Culprit Analytics from Detective Novels

Thesis submitted in partial fulfillment
of the requirements for the degree of

Masters of Science in Computer Science
and
Engineering by Research

by

Aditya Motwani
201401206
aditya.motwani@research.iiit.ac.in

International Institute of Information Technology
Hyderabad - 500 032, India
June 2020
It is certified that the work contained in this thesis, titled “Culprit Analytics from Detective Novels” by Aditya Motwani, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date

Adviser: Prof. Kamalakar Karlapalem
To my family
Acknowledgments

I would like to express my deepest gratitude to my adviser, Prof. Kamalakar Karlapalem for his constant support and encouragement to explore and chart my own research path. He has given me the opportunity to convert to a dual-degree program and has always encouraged me to think out of the box. He has always been prompt in helping me out and has an imbecile work ethic which has been a primary source of inspiration.

I would really like to thank my friends for their constant support and encouragement. My experiences at IIIT would be incomplete without them and I wouldn’t be able to pull off my final year at IIIT if they weren’t dual degree students themselves. Neeraj, Anirudh, Hemant, Shreyash, Sanket, PK, and Swaraj, thank you for being my family away from family. I will always cherish our time spent in college. A special thanks to Shantanu, with whom I had the opportunity to co-author two research papers, for helping me pick up the pieces and learn along the way, for reading and editing my drafts, and for all the fun we’ve had together.

This would be not be possible without my co-author Aayush Naik, with whom I began this research journey with, and who helped me take this thesis to its final conclusion.

My lab mates, Shantanu, Isha, Riya, and Dhruv - thank you for binding together and pushing each other to give our best. This thesis wouldn’t have been possible without the valuable contribution of volunteers who helped me annotate my data and those who participated in the study.

Last but not the least, my family - Ma, Dad and Kunal, who have always been a constant pillar of support and encouraged me to follow my heart and not worry about the road ahead. I wouldn’t be the man I am without your values and guidance.
Abstract

Novels are a mine of information stored in a narrative. Detective novels usually have a premise of a culprit and a detective consolidating the evidence. The key ingredient in a detective novel is obfuscation of the evidence so that the reader cannot easily determine the culprit. We formulate a computational problem here where we identify sentences containing potential evidence, if any, against each character in the novel.

While novels and stories from other genres have a more linear narrative, detective novels reference information from different points in the story. In this genre, the author is motivated to reveal only partial information to a reader to keep them engaged. So first, we propose a method to gather evidence against every character by extracting relevant sentences which seem to implicate them in the crime and are scattered throughout the story. Thus by identifying evidence sentences against each character, we computationally generate an ‘evidence summary’. We split the novel into pre and post culprit expose by identifying the reveal paragraph, and utilise the information post culprit expose to identify semantically similar sentences in their vector representations. We process each novel text as their graph representation, and employ different graph traversal techniques.

Second, from the exercise of extracting evidence we identify and categorise three major ways in which popular authors in crime and detection genre embed evidence in their stories. We compare our algorithm results with human evaluators and qualitatively determine the different writing styles.

Our results show we outperform the baseline method in extracting useful evidence sentence and categorising novels on the basis of clue obfuscation. For one of books in the dataset The Murder of Roger Ackroyd, our method obtains a score of 8/10 compared to a 3/10 baseline. Thus our contributions are two fold - we propose a method to extract evidence containing sentences, in a detective novel utilising the semantics of this genre, and also establish different ways in which authors obfuscate the evidence.
## Contents

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Contributions and Thesis Organisation</td>
<td>4</td>
</tr>
<tr>
<td>2 Related Work</td>
<td>5</td>
</tr>
<tr>
<td>2.1 Summarisation Methods</td>
<td>5</td>
</tr>
<tr>
<td>2.1.1 Search Techniques</td>
<td>7</td>
</tr>
<tr>
<td>2.1.2 Detective Novel Text</td>
<td>7</td>
</tr>
<tr>
<td>2.1.3 Text Segmentation</td>
<td>8</td>
</tr>
<tr>
<td>3 Culprit Determination from evidence</td>
<td>9</td>
</tr>
<tr>
<td>3.1 Introduction</td>
<td>9</td>
</tr>
<tr>
<td>3.2 Terms in Text Analytics of Detective Novels</td>
<td>9</td>
</tr>
<tr>
<td>3.2.1 Pipeline</td>
<td>12</td>
</tr>
<tr>
<td>3.2.1.1 Filtered raw text analysis</td>
<td>12</td>
</tr>
<tr>
<td>3.2.1.2 Vector Representation</td>
<td>13</td>
</tr>
<tr>
<td>3.2.1.3 Graph Representation</td>
<td>13</td>
</tr>
<tr>
<td>3.2.1.4 Sentence Extraction Algorithm</td>
<td>14</td>
</tr>
<tr>
<td>3.2.2 Methodology for culprit evidence collection</td>
<td>14</td>
</tr>
<tr>
<td>3.3 Culprit Frequency Binning</td>
<td>15</td>
</tr>
<tr>
<td>3.3.1 Manipulating text</td>
<td>19</td>
</tr>
<tr>
<td>3.4 Searching for Sentences</td>
<td>20</td>
</tr>
<tr>
<td>3.4.1 Vectorization and Implicit Graph Representation</td>
<td>20</td>
</tr>
<tr>
<td>3.4.1.1 Vectorization</td>
<td>20</td>
</tr>
<tr>
<td>3.4.1.2 Implicit Graph Representation</td>
<td>21</td>
</tr>
<tr>
<td>3.4.2 Graph Search Techniques</td>
<td>22</td>
</tr>
<tr>
<td>3.4.3 Augmented Search Function</td>
<td>22</td>
</tr>
<tr>
<td>3.4.3.1 Search Technique 2 - Weighted Clue Selection</td>
<td>23</td>
</tr>
<tr>
<td>3.4.3.2 Understanding the LDA Algorithm</td>
<td>24</td>
</tr>
<tr>
<td>3.4.3.3 Topic Tilling</td>
<td>26</td>
</tr>
</tbody>
</table>
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Detective novel evidence extraction pipeline.</td>
<td>12</td>
</tr>
<tr>
<td>3.2</td>
<td>A Dependency Tree. Text taken from <em>The Cuckoo Calling by Robert Galbraith</em>.</td>
<td>14</td>
</tr>
<tr>
<td>3.3</td>
<td>Culprit Frequency Binning for two novels. Bin size in the legend is the number of sentences in each bin.</td>
<td>17</td>
</tr>
<tr>
<td>3.4</td>
<td>Culprit Frequency Binning for two novels. Bin size in the legend is the number of sentences in each bin.</td>
<td>18</td>
</tr>
<tr>
<td>3.5</td>
<td>Plate Diagram of LDA.</td>
<td>25</td>
</tr>
<tr>
<td>4.1</td>
<td><em>Murder on the Orient Express</em> culprit binning</td>
<td>40</td>
</tr>
</tbody>
</table>
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Evidence sentences from The ABC Murders [5]</td>
<td>10</td>
</tr>
<tr>
<td>3.2</td>
<td>Name Entity Recognizer Tags of Interest and their Description</td>
<td>11</td>
</tr>
<tr>
<td>3.3</td>
<td>Topic Tiling Mean and Standard Deviation</td>
<td>29</td>
</tr>
<tr>
<td>4.1</td>
<td>A rubric for scoring sentences by degree of relevance, corresponding to quality of evidence</td>
<td>32</td>
</tr>
<tr>
<td>4.2</td>
<td>Evidence containing sentences of The Murder of Roger Ackroyd [4] by Agatha Christie (last 20 lines). Sentences in bold represent some crucial information about the crime.</td>
<td>34</td>
</tr>
<tr>
<td>4.3</td>
<td>Evidence containing sentences of Then There were None [6] by Agatha Cristie (last 30 lines). Sentences in bold represent some crucial information about the crime.</td>
<td>36</td>
</tr>
<tr>
<td>4.4</td>
<td>Examples of evidence sentences before and after reveal</td>
<td>38</td>
</tr>
<tr>
<td>5.1</td>
<td>Comparison of baseline vs our solution over 20 novels. CI is the percentage of readers who found the summaries useful in finding the right culprit. NCI is the percentage of readers who found the summaries as leading to a non-culprit character. NA is the percentage of readers who found the summary as not informative in labelling any character as the culprit.</td>
<td>43</td>
</tr>
<tr>
<td>A.1</td>
<td>Evidence Summaries picked from The Murder of Roger Ackroyd by Agatha Christie by one of the participants. The participant was presented the entire book with corresponding line numbers.</td>
<td>47</td>
</tr>
<tr>
<td>A.2</td>
<td>A sample result from Murder of Roger Ackroyd by Agatha Christie. The user was was to assign a score to the summary based on the rubric in table A.3</td>
<td>49</td>
</tr>
<tr>
<td>A.3</td>
<td>A rubric for scoring sentences by degree of relevance, corresponding to quality of evidence.</td>
<td>49</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

In the age of information, there is an explosion of publicly available data on the INTERNET. A good chunk of this data is text. Unstructured text is one of the largest human generated data source, holding a vast amount of untapped knowledge. The challenge is that of understanding this data, and reducing the complexity of large textual data sources. With the increase in data, we need to develop more efficient systems to curate and process large text documents to extract knowledge. Textual data, owing to how it is generated, has various caveats which are easily navigated by humans but creates difficulties for machines to comprehend and to mine information. For textual data to be a useful knowledge source, we need to teach machines how to navigate these caveats in text, utilising a largely untapped source of available data.

