A Deep Learning Approach for Categorizing Risk Impact in Software Domain

Baala Mithra SM  
Global Technology Office  
Cognizant Technology Solutions  
Chennai, India  
balamithra.sm@cognizant.com

Kuheelee Roy  
Global Technology Office Cognizant  
Technology Solutions  
Chennai, India  
Kuheelee.Roy@cognizant.com

Sanglap Sarkar  
Global Technology Office  
Cognizant Technology Solutions  
Chennai, India  
sanglap.sarkar@cognizant.com

Venkateshwar Rao Madasu  
Global Technology Office  
Cognizant Technology Solutions  
Chennai, India  
Venkateshwar.madasu@cognizant.com

Subrahmanya VRK Rao  
Global Technology Office  
Cognizant Technology Solutions  
Chennai, India  
Subrahmanyavrk.rao@cognizant.com

Raj Bala  
Global Technology Office  
Cognizant Technology Solutions  
Chennai, India  
Raj.Bala@cognizant.com

Abstract - This paper addresses the problem of identifying the impact areas of risk from a given text description about the risk. The challenge of this piece of work lies in the fact that the description is in natural language. Literature provides a wide range of proposed framework where statistical machine learning techniques have been used to predict the risk from quantitative features. This work views the problem in a natural language processing perspective. In order to envisage a more accurate classification of the risk impact category we have used a deep learning paradigm.

Keywords - n-grams, convolution, deep learning, backpropagation, bag of words

I. INTRODUCTION

In software domain, the task of risk prediction requires analysis of historical data pertaining to similar projects for assessing two vital things viz, estimating the probability that the objectives of the project will be reached and that the objectives have been actually reached when certain risk have occurred. According to [1], a vital component of a risk management process is identifying and analyzing the risk data. The risk data, pertaining to a software project development usually contains attributes like category, exposure, stage, impact area etc. [2]. Traditional machine learning techniques have been efficiently used in combination with evaluation metrics like Neural Network and Support Vector Machine in order to compute the prior and posterior probabilities for the failure and success of the project. The influx of modern technology necessitates rapid progress in software development to support the same.

Identifying risks and the related impact areas requires analyzing the historical data along with the risk stage, exposure and status. Finding the relationship between the attributes characterizing a risk is a critical part in analyzing historical data related to risk in software domain. Studies on impact of risk factors in large-scale IT projects have been provided in [3] [4]. A list of current software risk items has been provided in [5]. The main idea behind analyzing historical data related to risk and predicting the impact of risk in future is to aid in anticipating and avoiding problems prior to their occurrence. [6]

Unlike other works in literature, this work sees the problem of predicting impact of risk, from a natural language processing perspective. The challenge lies in the fact that the traditional POS tagging and chunking techniques for processing sentences in natural language will not suffice for the current problem. In order to attain greater accuracy deep learning methodology has been used.

II. RELATED WORK

Few existing framework for risk management have been provided in [7-9]. According to [10], budget, schedule, technical qualities are the important factors used to evaluate if the project objectives are met. On the other hand, according to [11], the factor contributing to the success of risk analysis depends on the way a risk is described. In [1], metrics like Domain, KSLOC, and Complexity has been used in order to obtain the impact areas of risk. The issue of delay risk has been dealt with in [12]. Machine Learning techniques have been used for learning the relationship existing among various attribute characterizing a risk [1] [2]. In [13], the chances of unforeseen circumstances related to failure or damage in terms of monetary aspects has been dealt with.

Prioritizing the risk and classifying the impact into high, medium and low was also considered as a part of risk history analysis. Classifying risks from low to high including questionnaire was highlighted in [14].

Though fair accuracy rates have been achieved in using machine learning approaches for analyzing history of risk and their impact, a better approach encapsulating the semantic features as well as other metrics is still an area of concern. Unlike shallow learning approaches where the features are learned explicitly and classified, deep learning
deals with learning the features implicitly towards classification. Various deep learning architectures include deep neural networks [15], convolutional deep neural networks [16], deep belief networks [17], Restricted Boltzmann Machine (RBM) [18].

