Imperial College London

Individual Project

Department of Computing

Imperial College of Science, Technology and Medicine

Using Answer Set Grammars For Text Summarization

Author:
Julien Amblard

Supervisor:
Alessandra Russo

Helper:
David Tuckey

Second Marker:
Krysia Broda

ASG Author:
Mark Law

Wednesday 20th May, 2020

Submitted in partial fulfillment of the requirements for the Computing MEng of Imperial College London

Contents

Chapter 1 **Introduction**

- General Problem
  - Specific Problem
  - Objectives

Chapter 2 **Background**

- Learning answer sets -¿ learning ASG
    - Encoder-decoder overview

Chapter 3 **Contributions**

1 Architecture Overview

The main pipeline, which performs story summarisation, is made of three main steps: the Preprocessor, multiple calls to ASG, and finally prost-processing and scoring (as seen in Figure 5.1. A description of each step can be found in the following chapters.

Figure 3.1: Main Pipeline

- Overview of architecture
- Lots of interesting examples

Chapter 4 **Preprocessor**

- Sub diagrams
    - Motivate steps (Preprocessor make job easier for ASG, making better quality summary)

Chapter 5 **ASG**

1 Overview

Our use ASG is two-fold. Firstly, we pass in each sentence from the story to ASG to obtain its semantic representation in ASP. Secondly, we take these actions and use ASG rules to generate possible summary components. These will later be post-processed and turned into actual valid summaries. A diagram of the two ASG steps is shown below in Figure **??**.

Figure 5.1: ASG Steps

2 Learning Actions

3 Generating Summary Sentences

- Sub diagrams
    - Learning is not really learning (ASG never learns how to summarize, we build in rules of feature extraction)

Chapter 6 **Post-Processing And Scoring**

- Sub diagrams

Chapter 7 **Evaluation**

- Generated dataset
    - NN

Chapter 8 **Literature Review**

- Reread to refer
    - Compare approaches