Books represent one of the oldest forms of written communication and have been used since hundreds of years as a means to store and transmit information. In contrast, a large fraction of electronic documents available online and elsewhere comprise of short texts. Webpages, news articles, scientific reports are some examples of documents which have been the focus of natural language processing algorithms. In this thesis, we aim to mine information from primarily long text documents like detective novels. Novels are differently structured as compared to other well-structured documents such as new articles. News articles show a number of characteristics that make it easy to identify some of the key passages without having to perform an in-depth semantic analysis of the text. Some of these characteristics are known locations of key items in a documents, cue words and template-like structure in the case of scientific papers. Such is not the case for literature. They lack a structure or a template of writing, which is what makes every novel different and interesting. Apart from the absence of a structure, there are various linguistic challenges: the use of metaphors, frequent use of dialogue, leaving things unsaid, text possibly having various interpretations, taping the readers skill of reading between the lines and many aspects which makes navigating literature books a interesting challenge.
One cannot expect to find ideas that have key points of the story as a summary, and even less to find it written in the same location across novels.

Novels are a mine of information stored in a narrative. A detective novel usually have a premise of a culprit and a detective consolidating the evidence. The key ingredient in a detective novel is obfuscation of the evidence so that the reader cannot easily determine the culprit. While novels and stories of other genres may have a more linear narrative, detective novels reference information from different points in the story. This presents us with a unique challenge of studying a genre where the author is motivated to not reveal accurate information to the reader and keep them engaged in the process. To perform any culprit analytics, we too need to act like a detective and search through the novel for evidence. Thus we formulate a problem of extraction of evidence against potential culprits. A novel has a narrative form, which means that every sentence in the novel does not hold the evidence against a culprit. There are specific sentences which we need to identify to conclude that a character is the perpetrator of the crime. The closest problem in natural language processing is the extraction of important sentences from long text. This extraction will lead to a shorter version or summary of the original long text with a focus on the evidence against a character. Thus we coin the term “evidence summary”, which is an extractive summarisation of the novel with an emphasis on sentences which provides information on the possible crime committed by the character. With this evidence summary, we can evaluate if a character is the real perpetrator of the crime and find if the evidence against him is sufficient to implicate the character. This gives us insights into how the evidence is woven into the narrative structure by the author. We can compare the different writing styles of crime fiction authors, and evaluate if it was possible for a reader to truly identify the culprit before he/she was eventually revealed.

Most of the text summarization research carried out to date has been concerned with the summarization of short documents [16] (e.g., news stories, technical reports), and in comparison, little work has been done on the summarization of very long documents [12]. Our work seeks to provide empirical evidence on aspects about detective crime and fiction, most of which have been studied qualitatively so far.

To study it quantitatively, we require a system which tackles caveats with respect to large unstructured data and absence of supervised training sets. We develop a novel processing framework which tries to dissect the novel into useful knowledge. Our framework breaks down textual input into multiple graph structures (a system of nodes and edges). Every graph structure has specific aim(s) - to extract evidence sentences, to identify relationships between entities and to capture semantic similarities.

Our study is a comparative study of methods, which aims to understand the problem domain and focuses on the uniqueness of detective novel text. The choice of using detective novels for
this study stems from trying to solve a puzzle in a long text documents. Clues (or evidence sentences), which are hidden in a complex narrative, are the flagposts that we aim to identify. The narrative structure which varies from book to book, requires critical reading which involves a deeper understanding of the text, and is very time-consuming. To streamline the process, we aim to create a pipeline which given a novel, or a set of novels, filters out the clues. Our framework performs pre-processing, text analytics and then post-processing. The focus is to provide an end-to-end solution, while trying various algorithms which fits the purpose of extracting evidence. It is a contribution of this thesis that extracting clues from detective fiction novels is feasible by employing tools in natural language processing, without the use of deep semantic understanding of text or directly using knowledge pools. By using semantics and observing patterns over novels, we produce evidence based summaries of text that would be helpful in identifying the major plot-twists.

For the purpose of our experiments we build a corpus of detective novels that primarily revolve around a singular culprit who is involved in a crime like robbery or murder. To incorporate various writing styles, we pick popular authors over the last century. Furthermore, we include both male and female authors spanning various nationalities in the popular crime genre to ensure diversity in the corpus. Some of the prominent authors in the corpus that we analyze are Arthur Conan Doyle, Agatha Christie, Stieg Larsson, J.K. Rowling, Dan Brown and Lee Child. We also pick novels of varying lengths.

The challenge of our task is to analyze novels which span around 30,000 words and are multi-document texts. Additionally, authors have widely varying writing styles. The classical approaches in Information Extraction work well for documents, which are about 500-700 words each [12]. It is difficult for standard Information Extraction (IE) based techniques to extract important sentences in novels across multiple sentences and authors. A standard approach for this task would be to train state-of-the-art deep networks and learn what constitutes as valid evidence. To train deep networks for such information extraction, we require a large training corpus of annotated data. But in the domain of detective novels, there is no existing annotated data of evidence sentences and no fixed method of evaluation. Moreover, making such a corpus is not as simple as just labelling a sentence as evidence or not evidence by just a single check of the sentence. One needs to have the relevant context from the novel and a good understanding of what piece of information could be considered as evidence. Thus it is difficult to extract data in a supervised way. To overcome this problem, we propose an unsupervised method - Augmented A* search to extract evidence sentences. Our technique exploits the underlying semantic structure in a novel to overcome the challenge of manual annotation of sentences. Post this, we evaluate the quality of our gathered evidence sentences from human evaluators.
Then we base our analysis of the detective novels on these extracted evidence sentences and study the different narration styles.

1.1 Contributions and Thesis Organisation

To sum up, our contributions are two fold - we propose an unsupervised method to extract evidence containing sentences in a detective novel. Second, we utilise this information to establish and classify different ways in which authors obfuscate evidence in their novels and extend work on theories in crime and detective fiction.

The organisation of thesis is as follows. In the next chapter, we discuss the literature surrounding our problem and how we aim to situate our research within the domain. In chapter 3, we discuss how we model our problem to extract evidence. In chapter 4, we discuss the results of our methods and discuss how the writing styles of various detective novel writer and present our result in chapter 5. We conclude in chapter 6.
Chapter 2

Related Work

We situate our research on unique challenges in summarisation for different types of texts and specifically - detective novel text, different methods for information extraction, summarisation techniques, and detective novels features and analytics.

2.1 Summarisation Methods

Typical text extraction techniques are usually referred in the NLP/IRE domain as summarisation. They are broadly categorised as extractive and abstractive summarisation methods. While extractive techniques pick sentences directly from the document for generating summaries, abstractive methods generate new phrases often encompassing the critical parts and information. These techniques largely prioritise coherence in between sentences, coverage, and diversity of topics for generating effective summaries of the text.

To achieve automatic summarisation, we broadly divide them into two methods: supervised and unsupervised. Supervised algorithms are trained on pre existing text such as document and document summary pairs. In the domain of unsupervised techniques, to rank a sentence for importance, typical methods account for the length of the document as well as the occurrence of the words in the sentences. Unsupervised techniques are built around properties and heuristics that are observed and derived from texts.

A large part of text summarisation work carried out has been on short documents such as newspaper reports and articles. [16]

However, documents of short length typically have one topic of discussion. Thus the techniques developed for short summarisation are not effective for long text summarisation as they do not take into account the varying number of topics throughout the text and the length of the novels. Another important aspect is the evaluation of long text summarisation. In [16],
Mihalcea et al. show that there is a large body of work concerned with the summarisation of short documents, with varying evaluation types focusing on news articles. The authors discuss the steps taken to improve long text summarisation, concluding that the performance of their system is mainly by creating features that take into account the length of the text document. These features include accounting for length, exclusion of positional scores, segment ranking, text segmentation, and a segment-based weighting scheme.

Mihalcea et al. test long text summarisation with different systems. Initially the authors start with MEAD system which is a centroid-based method to weigh the information in a sentence. This method is preferable for three reasons: 1) It has good performance in overall DUC evaluations e.g., (Radev et al., 2003 [19]; Li et al., 2005 [14]). 2) It does not require any training data and is an unsupervised method. The is difficult to create for long documents for mainly two reasons. The task of creating a exhaustive dataset for long documents is difficult, 3) It is centroid based technique that can be modified to include document and sentence length which is key for long text summarisation.

