SUBMISSION FORM

Please upload all required documents to your entry on the Challenge webpage: https://www.freelancer.com/contest/Risky-Space-Business-NASA-AI-Risk-Prediction-Challenge-2008562:

- Completed Submission Form (this document)
- Any files you refer to in this Submission Form (e.g., Diagrams or output files)

If you have any questions, please contact nasa-gcd@freelancer.com.

Submission Deadline: ***February 7, 2022 at 5:00 PM ET***

1. Submission Title: NASA AI Risk Prediction

2. Your Information

If you are participating as a team, please enter the Team Lead's details only. Please ensure the information provided below matches your approved Registration Form.

First Name: Alexander

Last Name: Poplavsky

Email: alexanderpoplavsky@yahoo.com

Freelancer.com Username: apoplavsky

3. Executive Summary [limit 500 words]

- Describe your solution and provide context for your solution for the judges.

The given solution comprises several AI text processing models and related algorithms to estimate the unknown project risk based on the historical Lessons Learned (LL) data and a target Project Plan (PP) information. The risk is estimated utilising the standard NASA classification: cost, schedule, technical, programmatic and green, yellow, red levels. The proposed solution has an advantage that it does not require creation of any new data formats, nor additional data or new risk classification. The existing documents are well suitable for the algorithms.

First, an unsupervised document classification model is trained on the LL data. It automatically generates jointly embeded label, document and word vectors and returns documents of topics modelled by manually predefined keywords (risk description). When trained, the model assigns risk categories to the LL entries. Then, the sentiment analysis of the LL summary data and the length of LL recommendations are used to calculate the risk value: from 0 (minimal risk) to 1 (maximal risk). Next, an unsupervised document embedding model is trained on the LL data in order to match the PP Technical Approach section text to the N most similar (in latent space) LL entries. The risks from those are combined and provided as a “category – value”, or “category – colour” (after applying thresholds).

The proposed solution is flexible, it allows using different text fields (either in LL or PP) for training models and fine-tuning parameters in order to improve prediction accuracy.

4. Proposed data to be collected [limit 1500 words]

- Recommended data fields and data structures to be used based on past data.
- Recommended new data fields to improve risk analysis.
- Describe the proposed structure.
    - Describe how the structure works with existing risk data (NASA's or another dataset) and how it will help predict risks and identify unknown risks on projects.
    - Provide a visual of your data fields and structure. For example, you can create a wireframe, picture, data schema, or diagram. Upload a file or provide a link.
- Describe how the structure supports program management needs, isn't overly constraining, and is easy to adopt.
- Describe how the output can be used by AI/ML to categorize and predict risk.

[1] The proposed solution utilises the LL historical data and a new PP information.

(a) The LL data is provided as a CSV table. The following entries are used for risk estimation: title, abstract, lesson(s) learned, driving event – for training unsupervised model for text classification by risk categories; title, abstract, lesson(s) learned – for sentiment analysis; recommendations – for length calculation. Abstract and lesson(s) learned are also used to train unsupervised text embeding model that results in similarity scores between LL entries and the new PP data. All the described text fields are extracted from the CSV table.

(b) For the text similarity model and unknown risk estimation, the "Technical Approach" section is extracted from the PP. The provided solution is flexible for using other sections from PPs as well. The target section may be prepared as a separate JSON file or extracted from a PP PDF directly. The provided code solves both of these tasks. The only requirements for the PP PDF are: it should contain a table of contents with section headings to be exactly the same as they appear further in the document, the page numbering should match the PDF pages with a tolerance within +/– 2 pages.

[2-4] The existing NASA project documents (LL table and a target PP) are suitable as inputs for this solution. However, as a potential improvement of the prediction quality, the historical project plans (at least, "Technical Approach" or other target section) should be attached to the LL entries.

The data extraction scheme is shown in FIG. 1.

