Deriving Domain Models from User Stories: Comparison of ML and LLM models

Adithiyan Rajan Indira Saravanan - 300421518, Aditya Rawal - 300442176, Thamil Vani Samiyappan - 300408700

1. Introduction

The project aims to automate the derivation of domain models from user stories using Machine Learning (ML) and Large Language Models (LLMs) to improve productivity and efficiency in software engineering. Automating this process can enhance consistency and completeness, reducing the risk of missing critical features and concepts. However, automating domain models' derivation remains challenging due to the ambiguity and imprecision of natural language requirements. Domain models are crucial in software engineering as they serve as a common language between stakeholders and provide a clear representation of the underlying structure guiding the development process

2. Related Work

Bragilovski, Dalpiaz et al [1] compare the performance of human evaluators to rule based NLP approach, a machine learning classifier and a generative AI approach with LLMs through prompt engineering. Due to a lack of benchmark datasets, they develop a dataset of user stories for this task and tag the classes and associations for evaluation. This advances their previous work in [2] which focuses on the task of recommending relationships between entities in a domain model using supervised machine learning models. It is assumed that these entities were previously extracted from a user story collection. We plan to build on this work in [1] by leveraging the proposed dataset and methodology to apply it to other machine learning models and more recent large language models and compare their performance.

3. Research Questions

- **Comparison of ML Models**
  How do different machine learning models (e.g., SVM, Logistic Regression, Gradient Boosting Networks, Neural Networks, BERT, Ensemble models) compare in their performance for deriving domain models from user stories?
- **Comparison of LLMs**
  How do various recent Large Language Models (e.g., GPT-4, Llama 3, Mistral 7B) compared to each other for the derivation of domain models from user stories?
- **Potential Additional Research Questions**:
  1. Can ensemble methods that combine ML and LLM approaches outperform individual methods?
  2. Can preprocessing or NLP techniques improve ML model accuracy?
  3. Do prompt engineering techniques (e.g., few-shot learning, chain-of-thought prompting) improve results?
  4. How do LLMs of varying sizes perform in domain model derivation?

4. Dataset

The dataset proposed in [1] comprises 9 projects with a total of 487 user stories from online repositories and masters level course projects on requirements engineering. It primarily consists of functional requirements and domains that are understandable with limited domain knowledge. The authors independently tagged the classes and associations to create the dataset, resolving discrepancies through discussions and finalizing the gold standard upon agreement. Fleiss' Kappa was used to measure reliability, showing substantial agreement for class identification and moderate to substantial agreement for associations. The dataset is available on [3].

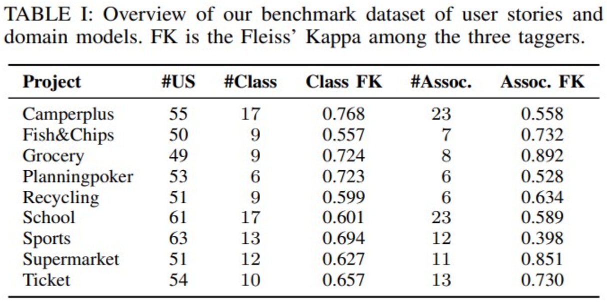


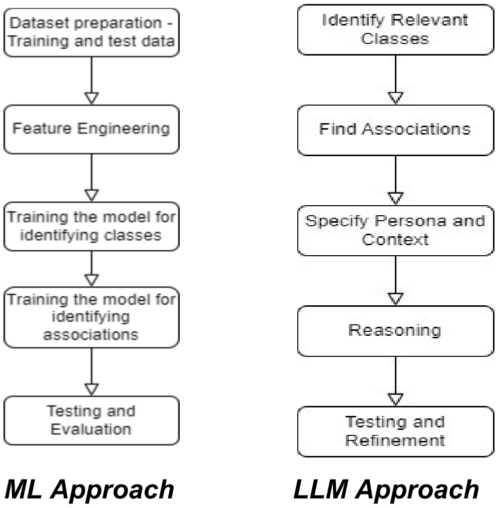
TABLE I: Overview of our benchmark dataset of user stories and domain models. FK is the Fleiss' Kappa among the three taggers.