In short text documents, the initial sentences have a greater information bias as compared to the rest of the document. To account for this, we add a positional score that gives a sentence a higher score if it is closer to the beginning of the document. For long text documents, the authors observe that by removing the position score leads to a better performance on ROGUE tests since sentences in the document are equally important as compared to short documents.

Supervised learning techniques are trained over a group of documents and a corresponding set of manually curated validation set of summaries. By aiming to identify the key components of a summary (the presence of key phrases, positional scores, named entities etc) we identify patterns across the collection of documents and develop a system. Wan et al. [24] constructs summaries of long chain email thread by identifying the main topic being discussed. They devise a comparison vector from the replies and perform a cosine similarity with the issue emails and rank these sentences.

In [16], the authors use ROGUE systems [13] as a metric of evaluation. The ROGUE-1 system seeks unigram matches between the reference and generated summaries. Other common systems used are ROGUE-2 (bigram matches) and ROGUE-SU4 (non-contiguous bigrams).

Typical summarisation techniques do not prioritise different kinds of information extraction requirements. There have been attempts to do meaningful information extraction based on different requirements. Graph based text summarization (Thakkar et al [22], Nenkova et al [17]) has been attempted on different domains using different algorithms. Thakkar et al [22] use a similarity metric based on common words in sentences to generate a graph and then use the PageRank algorithm to find important sentences in the domain of web articles. The use of pagerank algorithm by Thakkar does not incorporate the deep linguistic knowledge embedded
in the narrative. [7] works in the domain of scientific article summarization using the article’s discourse structure. They address the problem of evaluation of their summary by extracting citation-context in the reference article of each citation. This gives them a comparable baseline to evaluate their performance.

2.1.1 Search Techniques

Aker et al [1] use A* search with different heuristics for multi-document summarization on an established ground truth. In comparison, our problem does not have an established ground truth or a pre-annotated corpus of sentences for the A* search.

Another technique to generate summaries is by using query-based summarisation methods. Wenpeng et al [26] focus on three main aspects - novelty, query oriented relevance and information richness and treat them as sentence features. This along with the sentence vector help in the re-ranking of sentences for that particular query to generate a query focused summary.

2.1.2 Detective Novel Text

Since our work seeks to empirically analyse the writings in the detective novel genre, we looked at some related work surveying this genre. [20] explores literary theories surrounding detective novels. [20] talks about elements in a detective story such as, culprit-detective relationship, motivations of characters, tracing and retracing of clues. These are some features which outline the detective and crime fiction genre. Stowe, W. outlines which elements work well for such genre authors. He claims the moral grandstanding of the detective sets well with the readers of the genres. These stories keep readers engaged, and provide a definite closure. Throughout the last century, writers have moved from personal reasons as motivations for crime, to larger societal injustices as cause of some of the crimes. The genre has evolved, often incorporating writing in terms of chasing a culprit, typical whodunits and even written to show a mirror to society.

The whodunit genre [25] in detective-crime fiction unfolds any story in two primary ways. Either the crime and related events unfold and happen in real time, or the events of the incident are recalled and revealed over the story. This is described as double narrative, where one follows the present time and the other is slowly revealed as the story progresses. Some narratives move backward and forward chronologically, slowly revealing details of the crime and the ongoing investigation.
2.1.3 Text Segmentation

For long text documents, the paragraphs can be characterised as a series of topic shifts over short sections of text. In [10], Marti describes subtopic segments as multi-paragraph passages, where a combination of subtopic segments are present in the context of one or more main topic discussion. This is applicable for novels as well, where there is demarcation present across the novel as chapter headings. But there are some novels which lack pre-defined structure and for those texts, text segmentation according to topic shift can be a useful option. There are many methods to achieve demarcation, using graph based segmentation cuts (Malioutov and Barzilay, 2006) [15], Text Tiling Algorithms [10] and LDA based Topic Tiling Algorithms. In graph-based segmentation cuts, the document is modelled as a graph connected with sentences as nodes, and cuts are made in the graph to divide the text into segments. The algorithm requires the number of segments to be demarcated as input, which is different from [10]. Marti does not pre-determine the number of segments, as there should be a relation between the structure and style of text to the number of segments. Instead, the number of segments is determined as a function of the depth score of each segment cut, where the average depth and standard deviation of the scores are used to determine if this segment cut should be used as a topic shift boundary.
3.1 Introduction

In this chapter, we model our problem into a sentence extraction problem from text. We discussed some of the trends in detective novels in the previous Section. Our methods focus on building a pipeline which extracts evidence containing sentences. This pipeline takes in raw unfiltered text as input and performs evidence discovery to meet the task. As part of evidence discovery, we illustrate the story as a text graph which contains all the major entities and the relations with other entities. The aim is to help the reader navigate the story through a visual representation, which may help the reader analyse the text efficiently. We describe an overview of the pipeline, followed by the descriptions of each component namely, filtered raw text analysis, vector representation of text, graph representation of text and the search algorithm.

3.2 Terms in Text Analytics of Detective Novels

Entity of Interest

For the reconstruction of a narration/story, we are interested in those individuals or objects which are essential in dictating the sequence of events. We label them as entities of interest. They are helpful as they primarily form the nodes in our graph representation. Entities of interest can be of different types, and we can classify them into categories like person names, organisations, locations, products (objects, vehicles, food), events, quantities, monetary values, date and time. We use named entities recognizers to extract this data from the text. The different categories of entity of interest tags are shown in Table 3.2 below.
We identify such entities and seek to locate and classify named entity mentions in unstructured text into pre-defined categories. For example, in the novel, *The Adventure of the Blue Carbuncle*, Sherlock, Watson, Ryder, Horner, the goose, and many more are all entities of interest.

**Culprit**

The antagonist of the novel. If there is a crime or a misdeed in the novel, then, the primary doer of the deed is the culprit.

**Culprit Sentence**

The first sentence in the novel that makes it unambiguous who the true culprit is. Very often, this sentence is an assertion or deduction by the detective following which the detective explains how he/she arrived at the said deduction. Or sometimes, it is a confession by the culprit.

Table 3.1: Evidence sentences from The ABC Murders [5]

<table>
<thead>
<tr>
<th>Evidence Sentences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Up to the time of the Barnard murder, no facts about the A.B.C. murders had been</td>
</tr>
<tr>
<td>made public.</td>
</tr>
<tr>
<td>It therefore followed that whoever killed Betty Barnard must have had access to</td>
</tr>
<tr>
<td>facts known only to certain persons - myself, the police, and certain relations and</td>
</tr>
<tr>
<td>neighbours of Mrs Ascher.</td>
</tr>
<tr>
<td>Miss Grey here told us that she did not see or speak to any stranger on the day</td>
</tr>
<tr>
<td>that Sir Carmichael Clarke was murdered.</td>
</tr>
<tr>
<td>There was the possibility that the footmarks might have been made by somebody else</td>
</tr>
<tr>
<td>who happened to have the same kind of studs in his shoes.</td>
</tr>
<tr>
<td>According to the police theory, Ralph was wearing another pair of the same kind,</td>
</tr>
<tr>
<td>and I found out that it was true that he had two pairs.</td>
</tr>
</tbody>
</table>

**Evidence**

A sentence which provides information used in the detection of a crime or criminal. Any piece of information that increases the confidence of the reader that a particular character is the culprit.
Table 3.2: Name Entity Recognizer Tags of Interest and their Description.

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date</td>
<td>Absolute, relative dates, periods</td>
</tr>
<tr>
<td>TIME</td>
<td>Times smaller than a day.</td>
</tr>
<tr>
<td>WORK_OF_ART</td>
<td>Titles of Books, Songs</td>
</tr>
<tr>
<td>PRODUCT</td>
<td>Objects, Vehicles, Food</td>
</tr>
<tr>
<td>ORG</td>
<td>Companies, Agencies, Institutes</td>
</tr>
<tr>
<td>GPE</td>
<td>Countries, Cities, States</td>
</tr>
<tr>
<td>FACILITY</td>
<td>Buildings, Airports, Highways, Bridges etc</td>
</tr>
<tr>
<td>NORP</td>
<td>Nationalities or religious or political groups</td>
</tr>
<tr>
<td>PERSON</td>
<td>People, including fictional</td>
</tr>
<tr>
<td>LOC</td>
<td>Non-GPE locations, Mountain Ranges, Bodies of water</td>
</tr>
<tr>
<td>PRODUCT</td>
<td>Objects, vehicles, foods</td>
</tr>
<tr>
<td>EVENT</td>
<td>Named hurricanes, battles, wars, sports events</td>
</tr>
<tr>
<td>LAW</td>
<td>Named documents made into laws</td>
</tr>
<tr>
<td>LANGUAGE</td>
<td>Any named language.</td>
</tr>
<tr>
<td>MONEY</td>
<td>Monetary values, including unit.</td>
</tr>
<tr>
<td>QUANTITY</td>
<td>Measurements, as of weight or distance.</td>
</tr>
</tbody>
</table>
If the evidence points towards the true culprit then, it is true evidence. Else, it is misleading or false evidence. We show some examples of the evidence sentences retrieved by our search algorithm for the novel *The ABC Murders* by *Agatha Christie* in Table 3.1.

