Data Science Project On Crowdsourced Smart Home Requirements

02423555
MSc statistics (applied statistics)
qz1723@ic.ac.uk

May 15, 2024

Abstract

Over the years, crowdsourcing has gained significant popularity as a method for collecting data from a large and diverse group of individuals, providing valuable insights into creativity. This project aims to predict the creativity score of proposed requirements based on their application domains and tags. Initially, the project involved preprocessing the dataset to observe its characteristics. Subsequently, Natural Language Processing (NLP) techniques were applied to predict the tags and application domains of the requirements. Finally, two different models were employed to predict the creativity of the requirements, which is determined by their novelty and usefulness, using the identified application domains and tags.

Index Terms—Crowdsourcing, Creativity, Hypothesis Testing, NLP, Artificial Neural Network, Random-forest Classifier

1 Introduction

This project focuses on analyzing a dataset of crowdsourced smart home requirements, which includes creativity ratings, as presented in the study by Murukannaiah et al. [2016]. The dataset, "A Dataset of Crowdsourced Smarthome Requirements with Creativity Ratings," comprises comprehensive data collected from crowd workers engaged in generating smart home requirements. It includes presurvey and postsurvey questions and responses, DISC personality assessments, IPIP scores, user information, and creativity ratings for the requirements. This dataset was used to study the influence of emotions and teamwork on the novelty and usefulness of crowd-generated requirements, facilitating the development of models to predict creative outcomes efficiently.

Based on the dataset, the following research questions are addressed in this report:

- **RQ1:** Do crowd workers' emotions influence the novelty and usefulness of the proposed scenarios?
- **RQ2:** Do certain tags or application domains promote or inhibit crowd creativity? If so, what is the evidence?
- **RQ3:** Do crowd workers' emotions influence the efficiency (time spent) of a scenario?
- **RQ4:** Can we predict the novelty and usefulness of a given requirement scenario?

The subsequent sections will address these research questions one by one.

2 Data Collection

To address the proposed research questions, we conducted a study using Amazon MTurk workers, both individually and in asynchronous teams. We utilized two distinct datasets related to smart home use case scenarios, referred to as the 'solo-work' (2016 dataset) and 'group-work' (2022 dataset) datasets.

The 'group-work' dataset Murukannaiah et al. [2022] comprises 1,823 scenarios generated by 323 workers, with 639 scenarios rated for creativity on a scale from 1 to 5. This dataset includes information on group type (solo or group) and group size, as well as time taken (efficiency) and emotions (enjoyment, boredom, confidence, and anxiety) rated from 1 to 5. It also measures four personality traits using the DISC model:

Dominant, Influential, Steady, and Conscientious, and includes data on solo workers' personality types and team compositions.

The 'solo-work' dataset Murukannaiah et al. [2016] consists of 2,966 scenarios created by 609 workers. Similar to the group-work dataset, it includes personality trait information (with five traits instead of four) and ratings for novelty, usefulness, and clarity on a scale from 1 to 5. Workers selected an application category (health, safety, energy, entertainment, or other) and provided descriptive tags for each scenario.

Both datasets have limitations. The data were collected using tests of uncertain reliability, and their validity depends on the accuracy of participant responses. Some data manipulation is necessary before analysis.

2.1 Solo-Work Dataset

To prepare this dataset for exploring creativity, we first extract the novelty and usefulness ratings for each requirement, along with the application domain and tags. Each requirement is rated multiple times by different users, so we compute the mean score for both usefulness and novelty. Initially, we observe that a requirement can belong to an 'Other' application domain, with additional details provided. We include this in our exploratory analysis. After cleaning the dataset by removing empty rows and incorrect inputs, we retain 2,882 requirements produced by solo workers.

2.2 Group-Work Dataset

Although a processed format of the group-work dataset is available, we find the raw format more useful for exploratory analysis. From the raw dataset, we first retrieve emotion ratings and personality types from each user's post-survey responses, which include ratings for enjoyment, boredom, confidence, and anxiety. We then identify the group type that created each requirement. The subsequent steps mirror those for the solo-work dataset, where we calculate the mean scores for usefulness and novelty for each requirement. This dataset lacks application domains or tags. After cleaning the data by removing empty rows and incorrect inputs, we are left with 227 requirements.

