

Report of Task-2

Paper Title: Automated Extraction of Conceptual Models from User Stories via NLP

Summary:

1.1 Motivation:

The paper highlights the predominant use of natural language (NL) in representing software requirements and the challenge of not readily visualizing key concepts and relationships in NL. The motivation is to propose an automated approach to derive conceptual models from user stories, a concise and widely adopted NL notation for requirements.

1.2 Contribution:

The paper introduces an algorithm implemented in the "Visual Narrator" tool, aiming to automatically derive conceptual models from user stories. The tool combines state-of-the-art heuristics to enhance accuracy in model generation and represents the conceptual models as OWL ontologies.

1.3 Methodology:

The algorithm utilizes 23 natural language processing (NLP) heuristics, classified into concepts, non-hierarchical relationships, hierarchical relationships, attributes, and cardinality. The Visual Narrator tool is designed to fully automate the process, minimizing the need for human intervention.

1.4 Conclusion:

The paper reports promising precision and recall results (between 80% and 92%) from evaluations on two case studies. The generated models are perceived as useful by lead analysts for communicating and discussing requirements, particularly for team members unfamiliar with the project.

Limitations:

2.1 First Limitation:

The paper acknowledges that certain heuristics, such as the distinction between gerunds and present participles, may pose challenges for state-of-the-art part-of-speech taggers, potentially affecting accuracy.

2.2 Second Limitation:

The limitations of the Visual Narrator tool are not extensively discussed in terms of its scalability to handle larger and more complex user stories or its adaptability to different NL notations.

Synthesis:

The paper presents a novel approach to automated conceptual model extraction from user stories, addressing the inherent limitations of existing tools. The Visual Narrator tool, with its emphasis on automation and high accuracy, offers potential benefits for software engineering. However, the identified limitations suggest avenues for future research, focusing on refining heuristics and addressing scalability challenges for broader applicability.