### 3.2.1 Pipeline

Our solution consists of a pipeline that takes in unfiltered text. The pre-processing stage removes page numbers and chapter names. We describe our pipeline in Figure 3.1.

For understanding trends in text, we branch our pipeline into two parts - one path uses text in its raw form and the other path uses vectorized form of the text. We describe its components in abstract terms as follows:

#### 3.2.1.1 Filtered raw text analysis:

In block 1 from Figure 3.1, we identify key information such as main characters in the novel and finding the culprit paragraph.

Block 2 represents our culprit frequency binning algorithm from section 3.3. Modern day computers process text as simple sequences of character strings. This plain text file is processed into a more efficient data structure. By using the syntactic structure of text, we identify all key entities. To accomplish this, we use Spacy's language processing library. Spacy first tokenizes the text, assigns POS tags and uses a statistical entity model assigns labels to contiguous parts of text which it identifies as named entities.
In our analysis of trends in detective novels, we map the frequency occurrence of all major characters in a novel. We describe our efforts to use trends across novels to identify the culprit reveal paragraph in Section 3.3.

3.2.1.2 Vector Representation

In automatic text summarisation, sentences that are most relevant to a summary are chosen by our algorithm. This algorithm works by representing each word as a vector in an embedding space. To accomplish this, we use spacy’s word2vec tool to map every word to a 300 dimension vector. For more accurate embeddings, we use spacy’s en_core_lg model which has unique mappings of 1 million words. For our purposes, the smallest unit of computation is a sentence. Hence, we only deal with sentence vectors. Each sentence is assigned a vector which is an average of the word vector it contains. This is generally a good method to compute sentences vectors, but with some limitations [21]. It ignores the order of the words in the sentence, and is akin to bag-of-words representation. To remedy this issue, [21] use a parse tree, and combine the words in the given order. While this is a useful approach for sentences, it does not work well for paragraphs or blocks of text. We use the averaging method for vector representation of sentences for further operations.

A typical problem with sentences in long text is identifying the sentence boundaries. These boundaries can be determined using rule based methods, but are prone to errors depending on the quality of text and writing styles. For more accurate results, we use dependency parse trees to extract sentence boundaries. These parse trees work by analysing the subject and object of a sentence to understand if the current boundary encapsulates all the elements of the sentence. An example of a dependency parse tree is shown in Figure 3.2.

3.2.1.3 Graph Representation

The block 3 of the pipeline concerns with representation of the novel as a set of nodes and edges. Each sentence is represented as a node. The edges between sentences are drawn if they are related, as directed by the algorithm. The relations used in our problem are described in Section 3.4.1.2. This makes the useful information present in the text, usable by several graph algorithms. Thus it is possible to query a manifold of direct and indirect relations. Extracted information from text is converted into a graph representation of sentences and edges between the sentences.

Block 4 of the pipeline represents the extraction of the culprit sentence, which is defined in section 3.2.
3.2.1.4 Sentence Extraction Algorithm

The block 5 from Figure 3.1 represents the sentence extraction function from text. Once a graph representation is made out of text using characters and relations, it becomes usable as an input to several search functions. We describe one such approach in Section 3, where we augment a traditional search algorithm using the vector representation of entities and relations. The output from this block gives us a list of nodes (or sentences) which contain potentially meaningful information and are part of the summary.

3.2.2 Methodology for culprit evidence collection

We divide the task at hand into a composition of two sub-problems.

1. Locating the context in the novel where the culprit is revealed.

2. Extraction of evidence sentences in the novel before the culprit is revealed.

We now describe two algorithms for the same in detail.
3.3 Culprit Frequency Binning

To explore the patterns of "culprit" occurrence in our novel corpus, we begin by devising an algorithm that divides novel into parts depending on culprit mention frequency.

To understand how detective novel writers conceal a culprit from the readers, we begin by mapping the frequency of occurrence of the culprit over the novel. Some of the issues we faced while doing this was the frequent use of varying titles. Over the length of a novel, the same character can be referred to by different names. For example, in the mystery of *The five orange pips*, Sherlock Holmes is referred to as Sherlock, mister Holmes, and Sherlock Holmes. To remove all ambiguities, we consolidate all references to the same character under one name. We map all references to a single character by manually defining filters for common character occurrences, and replacing all the names of a character to the root name. This is done by ranking the common name occurrences, with the help of Named entity recognizers. This helps us maintain uniformity over the novel.

Our aim is to find the part in the novel text where the "culprit" occurrence is unusually high and then investigate it manually. We begin by dividing the text into equal sized batches of sentences, and count culprit frequency by varying the batch sizes. The size of these batches of sentences, referred to as *bin size b*, vary in the range of ten sentences to the average chapter length in the novel (*U*). We do not go beyond the chapter length because that is a good subjective parameter where the author has made their own demarcation in the text.

For a given novel text, we begin with bin size *b* = 10, and go until *U*, which is the average chapter length in the novel. That is, we begin by dividing the text into equal batches of 10 sentences, and vary until *U*. For each bin size *b*, we compute the highest frequency of occurrence of a character *c* from ANY of the bins and note it as *x* from bin size *b*\(^*\). From bin size 10 to *U*, we compute the highest of all the values of *x* and call it as *X*\(^C\), where *C* is any character.

Now, every character *C* in the novel text has an *X*\(^C\) from a bin size *b*\(^*\) where the character frequency is the highest. As hypothesized, when *C* = culprit, our empirical results show the frequency of culprit that comes from any of the bins is higher than that for any other characters.

For characters who are eventually revealed to be the culprit, we observe a common trend across detective novels. We see that just after a culprit is revealed, the frequency of occurrence of the culprit increases sharply. This is highlighted in the figures in 3.3 and 3.4 This can be attributed to events like the detective explaining how he/she deduced who the culprit is, or, the culprit himself/herself confessing.

With the culprit caught, there follows an explanation of the crime. This is the juncture in the novel where the two narratives from the detective and killers perspective merge. In the context of evidence sentence analytics, the sentences that follow are all ground truth facts for why a
Algorithm 1 Culprit Bucket Optimization.

1: function BUCKET(Sentence[] doc, Integer min_bucket_size, Integer max_bucket_size)
2:   bucket_size ⇐ min_bucket_size
3:   optimal_bucket_size ⇐ min_bucket_size
4:   Sentence[] culprit_bucket ⇐ 0
5:   max_mentions ⇐ 0
6:   while bucket_size ≤ max_bucket_size do
7:     Sentence[] Buckets = emptyset()
8:     for i in range(number of buckets) do
9:       Sentence[] bucket = Buckets[bucket_size * itobucket_size * (i + 1)]
10:   end for
11:   ▶ Divide the sentences into buckets of size bucket_size
12:   ▶ Count the number of times the culprit is mentioned
13:   ▶ Let mentions and buckets hold the values for the most mentions among all buckets and the bucket with most mentions
14:   if mentions > max_mentions then
15:     max_mentions = mentions
16:     optimal_bucket_size = bucket_size
17:     culprit_bucket = bucket
18:   bucket_size = bucket_size + 1 return culprit_bucket, optimal_bucket_size = 0
Figure 3.3: Culprit Frequency Binning for two novels. *Bin size* in the legend is the number of sentences in each bin.

(a) *Dr Sheppard* in *Murder of Roger Ackroyd*

(b) *Mr Stapleton* in *The Hound of Baskerville*
Figure 3.4: Culprit Frequency Binning for two novels. *Bin size* in the legend is the number of sentences in each bin.
character is a culprit. The sentences classified before this juncture come under the purview of evidence.

Novels as a dataset have high variation in their writing styles. In 17/20 of the novels in our dataset, this optimal bin with the highest culprit frequency occurs in the second half of our novels, while in novels such as \textit{The Lost Symbol}, \textit{Eye of the Needle} and \textit{The Girl with the Dragon Tattoo}, this optimal bin happens to occur in the first half. This implies that the part in the text where there were repeated mentions of the culprit was in the second half for seventeen novels and the first half for the other three. We observe that in these 3 cases there is another bin with an approximately similar frequency which occurs in the second half of the novels. Since mostly the culprit is not revealed before the first half, we pick this other bin in the second half as the culprit frequency is approximately the same.

### 3.3.1 Manipulating text

Once we have identified the part of the text where the culprit mention is relatively higher, we work backwards from this point. Our hypothesis is that working backwards from the point of revelation, we can gather sentences which relate to it by providing information about the culprit. These sentences potentially connect to sentence that lead to the "big revelation" in the end.