3 Data Exploration

Initially, I analyzed whether different personality types have a significant influence on emotions. As shown in Figure 1, different personality types exhibit only minor influences on the overall distribution of emotions.

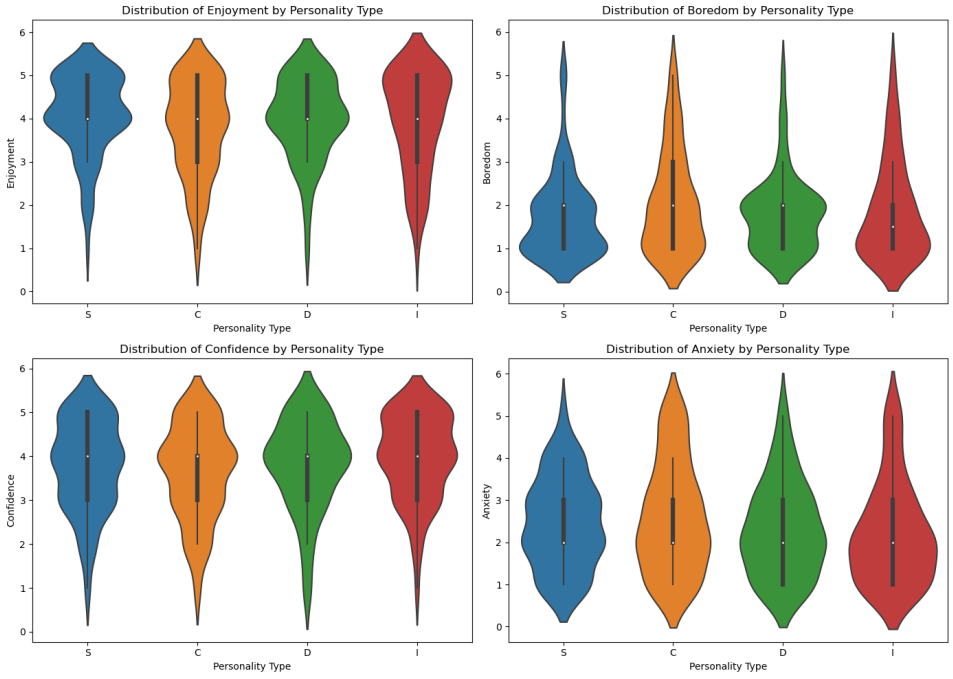

Figure 1: Violin plot for different Personality distribution on each emotion

Additionally, I investigated the frequency of the tags used for the requirements. The most frequent tags are displayed in Figure 2 below.

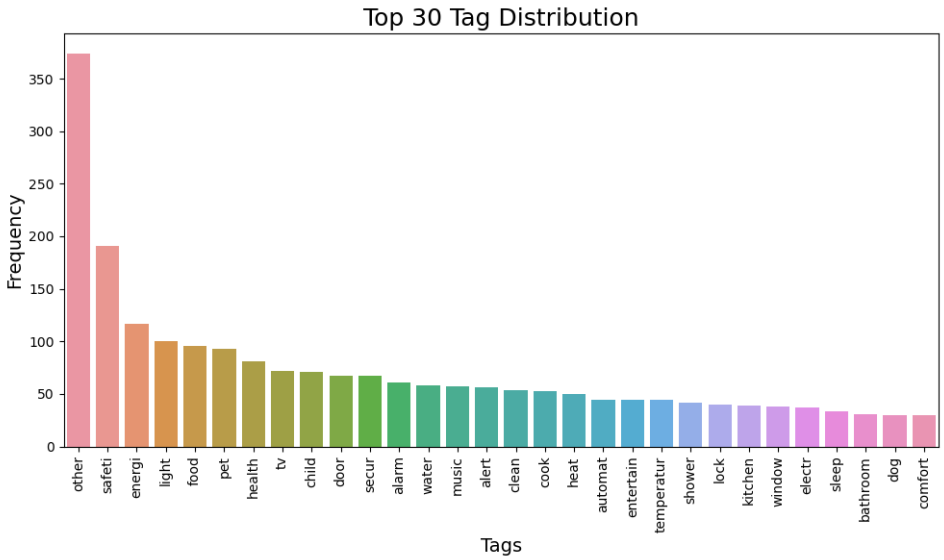

Figure 2: Top 30 frequentest tags

Furthermore, Figure 3 below illustrates the distribution of the tags, highlighting that a small number of tags account for the majority of occurrences.