5. Proposed method to populate the existing data into the new format [limit 1000 words]

- Describe the data extraction process, approach to working with GCD documents or your own dataset, the methodology applied to the process
- If you have an algorithm, demonstrate the data extraction process by showing the output from the algorithm. Provide the source code file name, instructions for use and compilation including any environmental requirements and dependencies, and sample outputs, if available. Upload the code to GitHub (instructions below).

[1] FIG. 1 shows the data extraction process utilised in the provided risk prediction system.

[2] The “ Technical Approach ” section text of a PP is used at the inference stage of the risk estimation model. This data could be prepared as a separate JSON file or extracted from the PP PDF. Both approaches are implemented and uploaded to GitHub. Please see the link to the code for mode information. The text may be read from the JSON file directly, or extracted from the PDF and then cleaned.

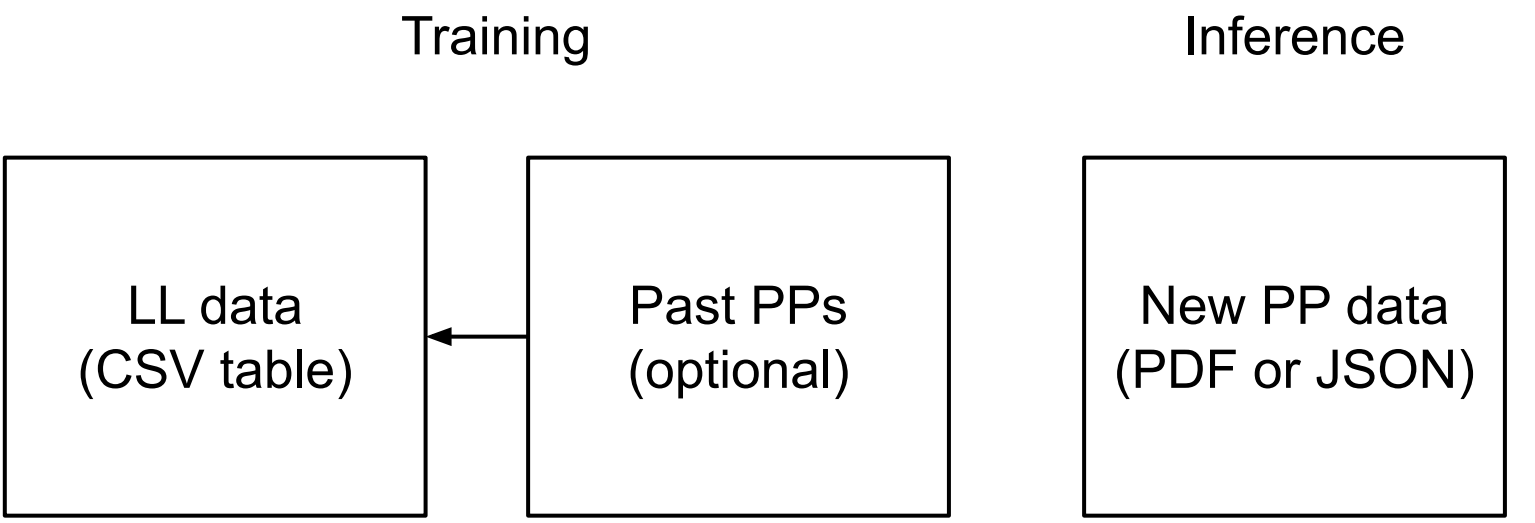

FIG. 1. Data extraction process.

6. Proposed method for predicting risks for each project given the new structured format [limit 1500 words]

- Describe the process, approach, and methodology using GCD data or your own dataset.
- Describe the method of building a library of risks and classifying risks from past projects
- Describe how you could use this to identify unknown risks

[1] The present solution is an AI based system that predicts future project risks based on the historical LL data. The output of the model is the standard NASA risk classification: cost, schedule, technical, programmatic and green, yellow, red levels. The given solution also provides risk values in the range [0, 1].