Project	#US	#Class	Class FK	#Assoc.	Assoc. FK
Camperplus	55	17	0.768	23	0.558
Fish&Chips	50	9	0.557	7	0.732
Grocery	49	9	0.724	8	0.892
Planningpoker	53	6	0.723	6	0.528
Recycling	51	9	0.599	6	0.634
School	61	17	0.601	23	0.589
Sports	63	13	0.694	12	0.398
Supermarket	51	12	0.627	11	0.851
Ticket	54	10	0.657	13	0.730

Example of user stories, classes identified and associations

As a customer, I want to be able to sign up for the newsletter, so that I am aware of discounts and anomalies.
As a customer, I want to be able to add products to my wishlist, so that I can easily find them in the app.
As an employee, I want to update the application with new recipes. etc.
Classes Identified - newsletter, customer, recipes, products, wishlist etc.
Associations - (customer,wishlist) (customer,newsletter) (wishlist, product) etc. Source Project – Supermarket

5. Overview


Dataset preparation - Training and test data
Feature Engineering
Training the model for identifying classes
Training the model for identifying associations
Testing and Evaluation
ML Approach
Identify Relevant Classes
Find Associations
Specify Persona and Context
Reasoning
Testing and Refinement
LLM Approach

LLM Based Approach: The LLM based approach involves using Large Language models to derive domain models from user stories. It begins by identifying potential classes from the dataset and refining them by removing irrelevant classes. The next step is to find associations among the identified classes. The model adopts the persona of a Requirements Engineer specializing in domain modeling and is provided with contextual information to aid in the selection process. The approach emphasizes reasoning and asks the model to explain the decision-making process to enhance reliability. Testing will be conducted using specific user stories from the dataset.

ML Based Approach: In the ML-based approach, datasets are selected for training and testing the model. Feature engineering is then performed to design features for identifying classes and associations. Two separate models are trained: one for class identification and the other for association identification. These models are tested using the user story dataset. The results from the ML approach will be compared with those from the LLM approach to determine the better method for accurately deriving domain models from user stories.

6. Analysis Procedure

Comparing Machine Learning Models

We will compare various machine learning models (e.g., Random Forest, SVM, Gradient Boosting) using Leave-One-Out Cross-Validation (LOOCV) for training and testing. The evaluation metrics include F1-score, alongside precision and recall specifically for class and association identification.

Comparing Large Language Models (LLMs)

To compare LLMs like GPT, standardized prompts are developed and applied across the same dataset. The metrics include F-scores, along with additional factors such as consistency (across multiple runs).

7. Teamwork Plan

The project tasks are divided among team members, with Adithiyan leading the initial setup and analysis as well as the implementation of large language models, while Aditya heads the machine learning model implementation and report writing. Vani will lead the evaluation and integration of results. Team members will assist in preprocessing, data validation, parameter tuning, running experiments, refining approaches, and contributing insights for the report sections related to their respective tasks.

References

[1] Bragilovski, Maxim, et al. "Deriving Domain Models From User Stories: Human vs. Machines." *2024 IEEE 32nd International Requirements Engineering Conference (RE)*. IEEE, 2024.
[2] Bragilovski, Maxim, Fabiano Dalpiaz, and Arnon Sturm. "From user stories to domain models: Recommending relationships between entities." *CEUR Workshop Proceedings*. Vol. 3378. CEUR-WS, 2023.
[3] Bragilovski, M., van Can, A. T., Dalpiaz, F., & Sturm, A. (2024). Material of the paper "Deriving Domain Models from User Stories: Human vs. Machines" [Data set]. Zenodo. https://doi.org/10.5281/zenodo.12740518