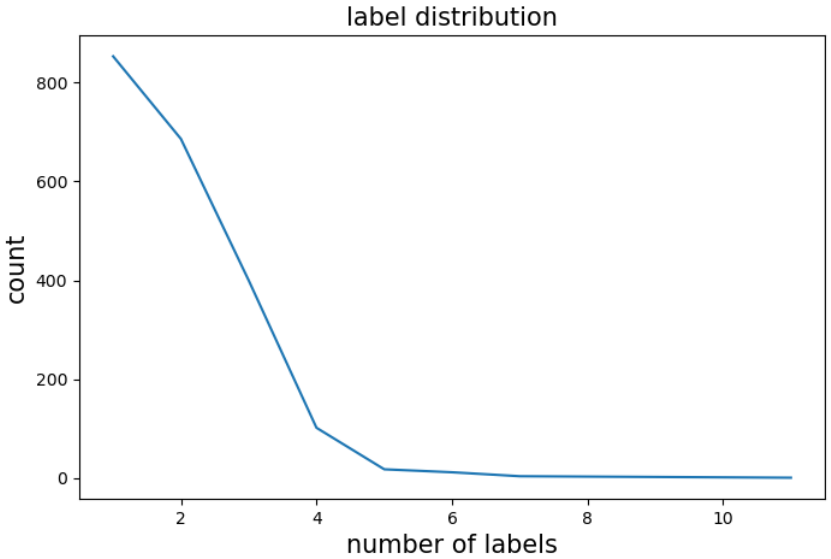

Figure 3: Tag Distribution

3.1 Hypothesis Testing

I have implemented the hypothesis testing method to determine if there is a correlation between emotions/personality and novelty. To test these hypotheses, I use the Kruskal-Wallis test, a non-parametric statistical method that compares the distributions of two or more ordinal samples at a 10% significance level Pachori et al. [2015].

Most of the null hypotheses were retained, indicating that the distributions are generally the same despite differences in personality and emotions. However, the following hypotheses were rejected:

- **H01:** For individuals, the influence of personality on boredom is not significant.
- **H02:** For individuals, personality has a significant relation to efficiency.

These results are intuitive, as individual workers are likely influenced by their personality and may become bored more easily when working alone. The findings suggest that boredom might significantly impact individual work and consequently affect their efficiency.

4 NLP

Furthermore, I have implemented Natural Language Processing (NLP) to predict the application domain and tags for the requirements that lack this information. In the individual workers' dataset, the requirements.csv file includes application domain information. However, the group workers' dataset does not contain similar information. Therefore, I devised a method to train an NLP model using the data from the individual workers' dataset and subsequently use this model to predict the application domain and tags for the group workers' dataset.

During the preparation stage, we convert the tags and domains from the solo-work dataset to a numerical format. This conversion is necessary to ensure compatibility with subsequent classification models such as Support Vector Classification (SVC). We start by encoding tags. Predicting tags is a multi-label classification problem, where each scenario may be assigned one or more tags. We convert the tags using a multi-label binarizer from Scikit-learn. Matrix 1 is an example of the resulting binary matrix, in which each column corresponds to a unique label, and each row corresponds to the tags assigned to a particular scenario.

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

On the other hand, predicting the application category is a multi-class classification task where each scenario is assigned to only one application domain. We use a one-hot encoder to convert the application domains to a binary sparse matrix, see Matrix 2. In this matrix, each column corresponds to a unique application domain, and each row corresponds to the application domain assigned to a particular scenario.

$$\begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

The scenarios used in the model are a combination of the 'feature' and 'benefits' of scenarios from the solo-work dataset, and the combination of 'context' and 'stimuli' from the group-work dataset.