[2.1] Based on the NASA risk classification (https://www.nasa.gov/seh/6-4-technical-risk-management), the {risk – keywords} dictionary is created. Please see the GitHub link for more details.

[2.2] The risk keywords together with the LL data (title, abstract, lesson(s) learned, driving event) are used to train an unsupervised document classification model. The utilised learning framework is Lbl2Vec (https://pythonrepo.com/repo/sebischair-Lbl2Vec). The trained model predicts the most probable risk for a given LL entry.

[2.3] The title, abstract and lesson(s) learned columns are also used as inputs of a sentiment analysis. This is implemented based on the VADER-Sentiment-Analysis framework (https://github.com/cjhutto/vaderSentiment). The resulting scores are then converted to the range [0, 1], where 0 means positive and 1 – negative.

[2.4] In adition, the LL recommendation text length is calculated. The data is cleaned in the following way. Empty cells, or " see report/PDF/etc. " are ignored, and, to exclude outliers, the maximal length is calculated at the 0.95 quantile. The length is then converted to the range [0, 1] (minimal, maximal).

[2.5] The overall risk value is calculated in the following way: risk = a * risk_length + (1 – a) * risk_sentiment, where a = 0 if the LL recomendation data in missing, or a = 0.25 otherwise (the constant is subject to fine-tuning, please see the code).

[3] After the steps [2] are done, we find N most similar entries to a given PP text (" Technical Approach " section) and their scores.

[3.1] This is achieved by training an unsupervised text embedding model, based on the Doc2Vec framework (https://radimrehurek.com/gensim/models/doc2vec.html). This model learns paragraph and document embeddings via the distributed memory and distributed bag of words models. The model is trained using the LL data: abstract, lesson(s) learned.

[3.2] Inference:

INPUT (PP " Technical Approach ") => DOC2VEC MODEL => OUTPUT (N most similar LL entries and scores)

[3.3] We next combine the risks [2.5] of the found N most similar entries. The same risk values are averaged with the weights proportional to their scores [3.2]. The missing risk categories are added with 0 values.

[3.4] The output of [3.3] is a dictionary: {risk: value}, for example:

{'cost': 0, 'schedule': 0, 'technical': 0.6043, 'programmatic': 0.4282}

[3.5] The risk values [3.4] can be categorised by green/yellow/red levels, the thresholds may not necessarily be linear. For example, green/yellow = 0.25, yellow/red = 0.6. They are subject to fine-tuning and optimisation.

7. <u>If external data sources are used in your analysis</u>, provide information about the source. For example, type of data, how it addresses risk, and the output risk analysis. Provide as much information as you can about the data source. Leave blank if not applicable.

Not applicable.

8. <u>If any algorithms are included</u>, please provide the following information:

- Demonstrate the process.
- Provide the source file name, instructions for use and compilation including any environment requirements and dependencies, and sample outputs, if available.

The scheme of the risk prediction system is shown in FIG. 2. The code with more details is provided in the GitHub repository.

To share your code:

1. Upload your solution and code to a private GitHub repository.
2. Add "***enterprise-admin@freelancer.com***" as a collaborator.
3. Make sure "***enterprise-admin@freelancer.com***" has permissions to fork your repository.

Your GitHub Username: alexanderpoplavsky

Your GitHub Repository Name: risk_prediction

Link to your GitHub Repository: https://github.com/alexanderpoplavsky/risk_prediction

