A Framework for Mining Enterprise Risk and Risk Factors from Text Documents

Tirthankar Dasgupta
TCS Innovation Lab,
gupta.tirthankar@tcs.com

Lipika Dey
TCS Innovation Lab,
New Delhi
lipika.dey@tcs.com

Prasenjit Dey
IIT Kharagpur
prsnjt002@gmail.com

Rupsa Saha
TCS Innovation Lab,
Kolkata
rupsa.s@tcs.com

Abstract

Any real world events or trends that can affect the company's growth trajectory can be considered as *Risk*. There has been a growing need to automatically identify, extract and analyze risk related statements from news events. In this demonstration, we will present a risk analytics framework that processes enterprise project management reports in the form of textual data and news documents and classify them into valid and invalid risk categories. The framework also extracts information from the text pertaining to the different categories of risks like, their possible cause and impacts. Accordingly, we have used machine learning based techniques and studied different linguistic features like n-gram, POS, dependency, future timing, uncertainty factors in texts and their various combinations. A manual annotation study from management experts using risk descriptions collected for a specific organization was conducted to evaluate the framework. The evaluation showed promising results for automated risk analysis and identification.

1 Introduction

A real world event that has an associated probability of causing damage, injury, liability, loss or any other negative impact is termed as a risk(Lu et al., 2009; Slywotzky and Drzik, 2005; Beasley et al., 2005; Lu et al., 2009). Organizations are always on the look out for information related to such events caused by internal and external vulnerabilities such that the possible negative impacts may be avoided through preemptive action. Sources of risk can be many. The difficulty of risk identification arises from the diversity of the sources. Risks can arise from uncertainty in financial markets(Leidner and Schilder, 2010; Ykhlef and Algawiaz, 2014), industrial processes or due to project failures. Unexpected events like natural disasters, legal issues, deliberate attacks from adversaries or certain competitor moves can all lead to situations that can impact an organization and hence can be termed as risks.

Generally, a risk has the following characteristics: The risk type R_T or a name for the description of the risk that characterizes the nature of the adversarial potential, The *cause* R_C or the event that may cause the specified risk and the *impact* R_I that deals with the severity of the damage caused once it materialize.

Like all expert-driven activities that involve knowledge about handling uncertainties and predictive capabilities, risk analysis is a complex task that requires expertise that is acquired with experience. It is difficult to document. Besides, experts differ in their opinions. Sifting through a large number of such analyst reports and summarizing them is a tedious activity(Kogan et al., 2009). In this work, we present text mining techniques that can analyze large volumes of analyst reports to automatically extract risk statements, aggregate them and summarize them into risks of various categories.

As mentioned earlier, experts predict risks as probable future events that can impact business outcomes. The proposed methods employ machine learning based techniques to learn linguistic features

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations, pages 180–184, Osaka, Japan, December 11-17 2016.

and their dependencies from labeled samples of risk statements. The learned classifiers are applied to input text, wherein every sentence in the text is subjected to binary classification as ”risk” or ”not a risk”.

The salient contributions of this demonstration are as follows:

1. A machine learning and computational linguistics based framework to analyze textual News events and classify them into true risk and false alarm categories,