We utilize word-embedding techniques to extract features from all of the collected scenarios. The idea of word-embedding is to convert text into numerical form Mikolov et al. [2013]. In particular, we use Count Vectorizer (CV) and Term Frequency-Inverse Document Frequency (TF-IDF). CV counts the occurrence of each word in a dictionary format, whereas TF-IDF assesses the significance of each word in the context of the scenario. It is defined as

$$\text{TF-IDF}(w, d, D) = \text{TF}(w, d) \times \text{IDF}(w, d, D),$$

where w denotes a word, d is a document, and D refers to all documents. The term frequency, $\text{TF}(w, d)$, of a word w is expressed as

$$\text{TF}(w, d) = \frac{\text{count of word } w}{\text{total word count in document } d}.$$

The inverse document frequency, $\text{IDF}(w, d, D)$, determines if the word w appears in all documents. It is defined as

$$\text{IDF}(w, d, D) = \log \frac{|D|}{|\{d \in D : w \in d\}|}.$$

4.1 Model: Train, Fit, and Predict

We take a subset of the encoded tags and application domains and the embedded scenarios from the solo-work dataset as training data. We fit different models by combining an embedding technique (either CV or TF-IDF) with a classification technique (either Multinomial Naive Bayes, Linear SVC, or Logistic Regression). Then, using the embedded scenarios from the group-work dataset, we predict their application categories and tags.

To evaluate the performance of the model combinations, we use three metrics. The first is the micro F1-score, computed in terms of the precision, P, and recall, R, values across all N classes:

$$\text{Micro F1-score} = \frac{2RP}{R+P} = \sum_{i=1}^{N} \frac{2\text{TP}_i}{2\text{TP}_i + \text{FP}_i + \text{FN}_i},$$

where TP_i, FP_i, and FN_i are the true positives, false positives, and false negatives for the i^{th} class, respectively. The next metric is the macro or unweighted F1-score, calculated as the mean of the F1-score across N classes:

$$\text{Macro F1-score} = \frac{1}{N} \sum_{i=1}^{N} F_i,$$

where F_i denotes the i^{th} F1-score. Lastly, the weighted F1-score considers the distribution of the data, giving less weight to classes with fewer instances:

$$\text{Weighted F1-score} = \sum_{i=1}^{N} \frac{w_i F_i}{\sum_{i=1}^{N} w_i},$$

where w_i denotes the number of instances of the i^{th} class.

Figure 4 below shows the performance of the NLP on the application domain of the requirement.

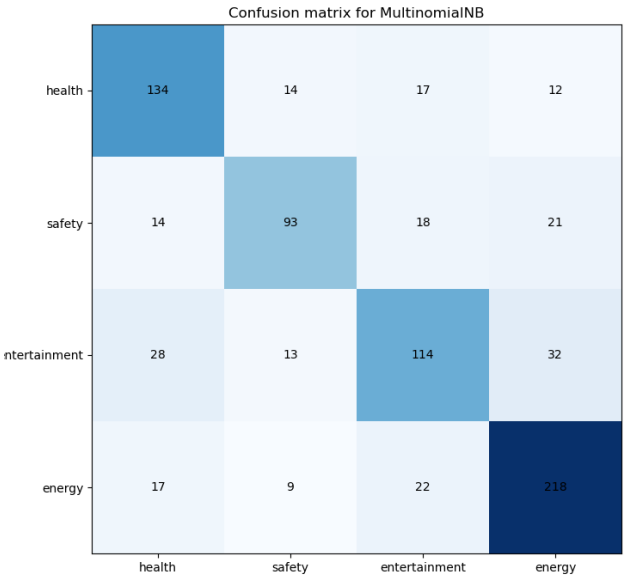

Figure 4: MultinomialNB confusion matrix

Figure 6 and 5 below shows the performance of the NLP on the tags of the requirement.

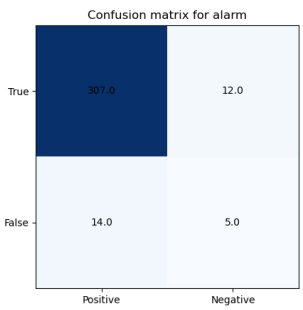

Figure 5: alarm confusion matrix

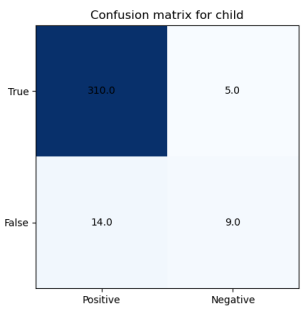

Figure 6: child confusion matrix

These models are subsequently used to predict the tags and application domains for the group-work dataset.

