DETRA NOTE

DETRA NOTE 2023-4

CYBER INCIDENT REPORTS: EXTRAPOLATING SEVERITY USING NEURAL NETWORKS

Justin Kher, Olivier Lopez and Hugo Rapior

Detralytics

DISCLAIMER

The content of the Detra Notes for a pedagogical use only. Each business case is so specific that a careful analysis of the situation is needed before implementing a possible solution. Therefore, Detralytics does not accept any liability for any commercial use of the present document. Of course, the entire team remain available if the techniques presented in this Detra Note required your attention.

ABSTRACT

Due to its emerging nature, cyber risk is a field in which very few data exist when it comes to calibrating models for the severity of such events. This makes the task of actuaries particularly challenging regarding pricing or claim management.

In the present note, we present a general methodology to process text data using neural networks, and how it can be used to determine the severity of a cyber incident when this information is missing. This methodology is illustrated on a public benchmark database. It can be used either to augment database, by adding some claim incidents to historical databases, from reported incident whose cost is unknown. It can also be used to quickly evaluate the severity of a cyber claim just after its occurrence.

Keywords: Cyber risk ; Natural Language Processing ; Neural Networks.

Short title: Cyber incident reports.

1 Introduction

The development of cyber insurance is slowed down by the important uncertainties in the evaluation of the risk. Behind the picture drawn by AMRAE in its LUCY study (see Association pour le Management des Risques et l'Assurance de l'Entreprise (2022)), the French market is an illustration of poor loss ratios (86% in 2021, 167% in 2020) in a context where the perimeter of the policies tend to shrink: even if deductibles increase and insurance capacity diminishes, these measure fail to anticipate the evolution of the threat and to build a sufficiently profitable ecosystem of cyber insurance.

This difficulty to properly quantify the risk and the losses is reinforced by the lack of data. In such a new market, where, in addition, risk evolves fast, insurance databases are usually to poor to achieve a sufficient statistical precision that would be key to evaluate the risk. Finding external information is hard: very few database (public or non public) exist to calibrate actuarial models, see Eling and Loperfido (2017) or Farkas et al. (2021). On the other hand, the ideal amount of data that would be required to achieve a proper statistical precision should be quite large: cyber risk leads to very volatile losses, and this important diversity of situation can be understood only through a large sample size.

However, the increasing attention devoted to cyber risk and cyber incidents leads to numerous reports (by the press or other medias) of incidents (not all of them being covered by insurance), and this non-structured source of information may be precious. Provided that it is processed in a proper way, these elements may enrich statistical analysis that contribute to the quantification of the risk.

In the present note, we explain how we can extract information from textual description of cyber incidents. We illustrate the methodology on a well known benchmark database, which gathers incidents for which we know the final amount (or, at least, an indicator of the severity). We rely on natural language processing (NLP), training a neural network architecture on this database. The model is then used to estimate the severity of some cyber events for which we only know the description.

The paper is organized as follows. In section 2, we describe the benchmark database that is used to calibrate the model. This database is publicly available, for reproducibility purpose. On the other hand, the methodology described in section 3 can be extended to any more precise insurance database. This procedure is based on text embedding and neural network architectures that allow to process textual data. The illustration on the database is done in section 4.

2 A database of cyber events

In this section, we briefly describe the database that we will use to illustrate the technique. The origin of the PRC database is explained in Section 2.1. Descriptive statistics and first empirical facts are shown in Section 2.2.

2.1 Description of the PRC database

The Privacy Rights Clearinghouse (see `https://privacyrights.org/`) is a US association whose goal is to educate the public to privacy issues. This association gathers, since 2005, a public database of data breaches which is widely used in the cyber risk literature, see for example Maillart and Sornette (2010). The reason of this popularity is mainly the presence of a marker of the severity of a data breach event, namely the "number of records" (that is the number of accounts that are exposed). Therefore, this database is considered as a benchmark in the academic literature related to cyber insurance, see for example Eling and Loperfido (2017), Farkas et al. (2021) or Edwards et al. (2016).

A link between this variable "number of records" and the true economic loss has been studied by Ponemon (2018), who propose, based on data from the Ponemon Institute, a model to link these two variables. See also Farkas et al. (2021) for an updated version. Let us mention that all these models seems very rough, and the unavailability of data related to their precise fit limits the capacity to measure the volatility of the formula proposed. We will therefore not compute this true economic loss in the following, focusing only on the variable "number of records" to determine if an event is severe or not. The methods we develop below, once applied to a true insurance portfolio, would not be affected by this imprecision, since true losses would then be available.

Among the important limitation of this database, let us mention that it is essentially US-focused: the scope of the association consists in only considering events that affect US citizen. Although US citizen may be affected by some events outside the US (due to the non-geographical component of cyber), the number of such events in the database is relatively small. Moreover, the database is fed by the investigations of the association. There is, for example, no automatic report from a given public authority. This can create a bias difficult to quantify. Finally, let us recall that data breaches are only a part of cyber risk, and that the conclusions may not be applied to any kind of cyber event.

The reason for considering this database in our case is essentially its availability to ensure reproducibility of the results. Again, our aim is mainly to describe and analyze a methodology that can be developed on more elaborate private database. From this perspective, the PRC database contains text reports on each incident. While the variable "number of records" is absent for a relatively large proportion of incidents (24 %), the reports are rich in information, and therefore represent a good illustration on how to use this textual information in the context of cyber.

2.2 Descriptive statistics

We decided to delete all the rows with 0 (or not filled in) for the variable “number of records”. These lines are not usable and represent 24% of the database. We considered that the information provided by these rows is not reliable and does not correctly represent the severity of the cyber event.

Claims are divided into two categories according to their number of records. The 40% most serious claims relative to the number of records variable are considered as serious. The rest of the claims are considered less significant.

The remaining database can be summarized as follows :

Count	6822
Mean	1522632
Std	41960690
min	1
25%	613
50%	2000
75%	10000
Max	3000000000

Table 1: Description of cyber claims

We seek to predict information related to the variable number of records. It is therefore necessary to observe the distribution of this variable by quartiles. The objective is to determine a threshold above which a claim description represents an important claim.

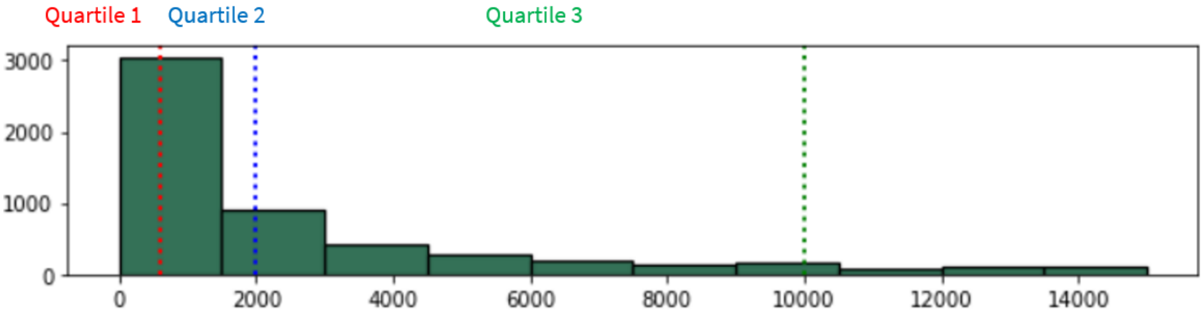

Figure 1: df records

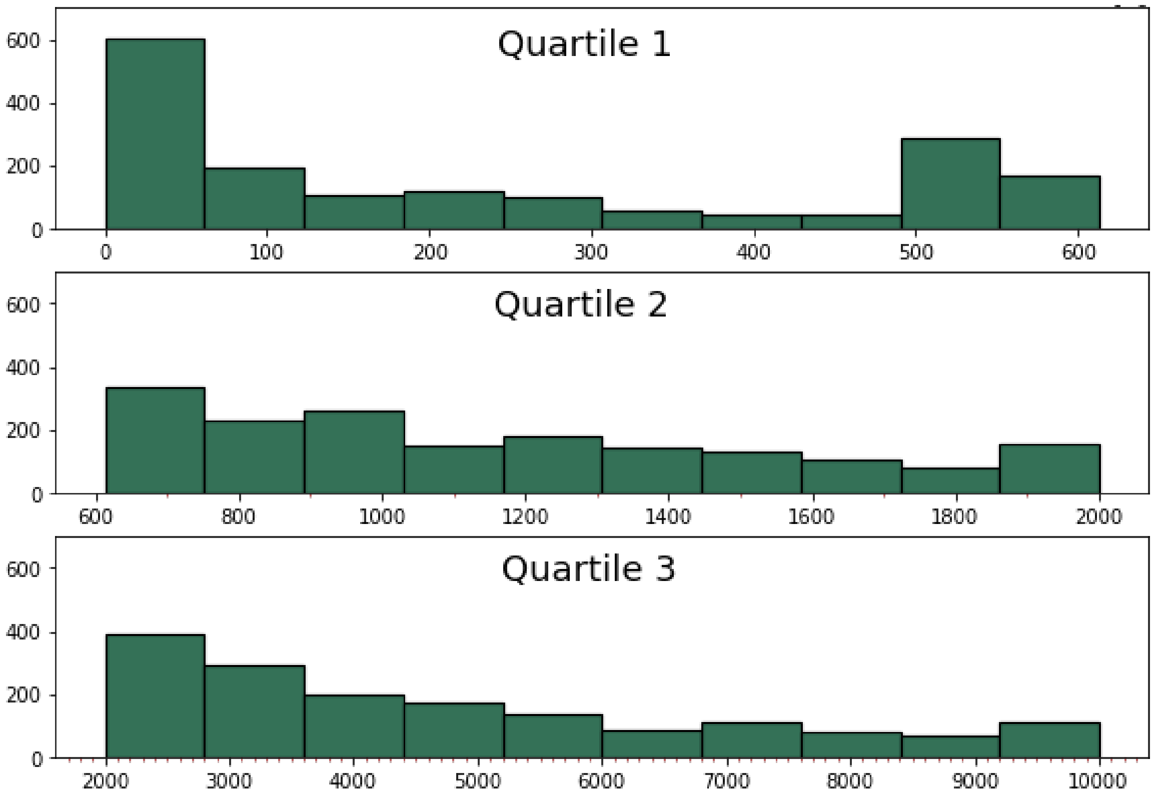

Figure 2: Number of records : Zoom quartiles

The data shows that claim descriptions associated with a high number of records are less frequent. The distribution of records indicates that few claims with a higher number of records are described. Thus, claim descriptions become rarer as the number of records associated with the description increases.