2. Extract different categories of risk factors like *causes* and their possible *impacts*.

2 Proposed Risk Classification Framework

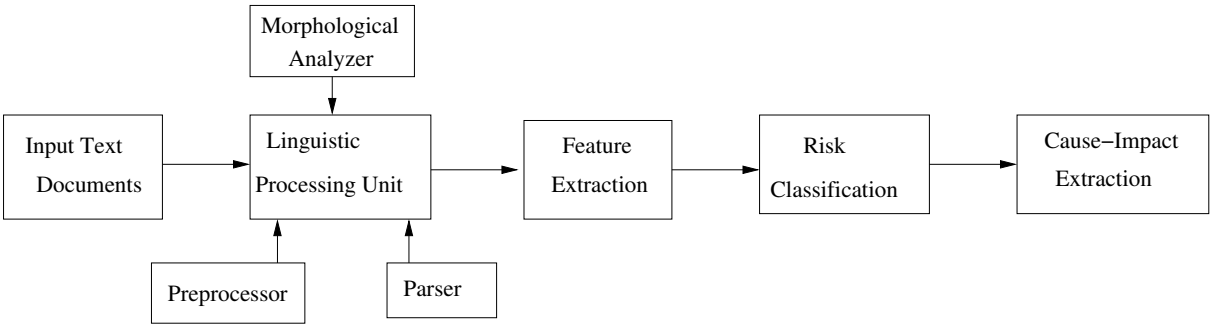

Figure 1: Architecture of the risk identification framework

The overall architecture of the risk classification and analysis framework is depicted in Figure 1. The proposed architecture has four primary modules: a) The *Linguistic pre-processing unit* b) *Feature extraction unit* c) *Risk classifier unit* and d) the *Risk analysis unit*. The input text is first passed to the preprocessing unit that removes html tags, and foreign language characters from the text. The preprocessed text is then passed to the Stanford parts-of-speech(POS) tagger and parser to label each word with their corresponding POS and to extract different dependency relations within the sentences. From the output of the POS tagger, root verbs are extracted and passed to an English morphological analyzer to identify the tense, aspect and modality of the root verb.

The syntactically analyzed text is then passed to the feature extraction unit. The features considered can be broadly classified into three types a) Future timing in texts, b) Uncertainty in texts and c) traditional linguistic features.

Future timing refers to the expressions that indicate (possible) upcoming events or states. For instance, the verb “expects” in the sentence *Testing of OCR division is expecting an overall fall in performance in the next few months*, indicate future timing.

Uncertainty mainly ”concerned with the speaker’s assumptions, or assessment of possibilities, and, in most cases, it indicates the speaker’s confidence or lack of confidence in the truth of the proposition expected” (Coates, 1987). Various levels of uncertainty can be inferred from the expression. As a preliminary study, we have used only the presence of epistemic modal expressions like, modal auxiliaries, epistemic lexical verbs, adverb, adjectives and nouns to determine uncertainty in a text (Coates, 1987).

In *traditional linguistic* features we have considered *N-gram counts (N)*, *POS features(POS)*, *Dependency features (D)* that includes dependency length, and occurrence of adverbial clause modifier, auxiliary, negation modifier, marker, referent, open clausal complement, clausal complement, expletive, coordination, passive auxiliary, nominal subject, direct object, copula, and conjunct.

2.1 SVM based risk classification Model

Once the textual feedback is provided as an input to the system, it is first preprocessed and analyzed by the linguistic processing unit. The classifier learns each of the above linguistic features from a training sample of 5000 news articles collected from various online sources. We have primarily employed support vector machines (SVM) to develop a binary classifier that, given a news event would assign label ”Risk” or ”Not a risk” based on the textual properties. The SVM was implemented using the LIBSVM (Chang

and Lin, 2011) software. Further, we have applied the SVM recursive feature elimination method (Duan et al., 2005) to significantly reduce the number of features from the training sample. We have tested four types of kernels namely linear, polynomial, radial basis and sigmoid on the data, but we have presented results against only linear and polynomial kernels as the other two functions were found to be significantly poor performers.

2.2 The risk analyzer

Once "risks" are identified, they are passed to the risk analysis module for the identification of risk causes and their possible impacts. For this, we have followed a similar technique as discussed in (Chang and Choi, 2004). The identified risk along with its associated metadata like cause, impact, and time of arrival are stored into the risk register to generate risk reports. Determination of cause-effect pair from the risk statement consists of two parts.(a) Sentence Segmentation,(b) Cause-Effect classifier

We define a probable candidate for cause or effect phrase as a verb-rooted syntactic tree, which connects one noun phrase to the other with causal relation. Here, we propose a novel and robust dependency tree based sentence segmentation algorithm which considers the syntactic variation in the sentences such as the passive and verbal chains to effectively extract the probable cause-effect phrases from a risk statement spread across a single long sentence or multiple sentences. The algorithm of the segmentation module is depicted as follows:

Algorithm 1 *

dependency tree based sentence segmentation algorithm

Input: Valid Risk statement: S;
Output: A set of causality candidates: C
POS tag S and get the dependency tree (or trees for multiple sentences) D of S
Q=set of nodes of D which are verbs(in any form or tense) and the root(s) node of D
C=""
Visited[q]=FALSE for all nodes q ;

for each node q $\in$ Q **do**
    segment=TRAVERSE(q)
    C = C $\cup$ segment

end for

TRAVERSE (Node x)
S=x

for each child c of x **do**
    if ($visited[c] = FALSE and c \notin Q$) **then**
        Visited[c]=TRUE
        S=S $\cup$ TRAVERSE (c)
    end if
end for
return S

end

The cause-effect classifier classifies the candidate $t_i \in T$ into cause (C_0) or effect (C_1) or none (C_2). In a long sentence or in a multiple sentence statement there may be some parts which expresses neither cause nor effect. To capture this, we have introduced the class C_2 which denotes neither cause nor effect. For example, in the sentence:
Requirements for a project may change over its lifetime. Change in requirements will lead to change

in test cases and test data. This will affect the schedule planned for testing which in turn may lead to schedule-slippage of the entire project.

The cause/impact candidates we get are: t_1 : "Requirements for a project may change over its lifetime" , t_2 : "Change in requirements will lead", t_3 : "to change in test cases and test data" , t_4 : "This will affect the schedule" , t_5 : "planned for testing" , and t_6 : "which in turn may lead to schedule-slippage of the entire project". Here, the candidates t_1 and t_5 belongs to neither cause nor effect. Since, we are driven towards finding a solution using unsupervised learning method, it is difficult to learn the classifier parameter for C_2. Thus, we will compute the optimal class $C*$ of the candidate t_i as:

$$\begin{aligned} C* &= argmax_{C=C_0,C_1} P(C|t_i),\ if\ Dist(t_i) > \mu \\ &= C_2,\ otherwise \end{aligned} \tag{1}$$

Where,

$$Dist(t_i) = |\frac{log(P(C_0|t_i)) - log(P(C_1,t_i))}{log(P(C_0|t_i)) + log(P(C_1,t_i))}| \tag{2}$$

and,

$$P(C|t_i) = \frac{P(C) * P(t_i|C)}{P(t)} \approx P(T_i|C) \tag{3}$$

We have considered unigram, as the features of the candidate t_i. All these features are considered independent of each other. Therefore $P(t_i|C)$ can be written as $\prod_{k=1}^{k=|t_i|} P(W_{k,t_i}|c)$.

Where, $|t_i|$ denotes the total no of words in t_i after removal of stop words, and W_{k,t_i} denotes the k^{th} word in t_i. All the above defined probabilities can be learn from the cause-effect annotated data-set. In this paper we have considered raw corpora instead of annotated corpora to automatically learn these probabilities. In the following subsection we will discuss the technique in details.

There are three training stages. In the first stage, initial probabilities of naive bayes classifier was learned from bootstrapping. From the raw corpora, we have extracted few cause-effect pair automatically using some predefined patterns. From these extracted cause-effect pairs, initial probabilities of the classifier are learned. For example, some of the sample pairs are as follows:

X may result Y , If X then Y , X will affect Y, Y because of X.

Here, X is the cause phrase and Y is the effect phrase. We compute the Unigram probability as,

$$P(w|c) = \frac{(No\ of\ occurrences\ of\ w\ in\ class\ c + 1)}{(N + |V|)} \tag{4}$$

where N= no of words in class c and V=vocabulary size of the corpus. The parameters are estimated with Laplace smoothing method for out of vocabulary words in the training data. The second stage is called the expectation step. The remaining training corpus where cause-effect pair are not been identified by bootstrapping is classified with the current classifier. The final training stage is called the maximization step. From the newly cause-effect classified data parameters are re-estimated. Parameters trained in EM are word probability $P(w_{k,t_i}|c)$. The parameters are estimated using Laplace smoothing method for words unseen in the training data. The expectation and maximization step are repeated while the classifier parameters improve.