While doing this is possible for character who were culprits because there will be a culprit revealing sentence, we cannot do this for other characters since they were never announced as culprits. However as our task is to identify sentences which link to evidence related information for the given character and our method is to work backwards from the point of big reveal, we modify the novel by adding a sentence which implicates a certain character as the culprit. At the beginning of the culprit paragraph, we add an asserting sentence which implicates every character in our set. For example, in the Murder of Roger Ackroyd by Agatha Cristie, we add the sentence “Ursala Borne killed Roger Ackroyd.” as Ursala Borne was not the culprit. This helps us maintain uniformity of the culprit sentence. That is, even if the text lacks an explicit implication of the culprit or any other character, we can add such an implication at the beginning. This is then used to extract evidence against all the characters. Additionally, this maintains consistency of the culprit sentence across novels. If no evidence against a particular character is found, then, we can state that character was not the culprit with high certainty.

Another variation of this algorithm is dividing the text into bins and applying PageRank [18] on each of the bins. If a particular bin has a high PageRank for the culprit as compared rest of the bins, then, this bin can be seen as analogous to the maximum culprit mention frequency bin in the regular culprit frequency binning procedure.
On testing this approach, we find that there was no significant improvement with this approach. The bins given after applying pagerank lie in the same neighbourhood of sentences (on an average of 3-4 sentences) without pagerank, giving us the culprit reveal paragraph. Hence going forward, we use our original method without pagerank.

### 3.4 Searching for Sentences

We describe a technique which works on the information provided by the binning algorithm. This algorithm is an augmented version of the A* search ([9]) algorithm. The A* Search algorithm is a path finding algorithm which tries to find the lowest path cost to the goal node. We instead, use a variation of A* Search which optimises the maximum path sum, as it solves our problem of finding relevant sentences. In our problem space, the nodes are the sentences of a novel represented in vector form. We define the edges between these nodes in Section 3.4.1.2. The aim is to find an optimal path, which is to increase the information density by picking evidence sentences, but discard sentences which have repeated information and sentences which are not useful in detecting the culprit. Our optimal path starts at the first sentence of the novel and ends at the culprit-revealing sentence.

#### 3.4.1 Vectorization and Implicit Graph Representation

In this Section, we describe the different components that we use for the extraction and representation of the sentences.

We convert each word into a word vector which is the projection of our word in a word space, as this helps us numerically represent our sentences. By doing this, we are able to cluster words with similar semantic meaning closer than over words which are not semantically related. This aids in the formation of sentence vectors which use the word vectors to create a single vector to represent a sentence. Next, we create edges between sentences in the story to form a collection of nodes (sentences) and edges. This representation of the text is the implicit graph representation. We call it implicit, as the graph is not represented as explicit objects in the system’s memory, but instead determined algorithmically at every run-time.

##### 3.4.1.1 Vectorization

We use spaCy’s en_core_web_lg model, which is an, ”English multi-task CNN trained on OntoNotes, with GloVe vectors trained on Common Crawl. Assigns word vectors, context-specific token vectors, POS tags, dependency parse and named entities” ([11]). This gives
every word a 300 dimension representation. The Euclidean distance (or cosine similarity) between two word vectors provides an effective method for measuring the linguistic or semantic similarity of the corresponding words. Using these vectors, we are able to employ the underlying semantics to extract useful information. SpaCy establishes similarity between sentences by creating sentence vectors which are an average of the word vectors in that sentence. These sentence vectors are the node representation in our graph.

### 3.4.1.2 Implicit Graph Representation

Every vectorized sentence is a node. For the \( i \)th and the \( j \)th sentences, with sentence vectors, \( v_i \) and \( v_j \) we define,

\[
    r_{i,j} = f(v_i, v_j) = \alpha C(v_i, v_j) + \beta C(v_j, v_n)
\]  

(3.1)

where \( \alpha \) and \( \beta \) are weights assigned to each of the similarity measures which we refer to as hyperparameters.

\[
    \alpha, \beta \in [0, 1]; \alpha + \beta = 1
\]  

(3.2)

\( C(v_a, v_b) \) gives the cosine similarity between the sentence vectors \( v_a \) and \( v_b \), and \( v_n \) is the sentence vector of the culprit sentence (or the \( n \)th sentence).

\( r_{i,j} \) captures how similar the two sentence vectors, \( v_i \) and \( v_j \) are, and also provides an heuristic measure on how relevant the \( j \)th sentence is to the final summary based on how similar the vector is to that of the culprit sentence.

By varying \( \alpha \) and \( \beta \), we empirically find \( \alpha = 0.6 \) and \( \beta = 0.4 \) as a good solution for our problem.

There is an edge between two sentence vectors \( v_i \) and \( v_j \) if their corresponding sentences are less than or equal to \( \tau_n \) sentences apart. \( \tau_n \) is set according to the size of the novel — larger for longer novels, that is,

\[
    | i - j | \leq \tau_n
\]  

(3.3)

and \( r_{i,j} \) is greater than a certain threshold \( \tau_c \). In other words,

\[
    r_{i,j} > \tau_c
\]  

(3.4)

\( \tau_c \) precludes formation of an edge between sentence vectors \( v_i \) and \( v_j \), if \( r_{i,j} \) is not sufficiently large. \( \tau_c \) thus governs how connected the graph is and plays a role in determining the length of the extracted evidence summary.
3.4.2 Graph Search Techniques

In this Section, we make use of our implicit graph representation to extract sentences. This is analogous to block 6 and 7 in fig 3.1. In the graph, the sentences are represented by the nodes and the edges represent a path between two nodes. The text is represented as a directed graph where the first sentence is the first sentence and the last node is the culprit reveal sentence. Using the graph representation of our novel, we traverse the sentences from one node to a neighbouring sentence, which is selected by the search function. We descend from the first sentence to the culprit reveal sentence while picking culprit sentences in between. The first sentence is set initially as the current sentence.

We define two sets $\text{frontier}$ and $\text{visited}$ which are set to null. $\text{Frontier}$ stores the neighbouring sentences of the current sentence and $\text{visited}$ stores all the sentences which have been selected by the strategy function. We move down the graph till the current sentence does not reach the goal sentence. Since we are performing the traversal for every character, the goal sentence changes with each iteration. We set the goal sentence to that line which implicates the character in that crime. Using graph search strategy functions in subsection 3.4.3, we update the $\text{next}$ variable and the state variables.

3.4.3 Augmented Search Function

If the current sentence is the $i^{th}$ sentence or $v_i$, the $(i+1)^{th}$ sentence chosen by the algorithm is the one for which there exists an edge between $v_i$ and $v_{i+1}$ and which maximizes

$$\gamma C(v_{i+1}, v_n) + \delta D[v_{i+1}]$$

(3.5)

where $C(v_{i+1}, v_n)$ is the cosine similarity between $v_{i+1}$ and $v_n$, and, $D$ represents the distance traversed till now. We initially set $\gamma$ and $\delta$ to 0.5.

$$\gamma, \delta \in (0, 1]; \gamma + \delta = 1$$

(3.6)

We perform nine variations of the weights $\gamma$ and $\delta$ at intervals of 0.1. We observe that on increasing the granularity any further, we do not get increased variation in results. This gives us tuple pairs like (0.1,0.9), (0.2,0.8) and so on. We empirically select the variation which optimises the occurrence of the culprit character in the summary, to gather maximum information around the character. $D(v_i)$ gives the sum of cosine similarities between nodes from the start node along the path, until (inclusive) the $i^{th}$ node.

In the formula above, $C(v_{i+1}, v_n)$ is a measure of the similarity between the next contender sentence to the culprit sentence. It is a heuristic measure of understanding the importance of
key words in the sentence. $D[v_{i+1}]$ is the total distance cost from the first sentence to the next contender sentence.

**Algorithm 2 Graph Traversal Function**

1: **function** SEARCH(SV doc, SV start, SV goal, Function strategy)  

   * Sentence Vector*

2:     current $\leftarrow$ start
3:     frontier $\leftarrow$ $\emptyset$
4:     visited $\leftarrow$ $\emptyset$
5:     **while** current $\neq$ goal **do**
6:         **yield** current
7:         visited $\leftarrow$ visited $\cup$ {current}
8:         new_nodes $\leftarrow$ neighbors(current) $-$ visited $-$ frontier
9:         frontier $\leftarrow$ frontier $\cup$ new_nodes
10:        next, $\mathcal{D}$ $\leftarrow$ AStarstrategy(doc, current, frontier, goal, $\mathcal{D}$)
11:        current $\leftarrow$ next
12:        frontier $=$ frontier $-$ {current}
13:     **end while**
14: **end function**=0

3.4.3.1 Search Technique 2 - Weighted Clue Selection

We propose another search technique to select evidence sentences, given the graph search algorithm. We change the way we measure how two sentences are connected in the graph. Instead of having a measure of how a sentence compares to an evidence sentence, we change our metric to reflect how well two sentences are similar to each other.