2.2.1 Other variables

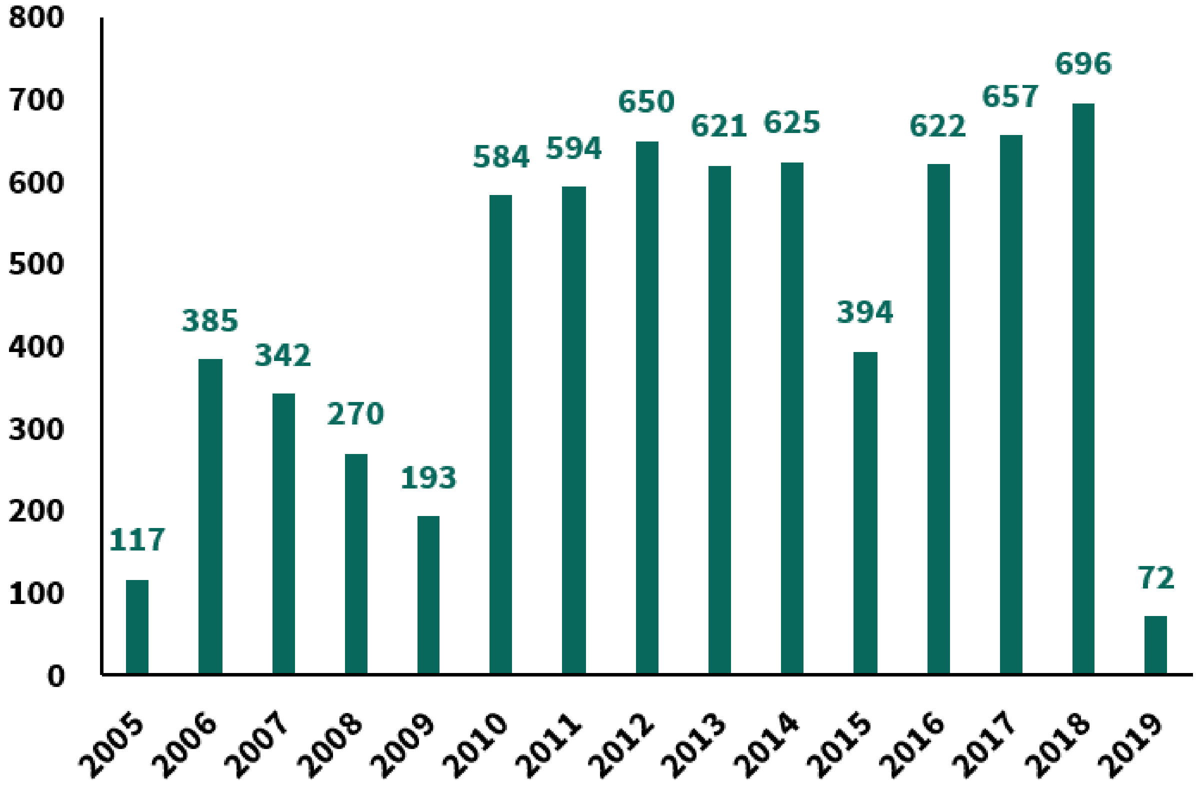

Figure 3: Number of claims per year

The bulk of the claims occurred between 2010 and 2018, with an unusually low number of claims in 2015.

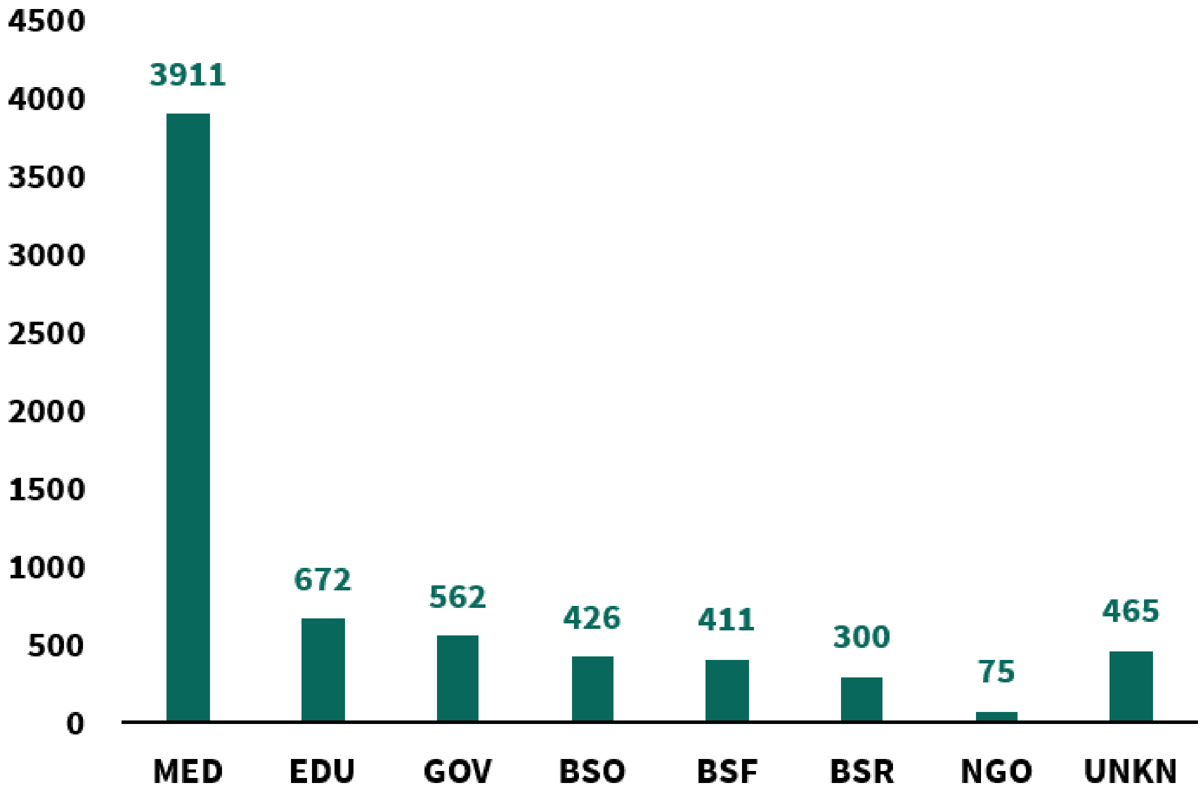

Figure 4: Number of claims per type of organization

The data shows a predominance of recorded claims in medical organizations. We expect the medical lexical field to dominate the frequency of words in the claims descriptions. We expect words such as SSN (Social Security Number), patients, etc. to occur frequently in the descriptions.

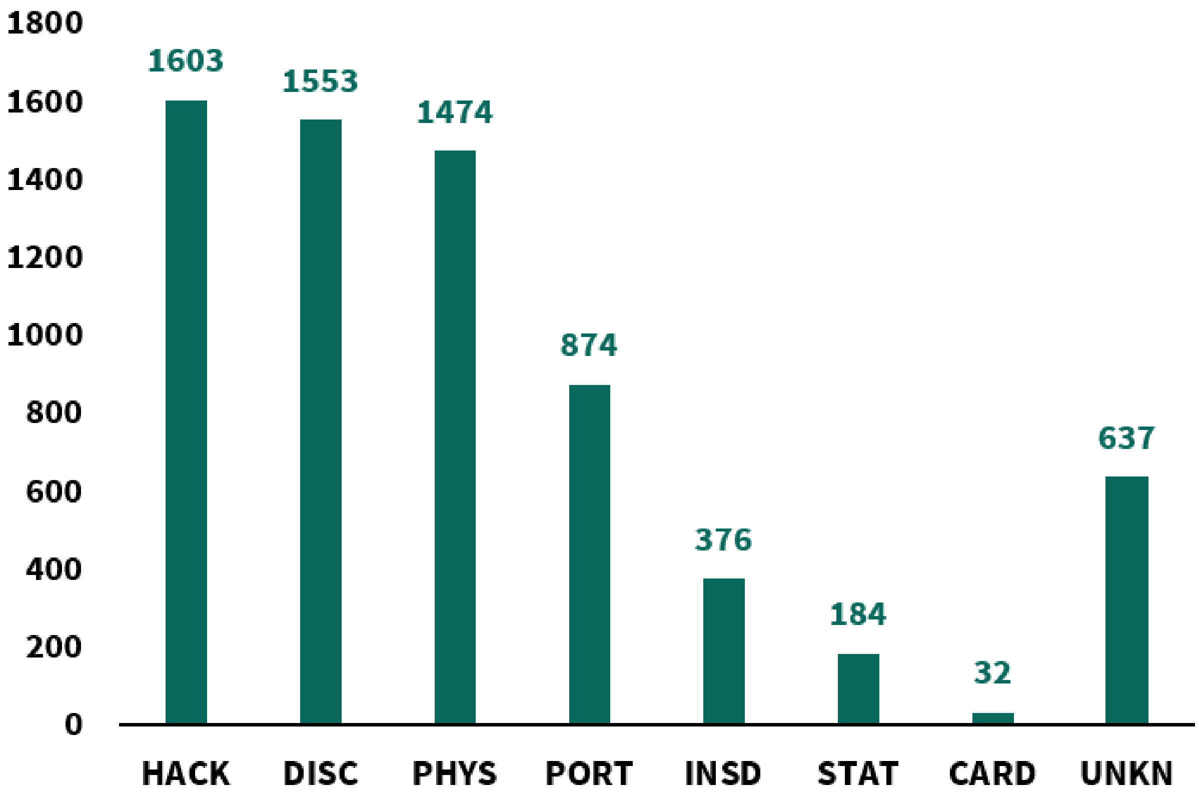

Figure 5: Number of claims per type of breach

Four categories of claims predominate. They correspond to the hack, the unintended disclosure, the physical breach (paper documents that are lost, discarded or stolen) and portable devices (lost, discarded or stole).