3 Experimentation and Evaluation

We have collected a corpus of around 7000 risk descriptions of a specific organization over the period of seven months. Each of the chosen risk descriptions were manually annotated by a group of project management experts. The annotation process involves identifying risk statements, their potential causes and impacts. 60% of the data is used for training the model and the rest for testing. We have evaluated the performance of both the risk classification system and risk analysis system by comparing its output with that of the expert annotations. We quantify the performance score in terms of the precision(P), recall(Re), F-measure(F) and accuracy(A) values (See Table 1). We have tested four type of kernels namely linear, polynomial, radial basis and sigmoid on the data. However, we have presented results against only linear and polynomial kernels as the other two functions were found to be significantly poor performers. To evaluate the quality of the classifications for SVM, multiple correlations (R) have been used.

Table 1: Evaluating the risk classifier. ALL is the combination of U,POS, D and Un features.

	Linear					Polynomial				
Features	**P**	**Re**	**F**	**A**	**R**	**P**	**Re**	**F**	**A**	**R**
U	82	89	85	83	.51	74	71	72	68	.46
B	71	73	72	74	.43	67	77	71	69	.40
POS	57	66	61	55	.67	51	63	56	51	.23
D	67	78	72	68	.63	69	78	73	74	.27
Wn	65	41	50	43	.43	54	47	50	53	.21
S	79	81	80	81	.39	72	76	74	73	.56
Un	77	76	76	79	.67	70	79	74	70	.67
ALL	**86**	**90**	**88**	**87**	**.71**	80	88	84	85	.73

4 Conclusion

In this demonstration we have presented a framework that processes human-reported risk descriptions to classify them into true risk and false alarm categories. In order to achieve this, we have used the SVM based machine learning framework and studied different linguistic features to automatically identify and label text descriptions as valid and invalid risks. The present work also extracts information from the text to generate reports on different causes of risks and their possible impacts as stated by human experts in their assessments. We have evaluated the classification framework by comparing the output of the system with that of the expert annotated dataset. Our evaluation showed promising results for automated risk identification.

References

Mark S Beasley, Richard Clune, and Dana R Hermanson. 2005. Enterprise risk management: An empirical analysis of factors associated with the extent of implementation. *Journal of Accounting and Public Policy*, 24(6):521–531.

Du-Seong Chang and Key-Sun Choi. 2004. Causal relation extraction using cue phrase and lexical pair probabilities. In *International Conference on Natural Language Processing*, pages 61–70. Springer.

Chih-Chung Chang and Chih-Jen Lin. 2011. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27.

Jennifer Coates. 1987. Epistemic modality and spoken discourse. *Transactions of the Philological society*, 85(1):110–131.

Kai-Bo Duan, Jagath C Rajapakse, Haiying Wang, and Francisco Azuaje. 2005. Multiple svm-rfe for gene selection in cancer classification with expression data. *IEEE transactions on nanobioscience*, 4(3):228–234.

Shimon Kogan, Dimitry Levin, Bryan R Routledge, Jacob S Sagi, and Noah A Smith. 2009. Predicting risk from financial reports with regression. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 272–280. Association for Computational Linguistics.

Jochen L Leidner and Frank Schilder. 2010. Hunting for the black swan: risk mining from text. In *Proceedings of the ACL 2010 System Demonstrations*, pages 54–59. Association for Computational Linguistics.

Hsin-Min Lu, Nina WanHsin Huang, Zhu Zhang, and Tsai-Jyh Chen. 2009. Identifying firm-specific risk statements in news articles. In *Intelligence and Security Informatics*, pages 42–53. Springer.

Adrian J Slywotzky and John Drzik. 2005. Countering the biggest risk of all. *Harvard Business Review*, 83(4):78–88.

Mourad Ykhlef and Danah Algawiaz. 2014. A new strategic risk reduction for risk management. *International Journal of Computational Intelligence Systems*, 7(6):1054–1063.