In this approach, we rethink how to create a connected graph by dividing the text more systematically. In the previous approach, we divide the text with a pre-defined number to create uniformity in splitting the text at paragraph boundaries. In this approach, we divide the text by identifying the topic of each sentence. We use an unsupervised method for topical segmentation of text. We represent text as a sequence of semantically coherent segments using the Bayesian topic modeling approach. In the subsection below, we define the two algorithms...
Algorithm 3 Augmented Search Function

1: function ASTAR([SV] doc, SV current, set(SV) frontier, SV goal, Map D)
2:     if D is empty then
3:         for all SV v in doc do D[v] ⇐ −∞
4:     end for
5:     D[current] ⇐ 0
6:     end if
7:     for all SV v ∈ neighbors(current) do
8:         D[v] ⇐ max(D[v], D[current] + C(current, v))
9:     end for
10:     next ⇐ argmax_v γC(v, goal) + δD[v]
11: return next, D
12: end function

used to generate the connectivity of a story graph - 1) LDA Algorithm and 2) Topic Tilling Algorithm.

3.4.3.2 Understanding the LDA Algorithm

LDA in text processing is an application of the Bayesian approach, namely topic modeling (Blei et al., 2003 [3]; Alpay-din, 2014 [2]). It serves its purpose for our method as it outputs the probability distribution over topics for each sentence of a given text. It does so in the following manner.

LDA imagines a fixed set of topics. Each topic represents a set of words. And the goal of LDA is to map all the documents to the topics in a way, such that the words in each document are mostly captured by those imaginary topics. The probabilistic topic model estimated by LDA consists of two tables (matrices). The first Table describes the probability or chance of selecting a particular part when sampling a particular topic (category).

LDA is parametric model and its size is fixed, but we can make this model non-parametric by making the number of topics increase as necessary and adapt them to data using a Dirichlet process (Alpaydin, 2014 [2]). However, in our project we did not automatically adjust the number of topics to data. We have chosen several fixed numbers of topics and evaluated our method against them to see which one gives the most satisfying results.

LDA model in general works in the following way:
1. Generate topics in advance.
2. Assign a topic to each word.
3. Check up and update topic assignments iteratively.
4. Output the topic probabilities for the document

In the plate diagram in Fig 3.5,

- $w_w$: represents a document (i.e. vector of ws) of N words
- D: Number of documents
- $z_w$: A topic from a set of k topics. A topic is a distribution words.

The algorithm is given below:

**Algorithm 4 LDA Algorithm**

1. function LDA ALGORITHM(Corpus D)
2. for all document $d_d$ in corpus $D$ do
3. 
4. for all position $w$ in $d_d$ do
5. 
6. Choose a word $w_w$ from $p(w_w|z_w, \beta)$, a multinomial distribution over words conditioned on the topic and the prior $\beta$.
7. end for
8. end for
9. end function
3.4.3.3 Topic Tiling

With the aim of being able to segment the given textual document into semantically coherent units, we apply the Topic Tiling algorithm to it.

The purpose of topic tiling is to segment the document into smaller parts which have a coherent topic. This ensures that the summary has good coverage and an overview of all the relevant topics from the document. This is mostly useful for books which do not have chapter demarcations over the novel, which makes it difficult to carry out the selection of relevant evidence sentences over the novel.

In the previous Section we have discussed LDA model and how it outputs probability distribution over topics for each sentence of a given text. We refer to this distribution as topic vectors and they serve their purpose as inputs for the Topic Tiling algorithm. In contrast to some older algorithms for text segmentation such as Text Tiling, Topic Tiling algorithm does not use real words, but it uses topic distribution over words in a sentence.

By using topic tiling, we have the ability to augment the number of topics for different books. This is a useful feature for books of varying sizes, where we can specify the number of latent topics that we want to create a topic distribution over.

We show the output of topic tiling in Table 3.4.3.3. From the Table 3.4.3.3, we identify three tags namely - segment, depthScore and text. In the text block, we have parts of the novel. In depthScore, we have the score for a possible segment boundary. This score varies from 0 to 1. The average score(mean) and standard deviation of the scores for each book is given in Fig 3.3.

Listing 3.1: Topic Tiling Output for A.B.C Murders

```xml
<segment>
<depthScore>0.29289321881345254</depthScore>
<text>
To grow the vegetable marrows! And immediately a murder occurs – and I send the vegetable marrows to promenade themselves to the devil. And since then – I know very well what you will say – I am like the Prima Donna who makes positively the farewell performance! That farewell performance, it repeats itself an indefinite number of times!
I laughed.
```
In truth, it has been very like that. Each time I say: This is the end.

But no, something else arises! And I will admit it, my friend, the retirement I care for it not at all. If the little grey cells are not exercised, they grow the rust.

I see, I said.

You exercise them in moderation.

Precisely. I pick and choose. For Hercule Poirot nowadays only the cream of crime.

Has there been much cream about?

Pas mal. Not long ago I had a narrow escape.

Of failure?

No, no. Poirot looked shocked. But I—I, Hercule Poirot, was nearly exterminated.

I whistled. An enterprising murderer!

Not so much enterprising as careless, said Poirot.
In Table 3.4.3.3, for longer books like Angels and Demons (148,768 words) and Killing Floor (144,467 words) which have the highest number of segments, the weighted clue selection allows greater flexibility in creating boundaries as there are a wide variety of topic shifts in these books. An attempt to make an absolute cutoff score for topic boundaries is tricky, as there should be some relation between the number of segments and the structure and style of the text. Instead, we devise a function which takes in the mean and standard deviation of these scores and determines a segment boundary (assuming the scores are normally distributed), as suggested by Hearst [10]. The function entails a segment boundary only if the score exceeds $\bar{S} - \sigma^2$ (mean - variance).

**Summary**

The algorithms and methods mentioned in this chapter illustrate our approach in extracting the evidence sentences in the novel. We define some of the terminologies used in Section 3.2. We introduce our pipeline which breaks down our method into blocks. We pick out our culprit reveal sentence using the binning technique in Section 3.3. In the first set of experiments, we divide the novel with the help of chapter boundaries. This allows us to use the author’s intuition...
<table>
<thead>
<tr>
<th>Book Name</th>
<th>Mean of depthScores</th>
<th>Std. Deviation</th>
<th>Mean - Variance</th>
<th>Segments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Killing Floor</td>
<td>0.5661</td>
<td>0.3009</td>
<td>0.4755</td>
<td>2664</td>
</tr>
<tr>
<td>The Five Orange Pips</td>
<td>0.4060</td>
<td>0.2921</td>
<td>0.320</td>
<td>129</td>
</tr>
<tr>
<td>Mysterious Affair at Styles</td>
<td>0.4732</td>
<td>0.3103</td>
<td>0.3768</td>
<td>624</td>
</tr>
<tr>
<td>Sign of Four</td>
<td>0.4252</td>
<td>0.3026</td>
<td>0.33</td>
<td>370</td>
</tr>
<tr>
<td>A study in Scarlet</td>
<td>0.3620</td>
<td>0.2747</td>
<td>0.286</td>
<td>321</td>
</tr>
<tr>
<td>ABC Murders</td>
<td>0.3620</td>
<td>0.2747</td>
<td>0.385</td>
<td>732</td>
</tr>
<tr>
<td>Red Dragon</td>
<td>0.4612</td>
<td>0.3115</td>
<td>0.364</td>
<td>1319</td>
</tr>
<tr>
<td>A Case of Identity</td>
<td>0.3569</td>
<td>0.2666</td>
<td>0.2858</td>
<td>52</td>
</tr>
<tr>
<td>Angels and Demons</td>
<td>0.5556</td>
<td>0.3221</td>
<td>0.451</td>
<td>2198</td>
</tr>
<tr>
<td>Monte Cristo</td>
<td>0.4003</td>
<td>0.3131</td>
<td>0.320</td>
<td>220</td>
</tr>
<tr>
<td>Adv. of Blue Carbuncle</td>
<td>0.4726</td>
<td>0.2854</td>
<td>0.391</td>
<td>75</td>
</tr>
<tr>
<td>Girl with the Dragon Tattoo</td>
<td>0.3922</td>
<td>0.3022</td>
<td>0.3008</td>
<td>1547</td>
</tr>
<tr>
<td>Five go adventuring again</td>
<td>0.5031</td>
<td>0.3001</td>
<td>0.4130</td>
<td>540</td>
</tr>
<tr>
<td>Moonstone</td>
<td>0.3461</td>
<td>0.2799</td>
<td>0.267</td>
<td>1472</td>
</tr>
<tr>
<td>And then there were none</td>
<td>0.5226</td>
<td>0.2914</td>
<td>0.437</td>
<td>745</td>
</tr>
<tr>
<td>Murder of Roger Ackroyd</td>
<td>0.4761</td>
<td>0.3083</td>
<td>0.381</td>
<td>907</td>
</tr>
<tr>
<td>Murder in Mesopotamia</td>
<td>0.4697</td>
<td>0.3083</td>
<td>0.371</td>
<td>786</td>
</tr>
<tr>
<td>The Maltese Falcon</td>
<td>0.4205</td>
<td>0.3114</td>
<td>0.323</td>
<td>751</td>
</tr>
<tr>
<td>Hounds of Baskerville</td>
<td>0.3569</td>
<td>0.2877</td>
<td>0.274</td>
<td>468</td>
</tr>
<tr>
<td>Five on a Treasure Island</td>
<td>0.4709</td>
<td>0.3035</td>
<td>0.378</td>
<td>499</td>
</tr>
<tr>
<td>The Cuckoo’s Calling</td>
<td>0.3932</td>
<td>0.3242</td>
<td>0.288</td>
<td>1215</td>
</tr>
<tr>
<td>Five Go to Smuggler’s Top</td>
<td>0.4672</td>
<td>0.3130</td>
<td>0.369</td>
<td>578</td>
</tr>
<tr>
<td>The Lost Symbol</td>
<td>0.4805</td>
<td>0.3250</td>
<td>0.374</td>
<td>1782</td>
</tr>
</tbody>
</table>
for the demarcation of a different topic. With topic tilling, we make use of topic shifts to create demarcations in the text. This allows us more flexibility with long documents as it ensures good coverage and provides us with the overview of topics in the book.