The frequency of each word is analyzed within the two categories of claims (severe and attritional). The aim is to identify words whose frequency is particularly high in the descriptions of the most severe claims. In the claim descriptions, it appears that some words are particularly linked to severe claims.

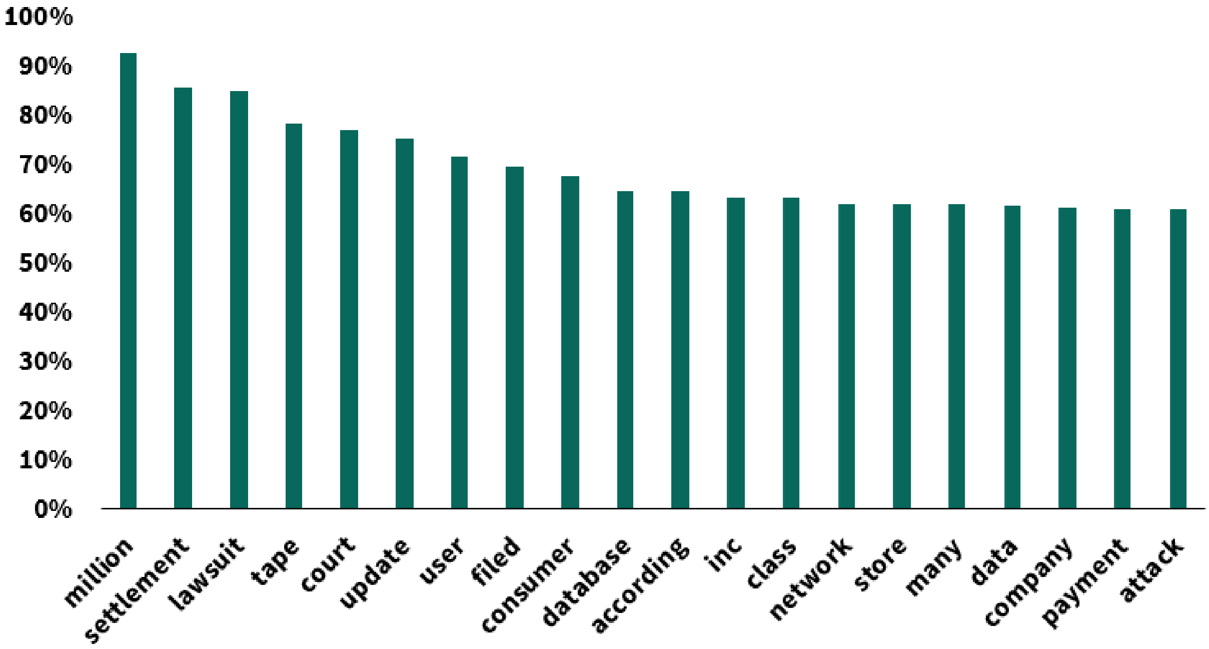

Figure 6: Most significant words in the description of major claims

Similarly we find that words like "paper", "document", "dishonest", "accidentally", "school", "employee" are related with less significant losses.

3 Neural networks and text embedding

In this section, we give a brief description and overview of the machine learning techniques that we use to process text. Section 3.1 provides a general introduction to neural networks through the description of the classical multilayer perceptron. The question of representing text through numerical vectors is addressed in section 3.2. Finally, in section 3.3, we introduce more modern methods that can be developed to study text, especially in the context of wider text corpus than the one that we consider in the illustration.

3.1 Neural networks framework

Neural networks are machine learning models characterized by a large number of parameters (allowing a low approximation error in estimating a function, due to an important flexibility in the shape of the output), and efficient algorithmic procedure to optimize the value of these parameters. These nice features make them competitive models to be used in insurance: for example Denuit et al. (2021) used these type of techniques for pricing ;

application to reserving and claim management has also been performed by Sabban et al. (2022) or Wüthrich (2018), who used deep neural networks to perform text analysis of claim reports ; neural networks have also been used in the study of mortality and longevity risk, see for example Hainaut (2018).

Neural networks are made of a combination and aggregation of neurons, or units. A neuron consists in performing a simple operation on inputs (i.e. characteristics) $\mathbf{X} = (X_j)_{1 \leq j \leq d}$, where $X_j \in \mathbb{R}$. It is characterized by:

- parameters $w \in \mathbb{R}^d$ called "weights" that allow to perform a linear combination of the inputs, $w^T \mathbf{X}$, where ^{T} denotes the transpose operator;
- an additional parameter b, traditionally called "bias", to shift this linear transformation;
- a (usually nonlinear) transformation, the "activation function" f.

The output of the neuron, is then $f(w^T \mathbf{X} + b)$. Typically, a Generalized Linear Model with fitted parameters w and b can be seen as a particular kind of neuron. Neural networks then combine the information contained in the different neurons. It is made of several layers, that is a collection of distinct neurons. The neurons of a given layer take the same information as input. The outputs of the neurons of the $i-$th layer are the inputs of the $(i+1)-$th layer. In terms of terminology, the last layer, which contains the final output of the network, is the output layer. All the previous layers are called "hidden layers". The whole set of parameters will be denoted by θ, and the corresponding output of the neural network by $\mathbf{x} \rightarrow g(\theta, \mathbf{x})$. This is summarized in Figure 7.

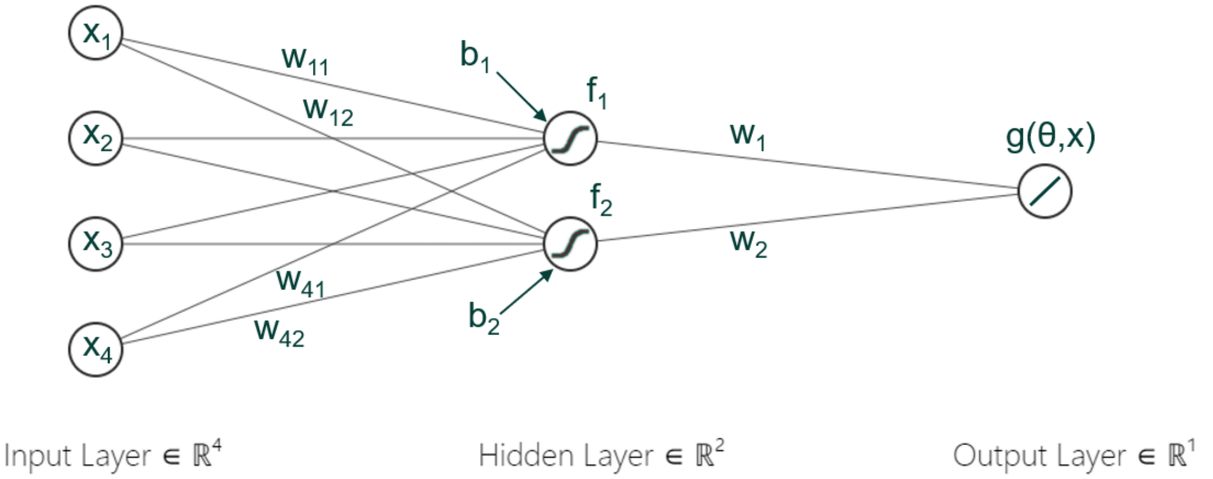

Figure 7: Neural network structure

The question is of course to optimize the value of the parameters ($n_i + 1$ parameters for each neurons of the layer $(i + 1)$, with n_i being the number of parameters of layer i and $n_0 = d$) in view of a given task. Typically, these parameters are fitted from a dataset $(Y_i, \mathbf{X}_i)_{1\leq i\leq n}$ of i.i.d. observations, where Y_i is a response variable that one wishes to predict from information $\mathbf{X}_i$. In our application, $Y_i \in \{0, 1\}$ corresponds to an indicator of severity of a cyber event, and $\mathbf{X}_i$ corresponds to the text description of the incident (how a vector in $\mathbb{R}^d$ is obtained to describe the reports is explained in section 3.2 below). The parameters of the network are chosen in order to minimize a loss function on the sample, namely

$$L_n(\theta) = \sum_{i=1}^{n} l(Y_i, g(\theta, \mathbf{X}_i)).$$

In the case of a binary response, the natural function l is the cross-entropy, where $-l$ is the Bernoulli log-likelihood.

A nice feature of neural networks is that this optimization can be performed relatively fast due to the combination of two techniques:

- stochastic gradient descent, see for example , which is a way to accelerate each step of a gradient descent algorithm;

- backward propagation see , that is a particularly efficient way to recursively compute the partial derivatives required to compute the gradient. This fast backward propagation algorithm is a consequence of the very structure of neural networks.

This generic presentation of neural networks corresponds to the case of the multilayer perceptron. This first generation of neural networks is relatively simple to develop, although modern variations like the one used in deep learning may provide more elaborate outputs, taking into account the structure of the data. See section 3.3 for more details. In this illustration on how to retrieve information on claims from text, we will mostly focus on this simple vision.

3.2 Word embedding methodology

To be processed as inputs of neural networks (or any other predictor), the text reports that we consider should be converted in vectors in $\mathbb{R}^d$. The most simple way to convert a word into numbers is to rely on one-hot encoding. The idea is to build a dictionary from all the words contained into our text corpus. If the total number of words is N, then a word is represented by a vector in $\mathbb{R}^N$, whose components are all 0, except one - corresponding to coordinate equal to the rank of the word in the dictionary - which is 1.