III. PROBLEM STATEMENT

A. Risk prediction

Given:
1. risk description (RD)
2. Risk category (RC)
3. Risk exposure (RE)
4. Risk status (RSS)
5. Risk stage (RST)
6. Impact area (RIA)

Expected outcome: Predicting the risk impact area.

The following nine categories of RIA(s) have been considered in this paper:
1. Cost
2. Legal
3. Quality
4. Timeline
5. Reputation
6. Budget Overrun
7. Future Revenue
8. Financial Penalties
9. Business Disruption

B. Shallow learning for classifying risk categories

IV. METHODOLOGY

A. Identifying Risk Token(RT)

As, mentioned in literature identification of risk at a higher accuracy rate is indispensable for the sake of attaining project completion objectives. Since the input for risk categorization is in natural language our proposed method attempts to use the deep learning model for Natural Language Processing as proposed by [15].
The following lexical features are considered:

- Left and right token of RT1
- Left and right token of RT2

1) **Paragraph Level Features**

The single word token as well as the pair of token limits the representation of the semantic composition of a risk description. The reason behind this is that pairs of token always might not suffice to represent the discriminating feature of a particular risk type by putting together long distance tokens within a paragraph. Paragraph describing a risk contains sentences. Windowing is applied on the entire paragraph without the consideration of stopwords separating two sentences. This leads to window feature (WF) and position feature (PF). This technique is similar to the one explored in [15]. Word features (WF) results in a vector of concatenated partitions of tokens in sentence(s). For example for a sentence with 6 words, each word starting from index 0 as given below:

\[ S: \{w0, w1, w2, w3, w4, w5, w6\} \]

The WF with window size 3 would result in a vector as follows:

\[ \{\{w0, w1, w2\}, \{w1, w2, w3\}, \{w2, w3, w4\}, \{w3, w4, w5\}, \{w4, w5, w6\}\} \]

While WF considers the entire paragraph, PF limits itself to one sentence only. The distance of the word (if contributes to positive value for BoW vector) is computed as follows:

Let the dictionary of uni-gram contains words: \( w_1, w_2, w_3, w_4, ..., w_n \)

Let the sentence contains words: \( s_1, s_2, s_3, s_4, ..., s_n \)

For word \( s \in \{s_1, ..., s_n\} \), that also belongs to the dictionary of uni-gram, the contribution of the word \( s \) is considered to be positive. Let such words in a particular sentence be denoted by \( s_{p1}, s_{p2}, ..., s_{pn} \)

For each of the words belonging to \( s_{p1}, s_{p2}, ..., s_{pn} \), the position feature is denoted as the distance between the first and the last word of the sentence the word belongs to. This contributes to the emphasis of the particular word based on its position in the sentence.

2) **Feature concatenation by Convolution**

A set of local features computed around words that contribute to the histogram of Bag-Of-Words. Each local feature is in turn the result of the windowing approach. Our next step is to apply a convolution function on the outputs of the window approach.

One-dimensional convolution is computed by dot product of the weight vector \( m \) with each of the outputs of the windowing approach. The type of convolution used is of type narrow that yields a vector of size smaller than the original ones.

**C. Convolutional Deep Learning of GRT**

The convolutions results in the following operation:

\[ c_j = W_1^T s_{j-(m-1)+1:j} \]

Where,

- \( j \) is the token number in the sentence
- \( m \) is the size of the filter (depends on varying window sizes eg, 3, 4, 5)
- \( n \) is the size if the hidden layer for each \( n_0 \) (with varying window sizes)
- The resultant feature map can be represented as:
  \[ c = \{c_1, c_2, ..., c_{n-m+1}\} \]

b) **Dealing with varying sentence size**:

In order to deal with varying sentence lengths, the maximum of each row in the matrix \( c \) is taken, denoted by \( c_{max} \). This represents the most useful feature in a particular dimension and is independent of the size of a sentence, which is further related to the size of a paragraph. This corresponds to the most distinctive feature corresponding to a particular filter.