In the following chapter, we glean into the writing styles of detective novel authors.
Chapter 4

Stylistic analysis of novels

4.1 Introduction

Evaluating sentences with a baseline from any generic summarising or IE techniques would not be a fair comparison owing to narrative dependencies, writing styles and a specific problem of gathering evidence sentences. Hence, we evaluate our extracted sentences from a pool of human readers. These readers were recruited from the author’s home university and a local book club. From a pool of 24 possible participants, we recruited 10 participants for our evaluation task, 7 people who identified as female and 3 who identified as male by ensuring that the participants had similar flair for reading comprehension and understanding. For each book we found 3 participants who had read the book and were then tasked with evaluating the quality of the our evidence sentences.

Since the participants were aware of the stories and specifically the culprit and how he/she was revealed, we asked them to pick out sentences from the novel which were indicating of the crime and could be classified as evidence. The 3 participants were to arrive to a conclusion on the sentences from the novel which could be classified as evidences related to the crime. This does not imply that only sentences which implicate the culprit were to be picked, but even events and objects which are red-herrings relate to the crime were to be selected. The broad guidelines were

1. "Does this sentence point to an event or object which implicates any character in the story?"
2. "Does the event or object of interest in this sentence have anything to do with the culprit who is (eventually) revealed?"
3. "Does this sentence point to another clue or seemingly important bit of information?"
<table>
<thead>
<tr>
<th>Score</th>
<th>Guideline</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>contains no overlap between the extracted sentences and readers’ set</td>
</tr>
<tr>
<td>1</td>
<td>The extracted sentences contain some of the major clues from readers’ set, but a lot of unnecessary information too which are not a good fit.</td>
</tr>
<tr>
<td>2</td>
<td>The extracted sentences contain most of the major clues from the readers’ set but contain a fair amount of unnecessary sentences</td>
</tr>
<tr>
<td>3</td>
<td>There is a major overlap between the extracted sentences and the readers’ set but the method extracts more sentences than readers’, some of which are not useful</td>
</tr>
<tr>
<td>4</td>
<td>There is a very significant overlap between the extracted sentences and the readers’ set and are very similar</td>
</tr>
</tbody>
</table>

Table 4.1: A rubric for scoring sentences by degree of relevance, corresponding to quality of evidence.

Following these broad guidelines, our participants picked sentences from the novels and came to conclusion amongst their group. The next task for each group was to compare the evidence sentences extracted from our method to the ones that they had selected. Following the rubric described in Table 4.1, they were to assign a score of 0-4, 0 broadly indicating that our method extracts sentences which are way off the actual sentences which indicate evidence, and 4 broadly indicating that there is a very significant overlap between our extracted sentences and manually selected set of sentences in the group.

The mean score over all the 20 novels is 3.1 and with a standard deviation of 0.55. This indicates that our method is able to capture a substantial set of sentences which are either potentially evidence or are indicative of a major event or object of interest. With this, we can use our developed technique in an unsupervised fashion over multiple detective novel texts to analyse writing styles in the genre. We show the results of our algorithm from The murder of Roger Ackroyd in Table 4.2 and Then There were None by Agathe Cristie in 4.3.
It was Poirot’s voice speaking, and I knew from the gravity of his tone that he, too, was fully alive to the implications of the position."

I just stood there telling him what I thought of him, and saying the coldest, cruelest things that came into my mind - trying my best to hurt him.”

The lamps were arranged in such a way as to throw a clear light on the side of the room where the chairs were grouped, at the same time leaving the other end of the room, where I presumed Poirot himself would sit, in a dim twilight.

As they did so, the door opened once more and two other people came in and sat down near the door.

Then I learnt that Ralph Paton had been seen coming up the path which led to the summer-house at twenty-five minutes past nine, and I also heard of a certain conversation which had taken place in the wood near the village that very afternoon - a conversation between Ralph Paton and some unknown girl.

Poirot leaned forward and shot the last words triumphantly at us, drawing back afterwards with the air of one who has made a decided hit Raymond, however, did not seem impressed, and lodged a mild protest

The thought of having possibly to give evidence which might incriminate his wife made him resolve at all costs to - to -” I hesitated and Ralph filled up the gap.

It occurred to me that he might have so insulted her - in such an unforgivable manner - that without knowing what she was doing -” He stopped.

If the good doctor is concealing the young man, what place would he choose?

Yes, at one of them a patient was brought there by the doctor himself early on Saturday morning.

That patient, though known by another name, I had no difficulty in identifying as Captain Paton.

For a moment I was inclined to think that the scene I had just witnessed was a gigantic piece of bombast - that he had been what he called “playing the comedy” with a view to making himself interesting and important.
He was on the terrace outside, and couldn’t catch the words clearly, but he distinctly heard the voices.”

“All the same,” he remarked, “this discovery of yours, brilliant though it is (I’m quite sure I should never have thought of it), leaves the essential position unchanged.

He is in a most unfortunate position, but if he were to come forward -” Poirot interrupted.

As soon as the murder was discovered, I realized that once the facts were known, suspicion could not fail to attach to Ralph - or, if not to him, to the girl he loved.

"I satisfied myself that the call could not have been sent by anyone in the house, yet I was convinced that it was amongst those present on the fatal evening that I had to look for my criminal.

For instance, it was something that the murderer had not been able to take away with him at the time that he committed the crime.

A person who was on the scene straight-away, but who might not have been if the crime had been discovered the following morning.

A person who was at the Three Boars earlier that day, a person who knew Ackroyd well enough to know that he had purchased a dictaphone, a person who was of a mechanical turn of mind, who had the opportunity to take the dagger from the silver Table before Miss Flora arrived, who had with him a receptacle suitable for hiding the dictaphone - such as a black bag, and who had the study to himself for a few minutes after the crime was discovered while Parker was telephoning for the police.

Table 4.2: Evidence containing sentences of The Murder of Roger Ackroyd [4] by Agatha Cristie (last 20 lines). Sentences in bold represent some crucial information about the crime.
<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
</table>
| 1 | She said, her breath coming with a slight catch in it: "One feels safer here, out in the open. . . . | 16 | The A.C. went on: "Ten people dead on an island and not a living soul on it.
| 2 | It’s one at a time, and it’s got to be done in a certain way.” | 17 | Inspector Maine went on: "It was Morris who made all the arrangements down at Sticklehaven.
| 3 | He may have got rid of Armstrong a couple of hours before that.” | 18 | The men got there on the afternoon of the 12th at the first moment possible to run a boat ashore there.
| 4 | "Tell me-you don’t think-” She broke off, went on: "I read a story once about two judges that came to a small American town from the Supreme Court. | 19 | Starting with the Rogerses; who were the first to arrive on the island.
| 5 | There was a man in it probably. | 20 | It’s true that there was a woman called Clees who was operated on by him way back in 1925 at Leithmore, when he was attached to the hospital there.
| 6 | He was spread-eagled on the stone terrace on the east side, his head crushed and mangled by a great block of white marble. | 21 | Isaac Morris died on the night of August 8th.
| 7 | So he was physically, and he was on the lookout too. | 22 | After his death Vera Claythorne’s diary states that Armstrong left the house in the night and that Blore and Lombard had gone after him.
| 8 | The sun was dropping towards the west. | 23 | He got to the island early on the morning of August 13th.
| 9 | The man was wedged between two rocks, flung there by the tide earlier in the day. | 24 | Here’s the position early on the morning of the 11th.
| 10 | Or was it the day before? | 25 | His body was down by the sea-near Armstrong’s. |
| 11 | She looked down at the dead man. |
| 12 | **Philip Lombard** was dead-shot through the heart. |
| 13 | The sun was setting when Vera moved at last. |
| 14 | She heard them crash on the stone of the terrace. |
| 15 | Funny, how she suddenly got the feeling again that Hugo was in the house. . . |
| 26 | That she shot Lombard, took the revolver back to the house, toppled the marble block onto Blore and then-hanged herself. " |
| 27 | When in due course I came to preside over a court of law, that other secret instinct of mine was encouraged to develop. |
| 28 | During the time I was in a nursing home I collected the case of Dr Armstrong—a violently teetotal sister who attended on me being anxious to prove to me the evils of drink by recounting to me a case many years ago in hospital when a doctor under the influence of alcohol had killed a patient on whom he was operating. |
| 29 | At a late hour one night the sole occupants of the smokingroom were myself and a good-looking young man called Hugo Hamilton. |
| 30 | **Justice Wargrave killed Rogers on the morning of August 10th.** |

Table 4.3: Evidence containing sentences of *Then There were None* [6] by Agatha Cristie (last 30 lines). Sentences in bold represent some crucial information about the crime.