Usually, N is very large, and one can guess that the sparse vector used to describe a given word is not an optimal representation. An obvious problem is that one-hot encoding does not allow to accurately define a proximity between words. The idea behind text embedding is to project one-hot encoded vectors in a space of smaller dimension k where two words with close meaning should be represented by two vectors that are close according to a metric on $\mathbb{R}^k$.

Word2vec is a method that uses an artificial neural network to embed words into vector spaces, so similarity between words can be measured by the cosine of the angle between the two word vectors. Word2vec is a type of shallow two-layer neural network designed to generate word embeddings, which are vector representations of words that capture contextual information. This method is popular because it is able to accurately represent relationships between words in large corpuses of text. These representations are learned in an unsupervised way from large amounts of text and can be used in natural language processing applications, such as classification and sentiment analysis.

3.3 Extensions

In the present paper, we simply feed a classical multilayer perceptron with embedded vectors, to focus on the most important ideas behind the method. In practice, more elaborate strategies can be use to improve this first analysis, and we do not present here the results of these approaches. Nevertheless, we here briefly describe these techniques, and references that the reader may access to if she/he is interested in going beyond. Two now classical way to proceed are to use Convolutional Neural Networks (CNN) as in Jaderberg et al. (2016), or Recurrent Neural Networks like LSTM (Long-Short Term Memory, see Graves and Graves (2012)) as in Bai (2018), see also Luan and Lin (2019). A more detailed review of these techniques and how they can be used in the insurance context can be found in Sabban et al. (2022).

To briefly summarize the idea behind CNN, this kind of deep networks (that is, with a very large number of layers) has been initially introduced for image processing. Increasing the number of layers in a multilayer perceptron rapidly reachs its limits, and does usually not improve the procedure. On the other hand, CNN are no fully connected network, which means that a single neuron only receives information from a small part of the inputs. Moreover, each neuron is associated to a localization, receiving information only from inputs in a window centered at a particular point of the image, or, in our case, of the text.

Using CNN for text is a way to force the network to take into account the position of a word in a sentence, hence a way to give the ability to the network to better associate these words into a meaning. LSTM add the concept of memory cell. An example of such a structure is summarized in Figure 8. The network first considers the beginning of a text (first words), and produces a first output, which can be understood as a first prediction of the response variable Y. The next set of words in the text in then sent to the same network, but with additional input the first prediction Y. The specificity of LSTM, compared to other classes of recurrent networks, is to also produce, at each step, a second "output" which is called the memory cell. The content of this memory cell is also sent as one of the inputs of the next step (in addition to the current prediction of the input and to the current words under study). The presence of this feature allows to keep track on information present at the beginning of the text. A more detailed description of the way these networks have the ability to forget or keep in memory parts of the text that are very distant from each other is described in more details in Graves and Graves (2012). See also section 4.2 in Sabban et al. (2022).

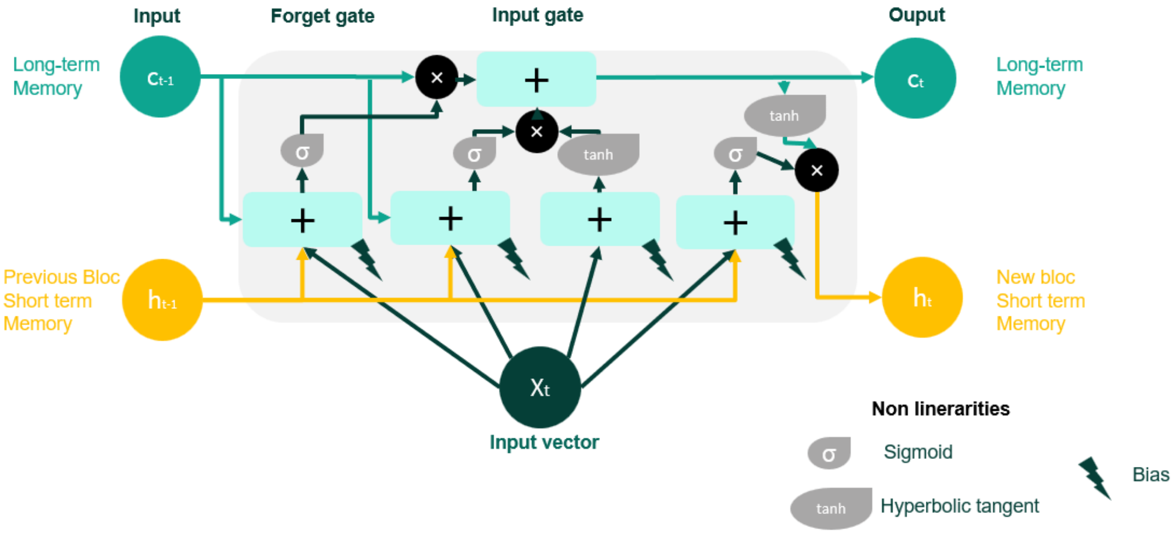

Figure 8: LSTM

4 Application to the PRC database

In this section, we propose an illustration of these text analysis methods to the PRC Database. We consider here the problem of binary classification, affecting a label 1 to the incidents above the 40% upper quantile of the variable "Number of records". In the simple procedure that we develop, the embedded text of each incident report is sent to a standard neural network model (a multilayer perceptron). Fitting the hyperparameters of this step is described in Section 4.1. The quality of the prediction and how to measure it is discussed in Section 4.2. This first automated step of the analysis is improved via the use of regular expression, described in Section 4.3. We then apply the calibrated models to the events in the PRC database for which the severity is missing in Section 4.4.

4.1 Neural architecture and hyperparameters

4.1.1 Approach and tools

First of all, the text was cleaned by removing patterns resembling dates and decimal numbers that could be confusing after punctuation processing. Stop-words that do not provide relevant information were also removed.

The approach uses various tools from different libraries for text data analysis. We start by cleaning and preprocessing the text with the NLTK library, which contains a punctuation and stopwords detector. Before creating bigrams and trigrams detectors with Gensim, we use a tokenizer to separate a string or sentence into individual tokens for easier parsing. Subsequently, we generate bigrams and trigrams detectors with the Gensim library. After that, we employ the Gensim's Word2Vec model for embedding and padding. Lastly, we apply the Keras library to create a neural network perceptron.

4.1.2 Challenges encountered during the analysis

In an initial approach, analysis of predictions on a test sample revealed identical loss probabilities for many claims. These identical predictions are related to recurring descriptions in the database.

The following description is given as an example to describe the problem encountered

"Location of breached information: Hacking/IT
Incident Business associate present: No"

> Note that a link to a website is always present and leads to a more detailed description of the incident.

This comment is repeated 166 times within the descriptions of different incidents. The difficulty of the model to correctly predict these comments is thus directly linked to the too imprecise description of the incidents in the database.

To remedy this difficulty and refine the prediction, it is possible to treat these cases in a brutal way by returning 1 or 0 according to the modality of the other characteristic variables. This method allows to treat separately these sensitive cases and to limit the prediction errors. Nevertheless, after the analysis of these claims, no variable appeared to be relevant for the prediction of the size of the claim. We therefore chose to exclude the rows corresponding to these claims.

4.2 Quality of the prediction

To categorize (as severe or attritional) a cyber event from its description, we use a multilayer perceptron as a neural network. Our dataset is divided into a training sample and a test sample. In this section we present the results on the test sample.

4.2.1 Approach: How to fit parameters ?

A gridsearch on the number of hidden layers and the number of neurons in each layer in the neural network allowed us to compare the performance of the different models. The model selected as optimal is the one whose parameters maximize the F1 score.

The F1 score is a performance indicator that combines both precision and recall. Precision measures the accuracy of the predicted results compared to the actual results. Recall measures the ability of the model to find all positive samples. The F1 score is a harmonic average of the two, weighted by their precision and recall. The higher the F1 score, the better the performance of the model.

$$\begin{aligned} F1\ score &= \frac{2 \times (Precision \times Recall)}{Precision + Recall} \\ &= \frac{2\left(\frac{TP}{TP+FP}\right)\left(\frac{TP}{TP+FN}\right)}{\frac{TP}{TP+FP} + \frac{TP}{TP+FN}} \\ &= \frac{TP}{MEAN(FP;FN)} \end{aligned}$$

In the confusion matrix:

- TP corresponds to true positives
- FP corresponds to false positives
- FN corresponds to false negatives

4.2.2 Results

The grid search gives the following optimal combination:

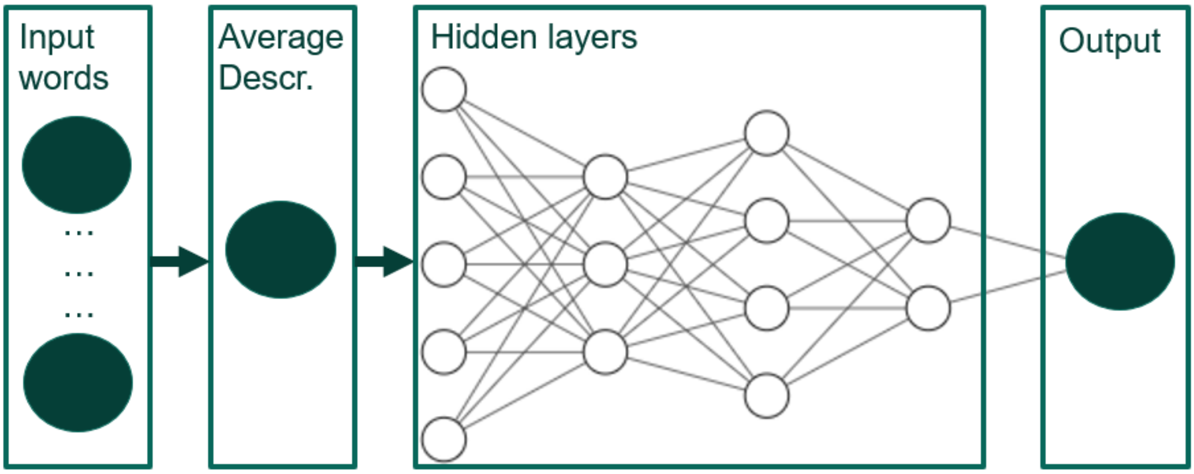

Figure 9: Optimal combination based on the grid search

The combination maximizing the F1 score is shown in the figure above. The associated confusion matrix is given below in Table 2.

