# Toward Automatic Generation of Goal Models using Natural Language Processing

Chenyu Zhang
Mentor: Alicia M. Grubb
Supervisor: Marsha Chechik

May 22, 2019

## 1 INTRODUCTION

Goal modeling has been extensively studied in the literature, with specialized approaches for different domains [1][2]. Some have questioned the value of goal modeling due to insufficient adoption and engagement in research studies among practitioners [3]. Grubb proposed a series of studies to measure the benefits of goal modeling tasks and utility of the overall approach to address barriers to adoption [4]. In this study, we want to experimentally compare the benefits of automatically generating goal models vs. manually drawing them. Through the lens of Berry et al.'s categorization of Natural Language Processing (NLP) usage in requirements engineering, we focus on *model generation* [5].

We are not the first to use NLP techniques in requirements analysis and model generation [6]. NLP has been used to assess the quality of natural language requirements [7, 8, 9]. In contrast to these approaches, we use NLP to automatically generate goal models from natural language requirement descriptions.

## 2 BACKGROUND

In this study, we use iStar 2.0, a modelling language that is commonly used in goal modelling. We briefly introduce syntax of an iStar 2.0 model using the Network Admin model in Fig. 1 (see [10] for a full presentation). This model has two actors, the Network Admin, and the CFO. The model consists of four different kinds of intentions: *goals* (e.g., Improve Network Infrastructure),

*tasks* (e.g., Maintain Network), *resources* (e.g., Network Expertise), and *qualities* (e.g., Have Reliable Network).

These intentions are connected by links. For example, goals and tasks can be refined with *AND-Refinement* or *OR-Refinement*. All elements can contribute to qualities via *Contribution Links*, while qualities can qualify tasks, goals, and resources via *Qualification Links*. Finally, resources can be *NeededBy* tasks. Intentions within an actor can have *Dependencies* with other actors or their intentions.

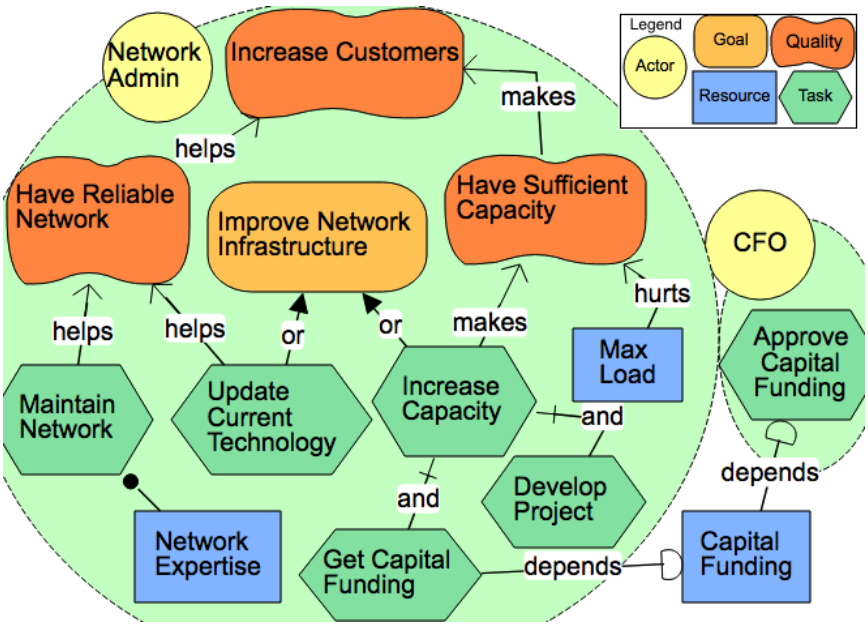


Figure 1: Fragment of Network Admin model, adapted from [11].