5 Data Modelling

To predict usefulness and novelty, we propose three models which can predict how novel and useful a requirement may be. As discussed, our features for this model are the application domain and tags of a requirement; and the labels are the usefulness and novelty ratings. We allocate 80% of the data for training and 20% for testing. Where applicable, we also split the training data into training and validation sets with an 80-20 split, respectively.

5.1 Random Forest Approach

The Random Forest algorithm is an ensemble learning technique used for both classification and regression tasks that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees.

The CART algorithm employs a decision tree approach suitable for handling both classification and regression tasks. Initially, it defines a Gini impurity function for a dataset D parameterized by θ :

$$\text{Gini}(D, \theta) = \sum_{i=1}^{n} p\left(x_i\right) \times \left(1 - p\left(x_i\right)\right),$$

where $p\left(x_i\right)$ denotes the probability of occurrence for class x_i, and n is the number of classes. The objective is to achieve a lower Gini score, which indicates higher purity within the dataset.

In constructing the decision tree, CART divides the dataset into two subsets, D_1 and D_2, based on a feature condition $A(x) = a$, and then computes the conditional Gini index:

$$\frac{|D_1|}{|D|} \text{Gini}\left(D_1, \theta\right) + \frac{|D_2|}{|D|} \text{Gini}\left(D_2, \theta\right).$$

For discrete features, CART sorts the feature values and considers midpoints between adjacent values as potential binary classification points. It selects the split point that minimizes the Gini coefficient, effectively discretizing the continuous feature into two categories. This strategy ensures the optimal split by minimizing the Gini impurity, thus optimizing the structure of the decision tree:

$$\theta^* = \text{argmin}_\theta \, G(D, \theta).$$

The best-tuned model has been saved, and the confusion matrix of its predictions is shown in Figure 8. As evident from the figure, the results are not satisfactory, as the model predominantly predicts a novelty rating of 4 and a usefulness rating of 5 for most entries. Therefore, I have applied another model.

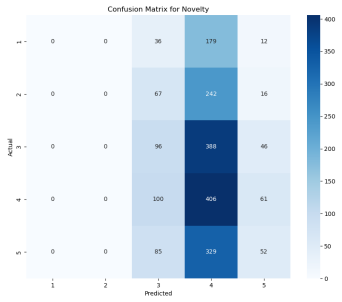

Figure 7: RF confusion matrix novelty

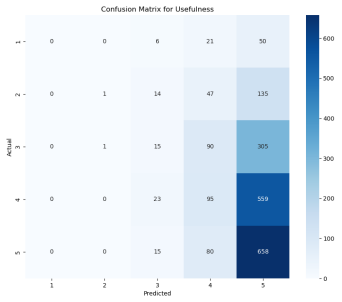

Figure 8: RF confusion matrix usefulness

5.2 Artificial Neural Network Approach

The implementation of the Artificial Neural Network (ANN) process involves following key steps, designed to manage data through a series of computational layers to achieve effective learning and prediction:

Step 1: **Forward Propagation** - Each input vector is processed across multiple layers of the network. Activation functions are applied at each layer to transform the data as follows:

$$\begin{aligned} Z^{(2)} &= \Theta^{(1)} X, \\ a^{(2)} &= g\left(Z^{(2)}\right), \\ &\vdots \\ a^{(n)} &= g\left(Z^{(n)}\right), \end{aligned}$$

where $a^{(n)}$ represents the final output of the forward propagation.
Step 2: **Cost Function Computation** - The model's performance is evaluated by a cost function(I will here use the corss entropy loss as an example), which includes a cross-entropy term for prediction accuracy and a regularization term to avoid overfitting:

$$J(\Theta) = \text{Cross-entropy}(y, \hat{y}) + \frac{\lambda}{2M} \overset{\text{layer number}}{\sum_{l=1}} \sum_{i=1}^{m} \sum_{j=1}^{n} (\theta_{ji}^{(l)})^2,$$

with λ as the regularization parameter.
Step 3: **Backpropagation** - This step involves calculating the gradient of the cost function with respect to each weight in the network by propagating errors back through the network:

$$\delta^L = \frac{\partial C}{\partial z^L} = \frac{\partial C}{\partial a^L} g'(z^L),$$

and the error for each layer is computed as:

$$\frac{\partial C}{\partial \Theta_{jk}^l} = a_k^{l-1} \delta_j^l.$$

Step 4: **Gradient Descent** - The weights are updated by applying gradient descent to minimize the cost function. The learning rate α is critical to ensuring smooth and effective convergence.