		Severity?	
		0	1
Pred	0	479	176
	1	182	265

F1 score = 60 %

Table 2: Confusion matrix for the optimal combination according to the F1 Score

Figures 10 and 11 show the predicted probabilities on the test sample of a loss being serious according to their rank in terms of number of records. If the probability is less than 0.5, the prediction suggests an attritional claim, while a severe loss is predicted in the opposite case.

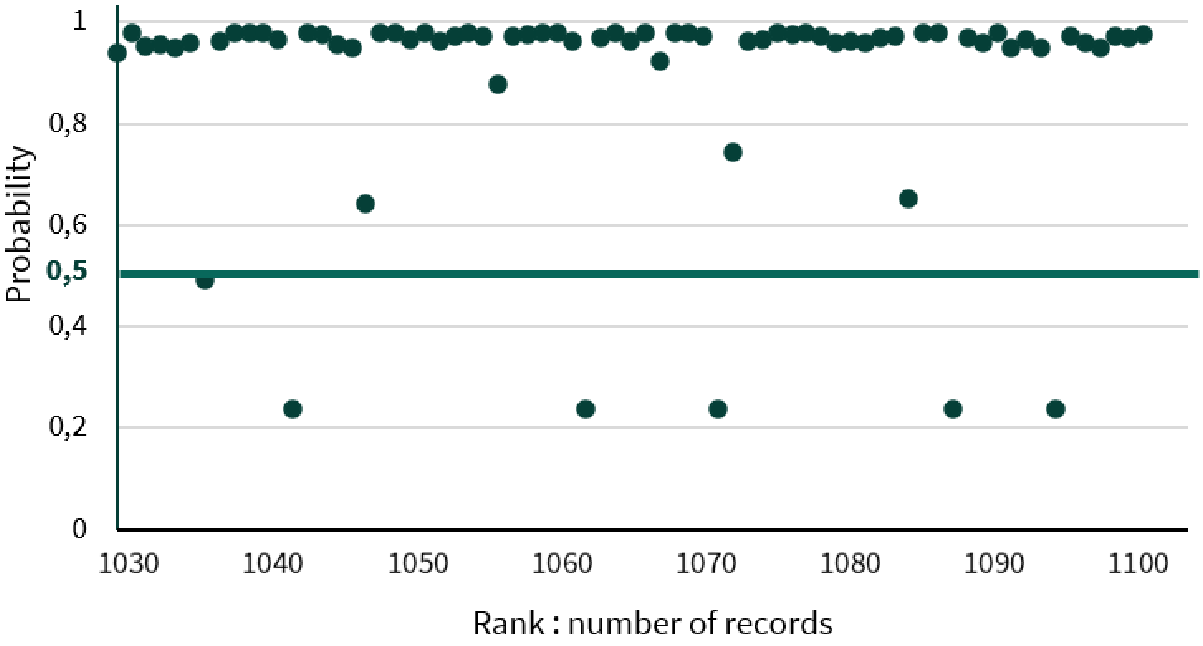

Figure 10: Predictability for claims over 300.000 records

Figure 10 depicts the largest claims (all above 300K records). It should be noted that the model constructed is particularly effective for truly exceptional claims (with a number of records exceeding 300.000).

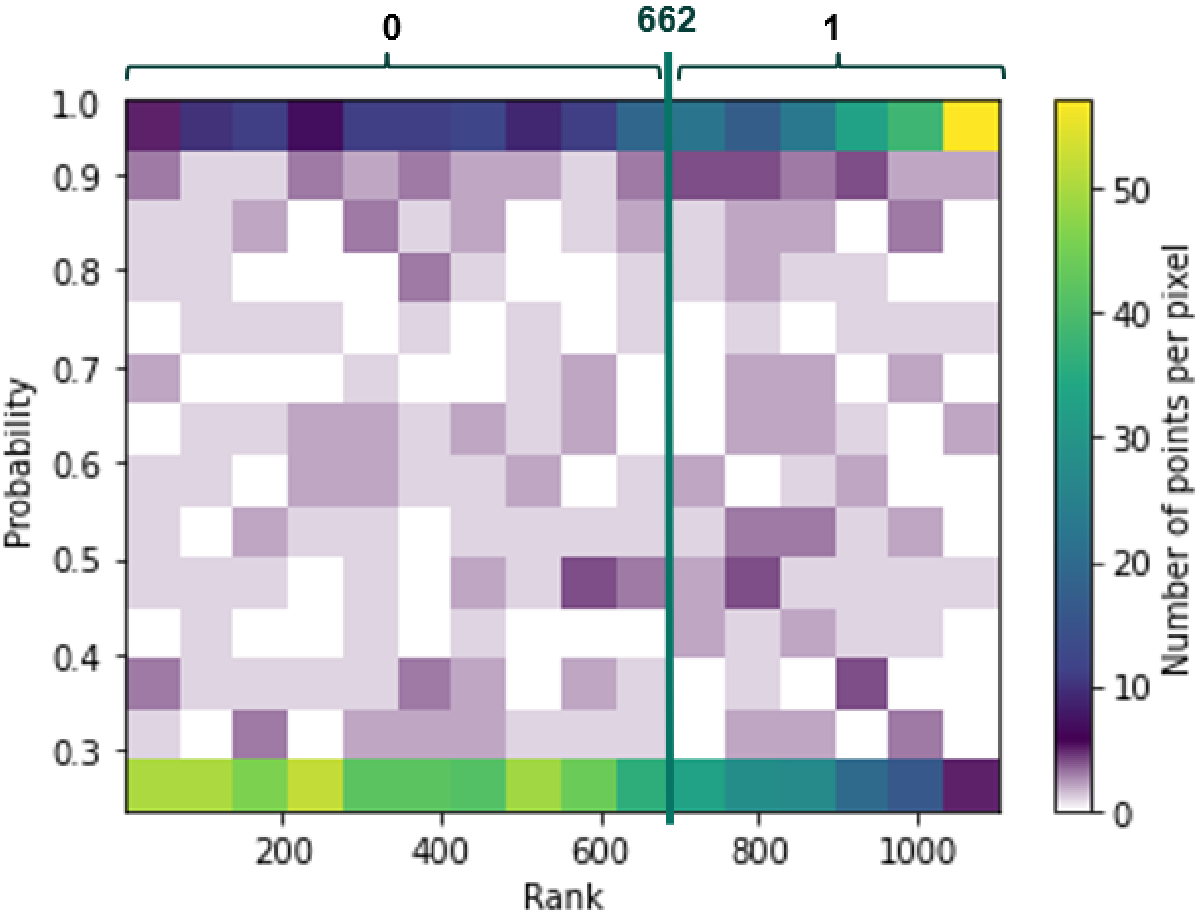

Figure 11: Predictions : threshold 4748 (40% of the most serious claims)

On Figure 11, the line drawn on the graph at the 662^{nd} observation separates attritional claims (on the left) from major claims (on the right) according to the number of records. This rank corresponds to a claim with 4748 records.

The probabilities are largely concentrated at the extremes (very close to 1 or the minimum value). The prediction is naturally more uncertain towards the borderline between attritional and serious claims. Note that the minimum probability is 0.23, this is related to the structure of the neural network. As we can see, the model is also quite efficient for the lowest ranks (very few records) and for the highest ranks (extremely high number of records).

For the 1102 observations in the test sample, the graphs below provide the severe claims rates in the test sample and the accuracy by modality. Figures 12 and 13 are predictions of severe claims, while Figures 14 and 15 are predictions of attritional claims.

