
Title: Report on Research Paper Classification Using AutoGluon

1. Introduction

The goal of this project is to classify academic research papers into two categories: **Publishable** and **Non-Publishable**. Papers classified as **Publishable** are further categorized into five specific conference categories: **CVPR**, **EMNLP**, **KDD**, **TMLR**, and **NeurIPS**. The project leverages machine learning to automate the classification process, using text-based features extracted from the papers.

2. Dataset Description

2.1 Structure

- **Publishable Papers**: Papers are divided into subcategories representing specific conferences.
- **Non-Publishable Papers**: A separate category for papers deemed non-publishable.
- The dataset contains a total of 15 PDF files:
    - **Publishable**: Divided into five subfolders (one for each conference), each containing 2 PDF files.
    - **Non-Publishable**: A single folder containing 5 PDF files.

2.2 Challenges

- **Class Imbalance**: The dataset is heavily imbalanced with more **Non-Publishable** papers.
- **Limited Data Size**: Only 15 PDFs are available, which presents challenges for training robust machine learning models.
- **Feature Scarcity**: Some PDFs provide limited textual data for feature extraction.

3. Feature Engineering

3.1 Text Extraction

Text was extracted from the PDFs using the PyPDF2 library. This process converts the PDF content into plain text for further analysis.

3.2 TF-IDF Vectorization

- **TF-IDF** (Term Frequency-Inverse Document Frequency) was applied to convert the extracted text into numerical features suitable for machine learning models.
- The following parameters were used:
    - Maximum number of features: 1000
    - Removal of common stopwords

3.3 Label Encoding

- Conference labels (CVPR, EMNLP, etc.) were encoded using LabelEncoder to make them compatible with machine learning models.

4. Training Methodology

4.1 Binary Classification: Publishable vs Non-Publishable

- **SMOTE (Synthetic Minority Over-sampling Technique)** was used to handle the class imbalance by generating synthetic samples for the minority class (Publishable).
- **Logistic Regression** was chosen for its simplicity and efficiency in binary classification tasks.

4.2 Multi-Class Classification: Conference Categories

- A **Random Forest Classifier** was trained to categorize Publishable papers into specific conferences.
- **Class weights** were calculated to address imbalances in the dataset.

4.3 AutoGluon Framework

- The AutoGluon library was employed for automated model selection and hyperparameter tuning.
- The **medium preset** was used, balancing training time and accuracy.

5. Evaluation

5.1 Metrics

- **Precision**, **Recall**, and **F1-Score** were used to evaluate the model's performance in both classification tasks.

5.2 Publishability Classification Results

Metric	Non-Publishable	Publishable	Overall
Precision	0.95	0.10	0.79
Recall	0.68	0.50	0.67
F1-Score	0.79	0.17	0.75

6. Challenges Faced

1. **Class Imbalance**:
    - The dominance of the Non-Publishable category required special handling.
    - **Solution**: SMOTE was applied to balance the classes during the binary classification task.
2. **Limited Data Size**:
    - The small number of training samples (15 PDFs) posed challenges for the model's generalization.
    - **Solution**: Additional techniques, like SMOTE, helped mitigate the effects of limited data.
3. **Imbalanced Subcategories**:

- The conference subcategories were heavily imbalanced, affecting the model's ability to classify papers into the correct conference.
- **Solution**: Class weights were computed to address this issue.

4. **Noisy Data**:
   - The text extracted from the PDFs included artifacts such as headers, footers, and other irrelevant content.
   - **Solution**: Text preprocessing steps, including stopword removal, were implemented to clean the data.

7. Improvements and Future Work

1. **Pretrained Models**:
   - Future work should include the use of pretrained models such as **BERT** or **SciBERT** to better understand the context of research paper content.
2. **Data Augmentation**:
   - Simulating additional training data using techniques like paraphrasing or sourcing more PDFs will help improve model generalization.
3. **Conference-Specific Keywords**:
   - Extracting domain-specific keywords for each conference could improve the classification accuracy for conference categories.
4. **Hierarchical Classification**:
   - Implementing a two-step hierarchical model could first predict whether a paper is publishable and then categorize it into the appropriate conference.

5. **Improved Feature Engineering**:
    - Future work could involve using more advanced embeddings like **Word2Vec**, **GloVe**, or **Transformers** to better represent the textual content of the papers.

8. Conclusion

This project demonstrates the potential of using machine learning to automate the classification of research papers into **Publishable** and **Non-Publishable** categories, as well as assigning them to relevant conferences. Despite facing challenges like class imbalance and limited data, the use of the AutoGluon framework allowed for efficient model training and selection. Further improvements, such as data augmentation and the use of pretrained models, could enhance the performance and robustness of the system, making it a valuable tool for automating the paper review process for academic conferences.