The best-tuned model has been saved, and the confusion matrix of its predictions is shown in Figure 10. As evident from the figure, the results are more satisfactory compared to the random forest (RF) results. This artificial neural network (ANN) model accurately predicts high usefulness and novelty ratings but performs relatively poorly when predicting low usefulness and novelty ratings.

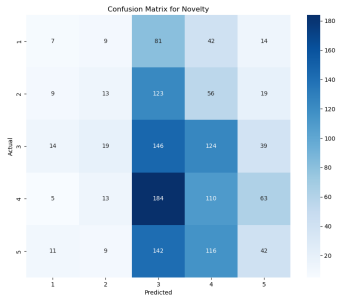

Figure 9: RF confusion matrix novelty

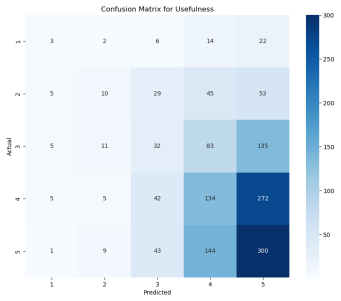

Figure 10: RF confusion matrix usefulness

6 Discussion

With the implementation of the models and the analysis of the dataset, we can now address the research questions presented in the introduction.

- **RQ1:** The answer is no. The results show that group workers have a high tolerance for emotions, especially boredom. Surprisingly, for individual workers, lower work efficiency does not decrease novelty and usefulness.

- **RQ2:** Given that tags and application domains successfully predict some novelty and usefulness in the ANN model, certain tags or application domains likely influence creativity positively. Due to time constraints, identifying the specific tags that most influence creativity is suggested for future research.
- **RQ3:** The answer is partially yes. Group workers exhibit high tolerance for emotions, particularly boredom. However, in individual work scenarios, emotions definitely influence novelty and usefulness.
- **RQ4:** Yes, we can confidently predict the novelty and usefulness of a proposed requirement. We have proposed three valid and distinct models for this purpose.

7 Conclusion

In this study, I explored the impact of emotions and personality on the creativity of smart home requirements generated by crowd workers. Our findings indicate that group workers exhibit a high tolerance for emotions, particularly boredom, and that individual workers' efficiency does not significantly affect the novelty and usefulness of their contributions. Using NLP techniques, we successfully predicted application domains and tags, suggesting certain tags positively influence creativity. We developed and validated models to predict the novelty and usefulness of requirements, providing a foundation for future improvements in creativity prediction.

8 Future Works

To enhance this work, the first improvement would be to refine the ANN. The current model is quite simple, with very few layers, which may be a significant reason it does not predict creativity effectively. Additionally, exploring other models could provide better results. Finally, while we have made progress, the question of whether certain tags or application domains promote or inhibit crowd creativity remains unresolved. Future work should focus on identifying the specific tags or application domains that most positively influence creativity.

References

T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013.

P. K. Murukannaiah, N. Ajmeri, and M. P. Singh. Acquiring creative requirements from the crowd: Understanding the influences of personality and creative potential in crowd re. In *2016 IEEE 24th International Requirements Engineering Conference (RE)*, pages 176–185. IEEE, 2016.

P. K. Murukannaiah, N. Ajmeri, and M. P. Singh. Enhancing creativity as innovation via asynchronous crowdwork. In *Proceedings of the 14th ACM Web Science Conference 2022*, pages 66–74, 2022.

R. B. Pachori, P. Avinash, K. Shashank, R. Sharma, and U. R. Acharya. Application of empirical mode decomposition for analysis of normal and diabetic rr-interval signals. *Expert Systems with Applications*, 42 (9):4567–4581, 2015.