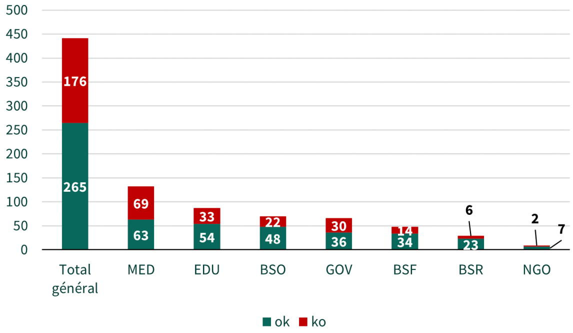

Figure 12: Predictions of serious claim by type of organization

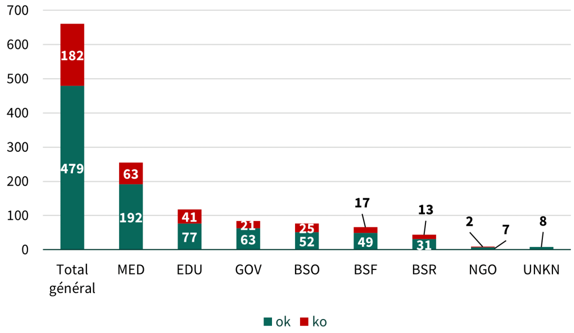

Figure 13: Predictions of attritional claim by type of organization

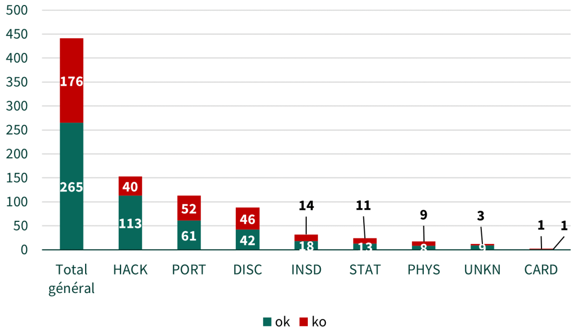

Figure 14: Predictions of serious claim by type of breach

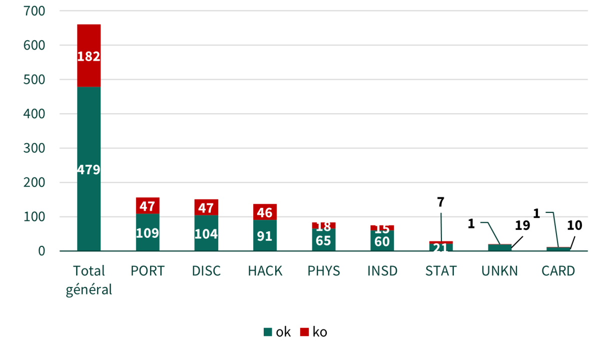

Figure 15: Predictions of attritional claim by type of breach

Regarding the type of organization, EDU (Educational Institutions), BSO (Businesses - Other), BSF (Businesses - Financial and Insurance Services) and BSR (Businesses - Retail/Merchant - Including Online Retail) are rather well predicted (more than 65% of severe losses correctly identified) while the prediction of MED severe losses is less efficient. The impact of medical claims therefore seems more complex to predict. Indeed, we can assume that the vocabulary of the descriptions is more specific and recurrent and does not allow to identify the impact of the claim. The importance of the loss is probably more reflected in the size of the breach than in the words used to describe it.

Regarding the type of breach, it appears that the predictions are rather good for hacks and portable device (lost, discarded or stolen laptop, PDA, smartphone, memory stick, CDs, hard drive, data tape, etc.).

4.3 Regular expressions

4.3.1 Introduction : improve the predictability of severe claims

The use of regular expressions to improve the binary classification performance of natural language text is a technique that can be very useful for neural network-based prediction systems. This section will discuss in detail how this method can be used to reduce the risk of false negatives and improve prediction accuracy in the context of cyber loss identification. The idea is to complete the output of the multilayer perceptron with a regex analysis. The objective is to determine the false negatives from the neural network.

Regular expressions (or Regex) can be used to great effect in various Python projects where data extraction and string manipulation is needed. It is a powerful tool when working with text, allowing for very efficient and precise search analysis. Additionally, regex can be used to check for specific formats, such as URLs, email addresses, and other strings, helping to identify and validate cyber claims within text. The "re" library is a package of functions and objects which are necessary for regular expressions to work in Python.

4.3.2 The limits of deep learning

It is difficult for a word embedding method (such as Word2Vec) and a neural network based classification algorithm to understand “number × word” numerical type expressions and their importance in the severity of a claim, because the neural network cannot intuitively determine the importance of the word and number combinations present in the sentence. Hence, the relationships between words and numbers are complicated to consider, making it difficult to identify and interpret the pattern. Moreover, the neural network cannot learn to recognize expressions without being exposed to a line already containing these combinations.

4.3.3 Analysis

In the claim descriptions of the test sample, we are interested in regular expressions of the form numerical number × word (e.g. 12 people, 315 patients, 1450000 million etc.).

We assume that the sum of the numerical values on these expressions for a selected vocabulary is an indicator of the severity of the claim description. Considering the description of a claim, the problem can be formulated as follows :

Given a set of words V and a set of pairs (N_u, M_u) for a description, where N_u and M_u are integers and words respectively.

The aim is to calculate the sum of the N_u for all words M_u belonging to the set V which represents the relevant vocabulary in the prediction of the most severe claims.

A severe claim is considered to have occurred if

$$\sum_{u=1}^{n} N_u \cdot 1_{M_u \in V} > Threshold_{REGEX}$$

With $Threshold_{REGEX}$ the chosen threshold for the regular expression (which can be different than the one initially chosen to separate attritionnal from large claims).

The following table illustrates the procedure:

Number of records	Description of incident	Description cleaned
27000	A hacker attack at payroll processing firm Ceridian Corp. of Bloomington has potentially revealed the names, Social Security numbers, and, in some cases, the birth dates and bank accounts of 27,000 employees working at 1,900 companies nationwide. In a Jan. 29 letter to an affected worker obtained by the Star Tribune, Ceridian said a hacker attacked its Internet payroll system Dec. 22 and 23.	hacker attack payroll processing firm ceridian corp bloomington potentially revealed name social security number case birth date bank account **27000** employee working **1900** company nationwide jan **29** letter affected worker obtained star tribune ceridian said hacker attacked internet payroll system dec 22 23
27000 + 1900 + 29 = 28929 >Threshold		

4.3.4 Preliminary steps

Claim descriptions are an important source of information for insurers, but the large amount of numeric data they contain can sometimes be confusing. To improve the analysis of claim descriptions, cleaning is required to remove irrelevant data such as dates, years between 2000 and 2017, and generally any numeric data not related to the number of records. This cleaning reduces the noise in the database and makes it easier to analyze the numeric values in the claim descriptions.

Description of breach	UPDATE (4/5/2011): The nurse who challenged her termination
Desc. clean without date treatment	update **452011** nurse challenged termination
Desc. clean with date treatment	update nurse challenged termination agreed resign rather fired

Table 3: Date processing : an example.

Moreover, it should be noted that after analysis of the numerical value in the *“number* × *word”* pattern, it was decided to remove the decimal numbers. Indeed, decimal numbers are only rarely used and are mainly associated with the word "million". (ex: 1.3 million dollars). The word “million” alone is sufficiently predictive within the neural network to predict a serious disaster.

4.3.5 Which vocabulary to use ?

The objective is to define a vocabulary for which it is relevant to sum the numerical data of each regex pattern.

To build the vocabulary, we consider all the patterns found on the set of descriptions and we look at the numerical value. It is now possible to measure the relevance of each word in the regex prediction of the descriptions. Concretely, it is the coherence between severe and attritional claims and the predictions given by the pattern with only this word (for both type of claims).

The table below indicates for each word the number of descriptions where the number in front of the word is consistent with the importance of the claim (severe or not). The regex threshold was set to the same value as the threshold determined at the beginning to separate attritional from severe claims.

Word	nb. Pred ko	nb. Pred ok	Total	Predictive power of the word ?
patient	16	55	71	77%
people	15	52	67	78%
million	43	5	48	10%
record	2	27	29	93%
student	4	24	28	86%
employee	2	22	24	92%
current	7	14	21	67%
individual	2	19	21	90%
year	12	7	19	37%
customer	3	15	18	83%
name	1	13	14	93%
social	2	12	14	86%
information	7	6	13	46%
month	10	3	13	23%
email	4	7	11	64%
california	-	10	10	100%
former	4	6	10	60%
new	-	9	9	100%
affected	3	5	8	63%
client	2	5	7	71%
credit	2	5	7	71%
hacker	5	2	7	29%
...	...	...	...	...

Table 4: Word predictive power

It appears that numbers associated with individuals (patient, people, student, employee, customer) are more predictive than words such as million, year, information, month or hacker whose quantity does not matter in the loss experience.

Thus, it is possible to isolate words whose preceding numerical values introduce an error on the severity of the prediction. In other words, this means that the number in front of the word is not correlated with the number of records in the claim.

Word	nb. Pred ko	nb. Pred ok	Total	Predictive power ?
1998	4	-	4	0%
2003	3	-	3	0%
dave	3	-	3	0%
appears	2	-	2	0%
breach	2	-	2	0%
desktop	2	-	2	0%
hundred	2	-	2	0%
incident	2	-	2	0%
massachusetts	2	-	2	0%
money	2	-	2	0%
separate	2	-	2	0%
system	2	-	2	0%
university	2	-	2	0%
update	2	-	2	0%
website	2	-	2	0%
wwwdatasettlementcom	2	-	2	0%
oct	1	-	1	0%
000	1	-	1	0%
12000	1	-	1	0%
18445782656more	1	-	1	0%
200	1	-	1	0%
23update	1	-	1	0%
...	...	...	...	...

Table 5: Non predictive vocabulary

4.3.6 Which threshold to choose for a severe claim by regex analysis ?

On the list of comments that include regular expressions with a “number × word” pattern: we look at the predictivity that emerges from these patterns only. We will therefore use, as the only prediction factor, the sum of the numerical values on the relevant vocabulary with a given threshold. It is then possible to build a confusion matrix, after which we then vary the REGEX threshold to find an optimum in terms of F1 score.