# 3 APPROACH

Fig. 2 illustrates our process for generating iStar 2.0 models. In Phase-1, the stakeholders begin by providing an Natural Language (NL) requirement description (i.e., corpus). Each sentence in the corpus is then parsed into noun phrases (NP) and verb phrases (VP), creating parse trees representing the syntactic structure of the corpus. We then extract the intentions in the model and apply the matching rules to find links in the model, compliant with the iStar 2.0 meta-model. Currently, the output of Phase-1 is a DOT language script [12] representation of the model. In Phase-2 (see Fig. 2), we

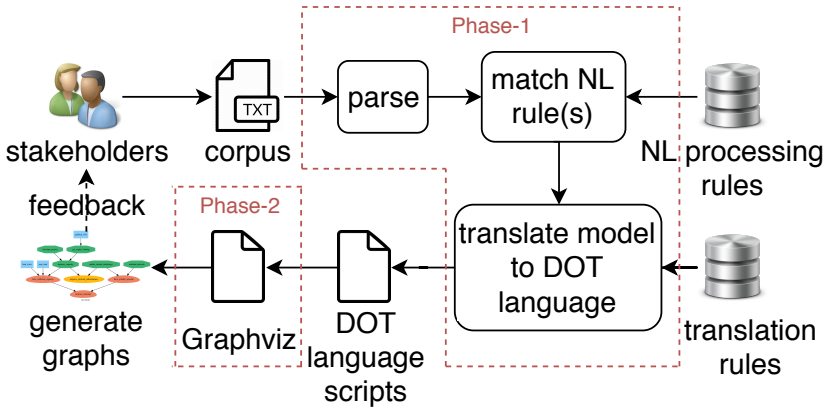


Figure 2: Overview of Model Generation Process.

visualize the model using Graphviz[1] to interpret DOT files and automatically render an iStar 2.0 model. We made three simplifying assumptions in our current design:

1. **Intention Name & Type** We assume unique naming of intentions within a model. We explicitly name each intention and specify the intention's type. These intentions are named entities in the model that can be referred to in the remainder of the corpus. Ideally, we would infer from the corpus whether an intention is a goal or a task, rather than require stakeholders to make this explicit.

**2 Limited Link Rules** Relationships between intentions are expressed in the corpus. We simplified dependency relationships by ignoring specific language about dependums and requiring both depender and dependee elements.

3. **Actor Relationships** We assume that all actors have at least one intention; thus, we cannot generate iStar Strategic Dependency diagrams that connect actors directly with dependency relationships [10].

# 4 ANALYSIS

Suppose stakeholders want to generate the goal model in Fig. 1 of a Network Administrator (Network Admin). Typically, in order to create this model, one or more stakeholder would draw the elements in a modeling tool. Consider

[1] https://www.graphviz.org

instead a stakeholder provided the description in Fig. 3. Given this corpus, we were able to successfully generate the model in Fig. 4 for the Network Admin.

> Have reliable network, increase customers, have sufficient capacity are qualities belonging to network admin. // Improve network infrastructure is a goal belonging to network admin. // Network expertise, max load are resources belonging to network admin. // Maintain network, update current technology, increase capacity, develop project, get capital funding are tasks belonging to network admin. // Approve capital funding is a task belonging to CFO. // Capital funding is a resource. // Improve network infrastructure is decomposed by update current technology or increase capacity. // Maintain network helps have reliable network. // Update current technology helps have reliable network. // Increase capacity makes have sufficient capacity. // Have reliable network helps increase customers. // Have sufficient capacity makes increase customers. // Increase capacity is decomposed into develop project and get capital funding. // Network expertise is needed by maintain network. // Get capital funding depends on capital funding. // Capital funding depends on approve capital funding. // Max load hurts have sufficient capacity.

Figure 3: Initial Corpus for Network Admin Example. "//" Indicates Line Breaks.

