

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 20<sup>th</sup> May, 2020

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Background</b>	<b>2</b>
<b>3</b>	<b>Contributions</b>	<b>3</b>
1	Architecture Overview . . . . .	3
<b>4</b>	<b>Preprocessor</b>	<b>4</b>
<b>5</b>	<b>ASG</b>	<b>5</b>
<b>6</b>	<b>Post-Processing And Scoring</b>	<b>6</b>
<b>7</b>	<b>Evaluation</b>	<b>7</b>
<b>8</b>	<b>Literature Review</b>	<b>8</b>

# 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 post-processing and scoring (as seen in Figure 3.1). 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**

- 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