, *7*(1), 59–66.

Ramos, J., et al. (2003). Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning. Vol. 242* (pp. 133–142). USA: New Jersey.

Sabban, I. C., Lopez, O., & Mercuzot, Y. (2020). Automatic analysis of insurance reports through deep neural networks to identify severe claims.

Stevens, E., Antiga, L., & Viehmann, T. (2020). *Deep learning with pytorch*. Manning Publications (Chapter 7).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. In *Advances in neural information processing systems* (pp. 5998–6008).

Wang, Y., & Xu, W. (2018). Leveraging deep learning with LDA-based text analytics to detect automobile insurance fraud. *Decision Support Systems*, *105*, 87–95.

Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., et al. (2015). Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision* (pp. 19–27).