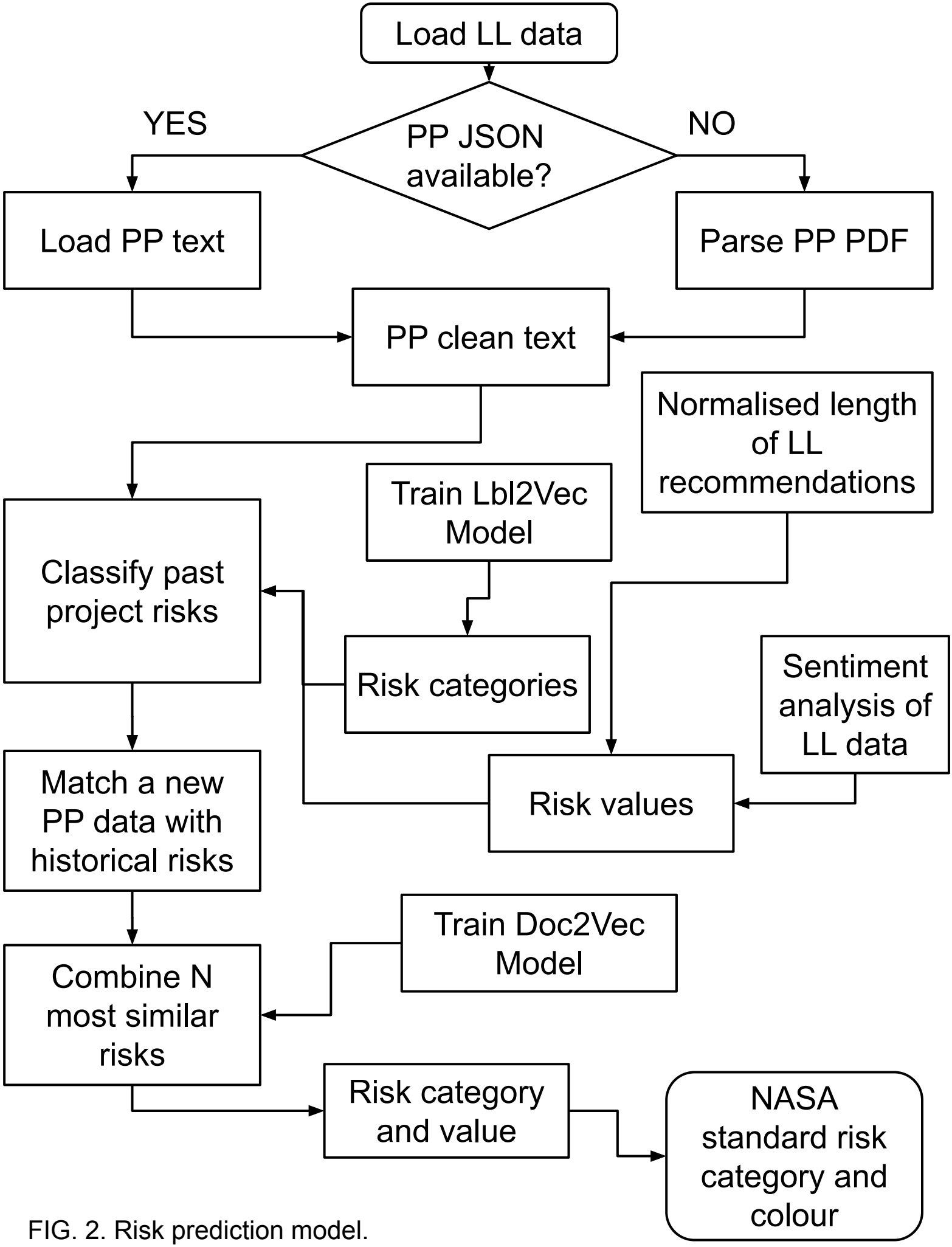

FIG. 2. Risk prediction model.

CERTIFICATION

- ☑ I certify that all information included in this submission is my own work.
- ☑ If algorithms are included, I certify that I have provided my GitHub information.
- ☑ I certify that I have read and abided by all rules of the competition.

Alexander Poplavsky

Signature of Team Lead

06 February 2022

Date

Please upload all required documents to your entry on the Challenge webpage: https://www.freelancer.com/contest/Risky-Space-Business-NASA-AI-Risk-Prediction-Challenge-2008562

If you have any questions, contact nasa-gcd@freelancer.com.

Submission Deadline: ***February 7, 2022 at 5:00 PM ET***