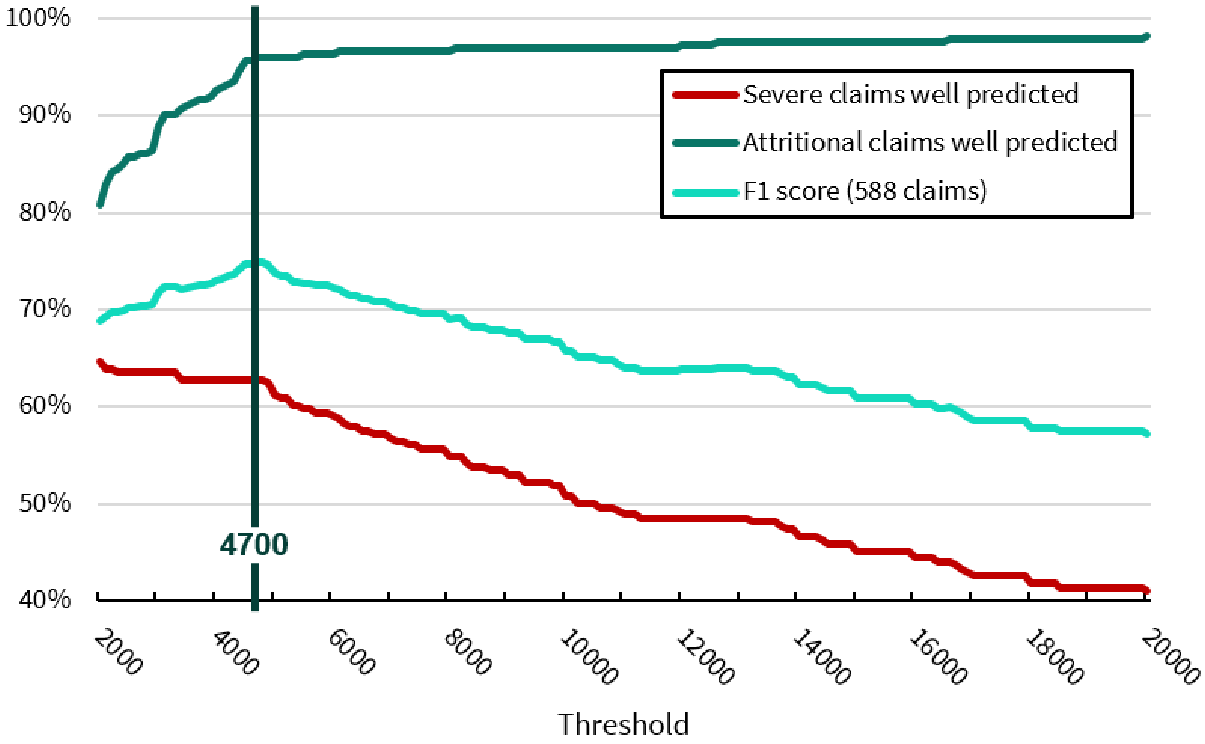

Figure 16: Regex threshold

The optimum that seems to emerge from Figure 16 is at the same value that has been chosen for the serious claim threshold, which is 4700.

To convince ourselves of the relevance of using regular expressions, we can now compare the number of records of a claim with the value calculated on a description with the method used (a sum of numerical values on a given vocabulary).

The lines of claims containing expressions corresponding to the selected pattern are represented on Figure 17. Depending on the number of records, these lines represent severe (green) or attritional (red) claims.

The sum of the numerical data in the description gives a value that is compared to a regex threshold (represented by a horizontal line on the graph) to classify the claim. Points above this line will be classified as severe losses.

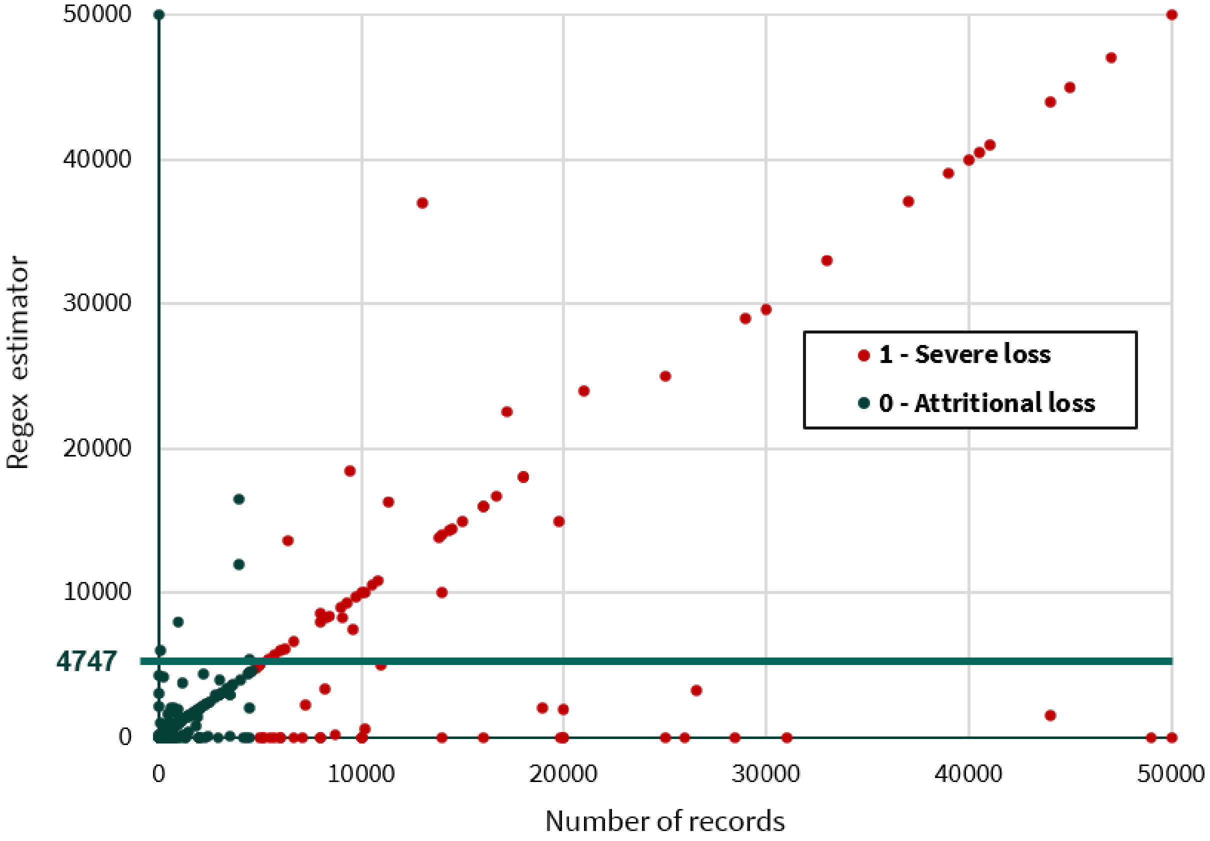

Figure 17: Identification of severe claims with regexes

The sum of the numerical data included in the description of a claim generally corresponds to the number of records. The linear curve in the graph illustrates this phenomenon. The six green points above the threshold line (4747 records) are false positives. For these lines, the sum of the numerical data does not predict the severity of the loss. Not surprisingly, the regex threshold is the same threshold that was used to distinguish severe from attritional claims.

4.3.7 Results

In this paragraph, we start again from the multilayer perceptron model presented previously.

As a reminder, the optimal results found with the grid search are given below.

		Severity?	
		0	1
Pred	0	479	176
	1	182	265

F1 score = 60 %

Table 6: Confusion matrix for the optimal combination according to the F1 Score

Among the 0 predicted by the perceptron, the table below summarizes the severe claims added by the regex analysis.

		Perceptron	2000	3200	4748	6000
Selected vocabulary	**False negatives**		42	42	40	35
	True negatives		21	8	6	6
	F1 Score	**60%**	**64%**	**65%**	**66%**	**65%**
All vocabulary	**False negatives**		46	45	44	38
	True negatives		39	26	22	21
	F1 Score	**60%**	**64%**	**65%**	**65%**	**64%**

Table 7: Improvement by analyzing regex expressions according to different thresholds

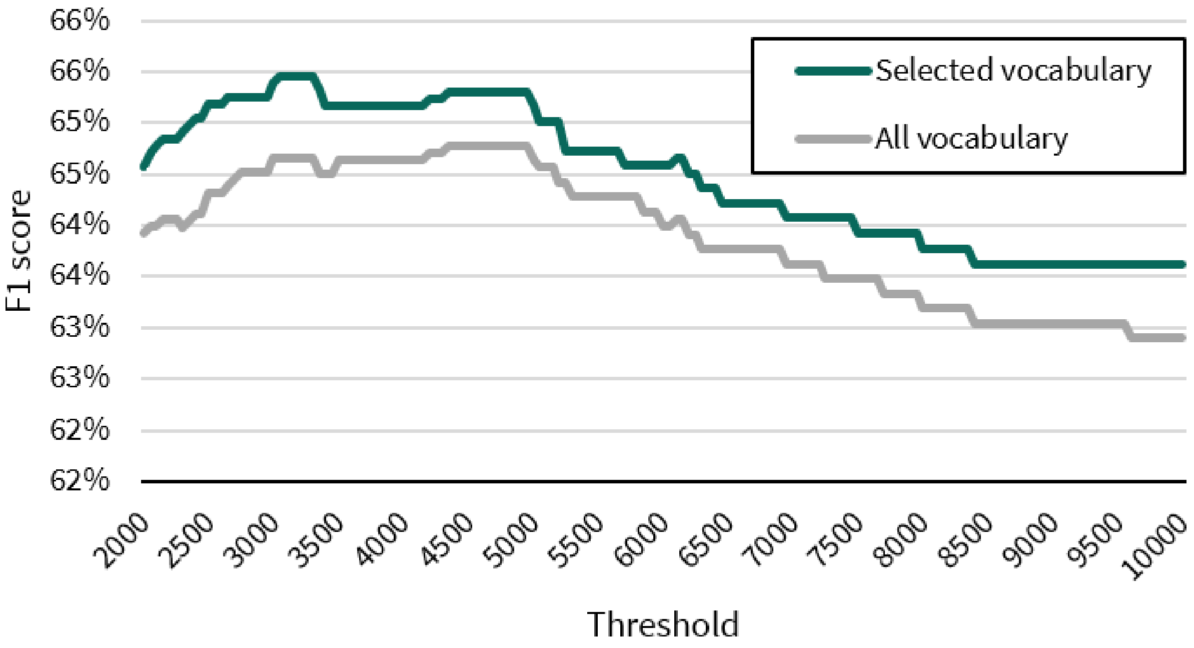

Figure 18: F1 score after the analysis of regular expressions according to the set threshold

Sensitivities were used to confirm the optimal threshold for the sum of the numerical values in the description of a claim. This threshold is actually the same as the one that initially allowed to separate the database into two categories according to the number of records for each claim. This therefore confirms that the sum of the numeric data preceding our selected words in a cyber loss description tends to the number of records. The analysis of the regex expressions thus allows us to significantly improve the detection of severe losses.