c) **Selection of activation function**:

Hyperbolic tanh function is selected as the activation function:

\[ \tanh(x) \]
\[
\frac{d}{dx} \tanh x = 1 - \tanh^2 x
\]

This function can be used for the backpropagation training stage. Hence the non-linear transformation in this stage can be written as:

\[
g = \tanh(W_2c_{\text{max}})
\]

Where, \( W_2 \in \mathbb{R}^{n_2 \times n_0} \)

\( n_2 \) is the number of hidden layer 2

d) Final Layer

This output is fed as input to a softmax classifier that outputs the probability score for each of the category labels.

e) Training

The backpropagation network needs to learn the weights \( W_1 \) and \( W_2 \).

V. EXPERIMENTAL RESULTS

A. Data Description

<table>
<thead>
<tr>
<th>Training Data</th>
<th>Testing Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category 1</td>
<td>42</td>
</tr>
<tr>
<td>Category 2</td>
<td>31</td>
</tr>
<tr>
<td>Category 3</td>
<td>49</td>
</tr>
<tr>
<td>Category 4</td>
<td>37</td>
</tr>
<tr>
<td>Category 5</td>
<td>38</td>
</tr>
<tr>
<td>Category 6</td>
<td>44</td>
</tr>
<tr>
<td>Category 7</td>
<td>43</td>
</tr>
<tr>
<td>Category 8</td>
<td>48</td>
</tr>
<tr>
<td>Category 9</td>
<td>39</td>
</tr>
<tr>
<td>Category 1</td>
<td>20</td>
</tr>
<tr>
<td>Category 2</td>
<td>18</td>
</tr>
<tr>
<td>Category 3</td>
<td>17</td>
</tr>
<tr>
<td>Category 4</td>
<td>22</td>
</tr>
<tr>
<td>Category 5</td>
<td>15</td>
</tr>
<tr>
<td>Category 6</td>
<td>12</td>
</tr>
<tr>
<td>Category 7</td>
<td>20</td>
</tr>
<tr>
<td>Category 8</td>
<td>16</td>
</tr>
<tr>
<td>Category 9</td>
<td>22</td>
</tr>
</tbody>
</table>

B. Confusion Matrix using Shallow Learning

<table>
<thead>
<tr>
<th>Predicted</th>
<th>Cat 1</th>
<th>Cat 2</th>
<th>Cat 3</th>
<th>Cat 4</th>
<th>Cat 5</th>
<th>Cat 6</th>
<th>Cat 7</th>
<th>Cat 8</th>
<th>Cat 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error</td>
<td>0.65</td>
<td>0.17</td>
<td>0.13</td>
<td>0.33</td>
<td>0.083</td>
<td>0.6</td>
<td>0.373</td>
<td>0.18</td>
<td>0.091</td>
</tr>
</tbody>
</table>

C. Confusion Matrix using Deep Learning

Learning the risk description using the deep learning approach shows error rate that is comparatively lesser to that of using shallow learning approach.

VI. CONCLUSION

This paper addresses the challenges existing in the current methods of risk classification in terms of two aspects viz, features related to risk and shallow learning approaches. The contribution of this paper can be summarized as follows:

1. Unlike using numerical metrics for classifying risks, this paper deals with features that are written in natural language.
2. Furthermore, since the descriptions written are not complete in terms of parts of speech, extraction of features corresponding to each risk type plays an important role.
3. The last section provides comparative results in terms of error rate with respect to shallow learning and deep learning technique.

REFERENCES


[12] Morakot Choetkiertikul, Hoa Khanh Dam, Truyen Tran and Aditya Ghose, Characterization and prediction of issue-related risks in software projects, Proceedings of 12th Working Conference on Mining Software Repositories (MSR), co-located with ICSE 2015, IEEE (acceptance rate 30%). To Appear. (ACM SIGSOFT Distinguished Paper Award)