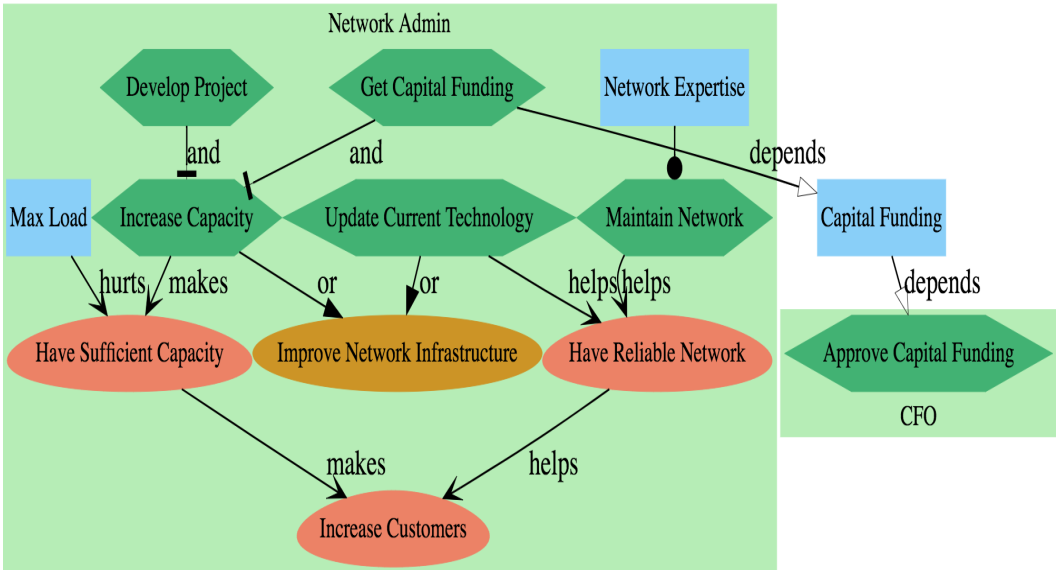


Figure 4: Generated Model of Network Admin Example.

# 5 CONCLUSION

In this project, we proposed to automatically generate iStar 2.0 goal models from natural language requirements documents, and created a prototype implementation. This process consists of two independent steps: (Phase-1) identifying and extracting model information that conforms to the iStar 2.0 meta-model, and (Phase-2) automatically visualizing the model (i.e., auto-layout). We primarily focus on Phase-1 and present a set of simplifying assumptions.

# 6 FUTURE WORK

As mentioned above, we incorporate three simplifying assumptions. Work is ongoing to relax each of the simplifying assumptions made in this proposal, enabling us to produce models with a greater variety of stakeholder input. Future work will investigate using Leaf2.0[2] instead of Graphviz in Phase-2 for visualizing our generated models.

# References


[1] J. Horkoff, T. Li, F. Li, M. Salnitri, E. Cardoso, P. Giorgini, J. Mylopoulos, and J. Pimentel, "Taking goal models downstream: A systematic roadmap," in *Proc. of RCIS'14*, 2014, pp. 1–12.

[2] J. Horkoff, F. B. Aydemir, E. Cardoso, T. Li, A. Maté, E. Paja, M. Salnitri, J. Mylopoulos, and P. Giorgini, "Goal-Oriented Requirements Engineering: A Systematic Literature Map," in *Proc. of RE'16*, 2016.

[3] A. Mavin, P. Wilkinson, S. Teufl, H. Femmer, J. Eckhardt, and J. Mund, "Does Goal-Oriented Requirements Engineering Achieve Its Goal?" in *Proc. of RE'17*, 2017, pp. 174–183.

[4] A. M. Grubb, "Reflection on Evolutionary Decision Making with Goal Modeling via Empirical Studies," in *Proc. of RE'18*, 2018.

[5] D. Berry, R. Gacitua, P. Sawyer, and S. F. Tjong, "The Case for Dumb Requirements Engineering Tools," in *Proc. of REFSQ'12*, 2012.


[2] https://github.com/amgrubb/Leaf2.0

[6] H. Meth, M. Brhel, and A. Maedche, "The State of the Art in Automated Requirements Elicitation," *Information and Software Technology*, vol. 55, no. 10, pp. 1695 – 1709, 2013.

[7] F. Fabbrini, M. Fusani, S. Gnesi, and G. Lami, "The Linguistic Approach to the Natural Language Requirements Quality: Benefit of the Use of an Automatic Tool," in *Proc. of SEW'01*, 2001, pp. 97–105.

[8] D. Popescu, S. Rugaber, N. Medvidovic, and D. M. Berry, "Reducing Ambiguities in Requirements Specifications Via Automatically Created Object-Oriented Models," in *Innovations for Requirement Analysis. From Stakeholders' Needs to Formal Designs*, 2008, pp. 103–124.

[9] M. Landhäußer, S. J. Körner, and W. F. Tichy, "From Requirements to UML Models and Back: How Automatic Processing of Text Can Support Requirements Engineering," *Software Quality Journal*, vol. 22, no. 1, pp. 121–149, 2014.

[10] F. Dalpiaz, X. Franch, and J. Horkoff, "iStar 2.0 Language Guide," *arXiv:1605.07767*, 2016.

[11] A. M. Grubb and M. Chechik, "Modeling and reasoning with changing intentions: An experiment," in *Proc. of RE'17*, 2017.

[12] E. Gansner, E. Koutsofios, and S. North, "Drawing Graphs with Dot," January 2015, available at https://www.graphviz.org/pdf/dotguide.pdf.