		Severity?	
		0	1
Pred	0	473	176
	1	148	305

F1 score = 66 %

Table 8: Confusion matrix for the optimal combination after regex analysis

The use of regular expressions is therefore very effective in classifying claims because the numerical information contained in a claim description corresponds rather well to the number of records. Nevertheless, this method is not sufficient to analyze and classify all the descriptions, as this numerical data is largely absent from the descriptions.

4.4 Events with missing data

In the original PRC dataset there are 2186 claims for which the number of records is zero or missing. These rows were initially removed because the number of records did not allow to classify the severity of the claim. We now seek to compare the claims in this incomplete database with the usable database. The descriptions of the claims in the incomplete database are analysed using word embedding techniques and the neural network constructed in section 4.

Figures 19 and 20 compare the number of claims in the two databases for each modality while figures 21 and 22 compare the severe claim rates.

Number of claims for each modality in the two data sets

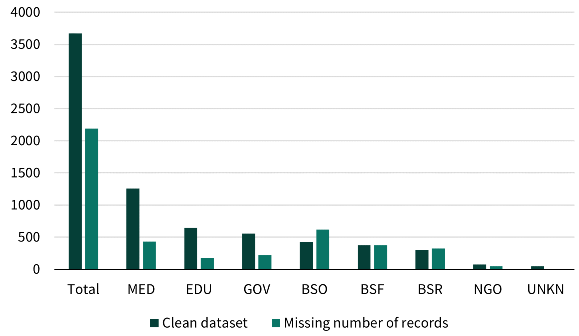

Figure 19: Per type of organization

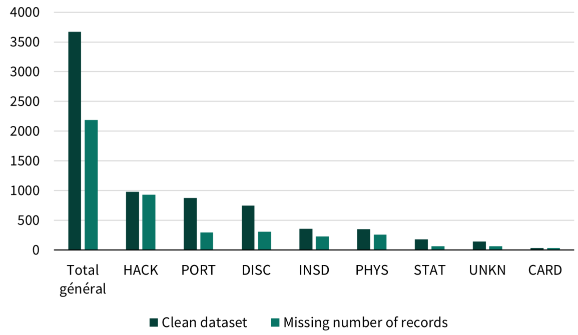

Figure 20: Per type of breach

Medical (MED) and governmental (GOV) institutions have an obligation to be transparent about the claims that have occurred. It is therefore not surprising that their proportion is lower in the data where the number of records is missing. Conversely, business related modalities (BSO, BSF, BSR) are supposed to be worse and less transparent in diagnosis, so it makes sense that these claims have a missing number of records. They don't know how much data they have lost or they don't want to report it. On the types of breaches, it appears that hacks and physical breaches are particularly important on this uncomplete database.

Proportion of severe claim rate of each modality for the chosen model

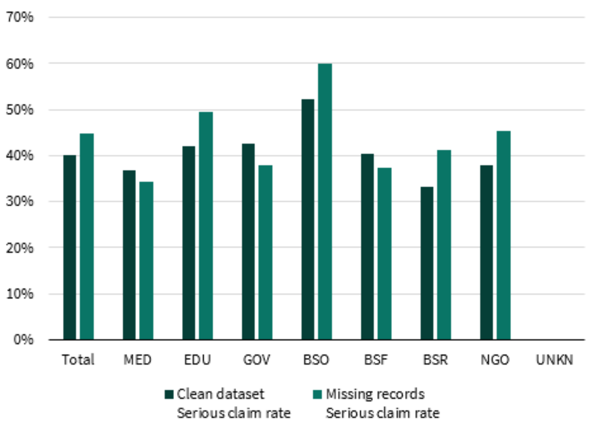

Figure 21: Per type of organization

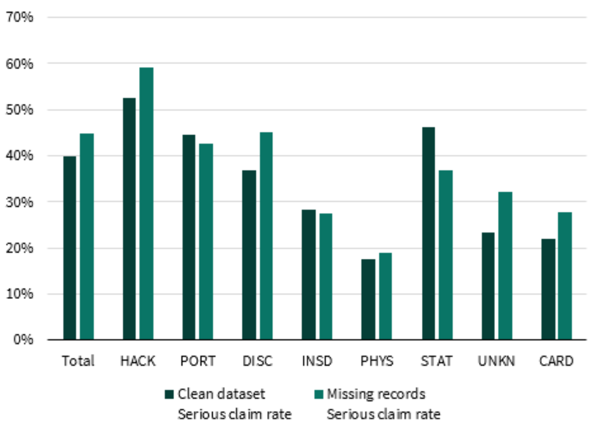

Figure 22: Per type of breach

The severe claim rate on MED, GOV and BSF[1] is lower on this database than on the database where the number of records is completed. We assume that for the sake of transparency, these organizations report small claims, for which they do not bother to quantify the impact.

On the other hand, educational institutions (EDU) and businesses that are not related to banking, insurance and financial services (BSO[2], BSR[3]) have a higher severe claim rate on this incomplete database. One hypothesis to interpret these observations is that these organizations are worse at diagnosis, which explains the over-representation of predicted severe claims in the database where the number of records is missing.

Looking at the types of breaches, we mainly observe that physical breaches (PORT [4], STAT [5]) are less severe in the incomplete database than in the exploitable database.

[1] Businesses (Financial Services, Banking, Insurance Services)

[2] Businesses (Manufacturing, Technology, Communications, Other)

[3] Businesses (Retail/Merchant including Grocery Stores, Online Retailers, Restaurants)

[4] Portable Device (lost, discarded or stolen laptop, PDA, smartphone, memory stick, CDs, hard drive, data tape, etc.)

[5] Stationary Computer Loss (lost, inappropriately accessed, discarded or stolen computer or server not designed for mobility)

5 Conclusion

In this note, we described a generic methodology to predict the severity of a cyber incident from text. The illustration is intentionally done on a benchmark database, in the particular case of data breaches. But it can be easily extended to more elaborate cyber claims and richer database. Our purpose is essentially to demonstrate that, with relatively standard machine learning techniques, a satisfying prediction quality can be obtained. Indeed, let us observe that all the predictions are obtained using relatively simple models (multilayer perceptron could be replaced by LSTM or Transformers as in , requiring more computer time to calibrate but which are known to be better adapted to text). Moreover, let us note that we directly used pre-trained embedding techniques, but adapting them to the particularities of the text corpus that we consider (as in) could also be a way to increase the performance. Apart from these technical improvements, let us note that we made the choice to use text data without taking into account any other type of information. This was to stress the ability of textual data to convey a usable information. On the other hand, additional variables (like the sector activity of the victim, its size and revenue, its budget spent on cybersecurity) can also be added to the prediction models.

Let us point that, in our opinion, text data are a really rich vector of information in analyzing a risk which is, for now, hard to precisely quantify. In the present note, we focus on predicting a severity indicator, but the methodology can easily be extended to analyzing a quantitative variable. This task is, of course, more challenging due to the volatility of the losses. In any case, the idea is either to extrapolate missing information (in view of consolidating a richer database), or to predict the evolution of a claim. Regarding this last point, let us point that claim management of cyber incidents is not the less challenging task: after the occurrence of a claim, its cost may be hard to anticipate: the stabilization of its amount could take time, since the consequences of cyber claim can span over a relatively long period. Many actions may be required to sweeten the cost, but such strategies require to sort out the incidents. The present methodology and its extensions ambition to play a critical role in achieving this goal.

References

Association pour le Management des Risques et l'Assurance de l'Entreprise (2022). Lumière sur la cyber-assurance. https://www.amrae.fr/bibliotheque-de-amrae?

combine=rapport%20lucy&ref_id=4022&ref_type=publication&items=4022&sort_by=created&sort_order=DESC&page=0.

Bai, X. (2018). Text classification based on lstm and attention. In *2018 Thirteenth International Conference on Digital Information Management (ICDIM)*, pages 29–32. IEEE.

Denuit, M., Charpentier, A., and Trufin, J. (2021). Autocalibration and tweedie-dominance for insurance pricing with machine learning. *Insurance: Mathematics and Economics*, 101:485–497.

Edwards, B., Hofmeyr, S., and Forrest, S. (2016). Hype and heavy tails: A closer look at data breaches. *Journal of Cybersecurity*, 2(1):3–14.

Eling, M. and Loperfido, N. (2017). Data breaches: Goodness of fit, pricing, and risk measurement. *Insurance: mathematics and economics*, 75:126–136.

Farkas, S., Lopez, O., and Thomas, M. (2021). Cyber claim analysis using generalized pareto regression trees with applications to insurance. *Insurance: Mathematics and Economics*, 98:92–105.

Graves, A. and Graves, A. (2012). Long short-term memory. *Supervised sequence labelling with recurrent neural networks*, pages 37–45.

Hainaut, D. (2018). A neural-network analyzer for mortality forecast. *ASTIN Bulletin: The Journal of the IAA*, 48(2):481–508.

Jaderberg, M., Simonyan, K., Vedaldi, A., and Zisserman, A. (2016). Reading text in the wild with convolutional neural networks. *International journal of computer vision*, 116:1–20.

Luan, Y. and Lin, S. (2019). Research on text classification based on cnn and lstm. In *2019 IEEE international conference on artificial intelligence and computer applications (ICAICA)*, pages 352–355. IEEE.

Maillart, T. and Sornette, D. (2010). Heavy-tailed distribution of cyber-risks. *The European Physical Journal B*, 75(3):357–364.

Ponemon, L. (2018). Cost of a data breach study: global overview. *Benchmark research sponsored by IBM Security independently conducted by Ponemon Institute LLC.*

Sabban, I. C., Lopez, O., and Mercuzot, Y. (2022). Automatic analysis of insurance reports through deep neural networks to identify severe claims. *Annals of Actuarial Science*, 16(1):42–67.

Wüthrich, M. V. (2018). Neural networks applied to chain–ladder reserving. *European Actuarial Journal*, 8:407–436.

Detralytics

People drive actuarial innovation

www.detralytics.com - info@detralytics.eu