### 4.2 Analysis of Author’s writing style

With the help of two graduate students in English literature, we analysed the evidence sentences gathered from our method for a set of 35 novels, 15 more than the ones we used to evaluate our method previously. We investigate the different writing styles of popular detective novel authors, and primarily find how they obfuscate evidence in the narrative.

From the gathered evidence sentences, and our knowledge of the novels and their culprits, we analyse the three questions sourced from Twenty rules for writing detective novel storeis [23].

• Does the reader have the same opportunity as the detective to solve the crime from the presented evidence?

• Was the culprit found by logical deduction, or by luck, accident, or unmotivated confessions?

• Did the culprit play a prominent part of the story?

Based on this, we categorised the novels into hard, elusive and trivial.

**Hard Category**

Stories that fell into the *hard* category typically aimed to surprise the reader by hiding information. *Hard* novels are the ones in which most of the twenty rules given by [23] are violated and/or simply not written for reader deduction. Some of these rules include

• “The reader should have the same opportunity as the detective to solve the crime”.

• “The villain must be found by logical deduction, not luck, accident, or un-motivated confessions”.

• “The villain has to be someone who plays a prominent part of the story”.

To illustrate this category, we analyse some of Arthur Conan Doyle’s work. In Hound of Baskerville, Doyle weaves a gritty, fast-paced adventure crime novel where Sherlock tries to uncover the crime. The narrative is distributed with evidence containing sentences and plot twists making it one of his best selling books. We analyse the clues revealed by the author before the culprit reveal. Examples of some evidence containing sentences before and after the culprit reveal are shown in Table 4.4.
Table 4.4: Examples of evidence sentences before and after reveal.

<table>
<thead>
<tr>
<th>Before</th>
<th>After</th>
</tr>
</thead>
<tbody>
<tr>
<td>His lip had fallen, his eyes were protruding, his skin the colour of putty, and he glared at the envelope which he still held in his trembling hand, K. K. K.! he shrieked</td>
<td>have you never heard of the Ku Klux Klan?</td>
</tr>
<tr>
<td>There was nothing else save the five dried pips.</td>
<td>“that the sudden breaking up of the society was coincident with the disappearance of Openshaw from America with their papers.</td>
</tr>
<tr>
<td>The letter arrived on March 10, 1883. His death was seven weeks later, upon the night of May 2nd.</td>
<td>The body exhibited no traces of violence, and there can be no doubt that the deceased had been the victim of an unfortunate accident</td>
</tr>
<tr>
<td>Set the pips on McCauley, Paramore, and John Swain, of St. Augustine</td>
<td></td>
</tr>
</tbody>
</table>

From Table 4.4, we see that Doyle withholds critical information until revealing the culprit. This does not provide the reader with a realistic chance of guessing the culprit before the final reveal by the detective. The evidence gathered before the reveal and the evidence presented by the detective later have an information gap. The evidences picked directly from the text fall short as Doyle’s evidences are inter-wined with characters and components that advance the story. For instance in Five Orange Pips, Doyle connects a parcel received with KKK written on it to the Ku-Klux-Klan. The element of a parcel advances the story, but neither our method nor usual readers can make this connection. This connection forms the basis using which the detective is able to pinpoint the culprit. The novelty in Doyle’s writing stems from him leaving the reader in awe of the detective who could draw such amazing deductions. These deductions require information beyond the presented evidence in the text. Most of Doyle’s writing fell in this category as he presents evidence but it requires more intuition on reader’s behalf to make connection. *It requires the reader to be knowledgeable to the subject of discussion in Doyle’s books.*

Another popular author from our data set in this category is *Dan Brown*. In the evidence sentences extracted from *Angels and Demons*, none of the sentences mention the motivations of the culprit (or any of the other characters) strongly until the culprit paragraph we detected.
This gap cannot be bridged without additional knowledge provided by the author. Without information on motivations of character, it is difficult for the reader to guess the culprit of the story.

**Trivial Category**

Stories that we categorised as trivial, were ones such as *Eye of the Needle*, *Red Dragon* and *Lost Symbol*. Evidences gathered from the sentences, trivially help identify the culprit and the author does not put any additional effort into hiding information or misleading the reader. The evidence gathered along the story concludes the novel with apprehending the culprit. In *Red Dragon*, the author *Thomas Harris*, aimed to show the culprit in a gruesome fashion and how the culprit *Francis Dolarhyde* eludes the detective. The evidence sentences explicitly contained the identity of the culprit, and was mentioned 21 number of times. Our method was hence able to capture the entire murder trail of the culprits in this category. Our sentences also inform us that the evidences are collected either as a direct observation, or a recall. They are not hidden in other narrative components, and reader does not find it all that difficult to identify. Our evidence sentences also inform us that such stories are often a chase between the culprit and the detective and have a linear narrative very similar to non-detective genre stories.

**Elusive Category**

In the stories which we classified in the elusive category, the evidence sentences indicated a certain degree of obfuscation of evidence using narrative tricks. This was done by making it hard to sift out the evidence using certain crafty tricks like addition of phony clues, misdirection by adding spurious culprits and mention of seemingly irrelevant information which turned out to be key in the near future. For instance in *The Murder of Roger Ackroyd*, evidence sentences as shown in Table 4.2 frequently mention Ralph Paton and Ursala Bourne and they seemed to have strongest motives for wanting Mr Ackroyd out of the way but they were not the culprits. Though writers in this category present rich evidence, only an astute reader can always identify the culprit in advance. Summaries for *Agatha Christie* novels contain lines which readers overlook but are found to be of importance. For example, in *The Murder of Roger Ackroyd*, since the reader does not expect the narrator to be the culprit, many clues are left unnoticed; and in *Murder on the Orient Express*, 12 out of the 13 suspects are the culprits — something a reader would not expect apriori.

We present *Murder on the Orient Express* as a case study for its departure from the archetypal detective novel. In this book, a murder takes place while on a train journey and the detective in the book, Hercule Poirot, tries to find the evidence to catch the culprit. What is contrasting
about this particular book is that there are multiple people who are guilty of orchestrating the crime. Out of the 13 suspects, 12 were found to be guilty. This makes it an interesting book for analysis. Many of the characters use pseudonyms to conceal their true identity and motives. While conducting our analysis on this book, we find how the culprits are similar in the book. The graphs relating to four of the twelve perpetrators is shown in Figure 4.1.

From the graph, we see that the occurrence of these perpetrators sharply increase in the same neighborhood. We extract the evidence summary for all the characters in the novel. To do this, we mutate the text by adding an assertive sentence which makes one of the characters guilty of the crime. For example, *It was Miss Debenham who killed Ratchett*. This sentence is added at an appropriate part in the novel, where all the perpetrators were revealed. We observe that the evidence summaries of the different culprits overlap significantly. This can be attributed to the fact that all the culprits coordinated to commit the crime.

The sentences generated which mention the two characters from *The Murder of Roger Ackroyd* - Dr Sheppard and Ursula Bourne are shown in Table 4.2. It can be seen that much more relevant information is present in the evidence summary corresponding to Dr Sheppard.
Chapter 5

Results

In the following chapter, we discuss how our search function outperformed our baseline, LexRank algorithm through a study with human evaluators. We utilise these results from our human evaluators in categorising the different ways in which author obfuscate crime related evidence in their novels related to the main culprit. We also present a comparison on the possible reason why our algorithm performs better than the baseline.

5.1 Evaluation

We evaluated our evidence summaries with ten human evaluators who have read all the books or read the long summaries on Wikipedia and Sparknotes. The long summaries were a good representation of the story as verified by the authors of this paper. Initially, the evaluators were presented with the summaries from our baseline method and then they are presented with evidence summaries from our Augmented Search. The participants were asked to report if the evidence summaries presented to them for different characters in the novel assisted them in identifying the culprit in the novel. The responses of the participants were broadly categorized into three classes -

- The evidence summaries were useful in finding the right culprit (CI)
- The evidence summaries are misleading and direct the crime to a non-culprit character (NCI)
- The evidence summaries are not informative in labelling any character as culprit (NA)

From Table 5.1, we see that the Augmented Search has a higher CI percentage across novels, as compared to the group given the output of the LexRank. The highest CI score for
the baseline lexrank is 30%. That is, 3/10 readers found the baseline summary useful in finding
the culprit in the best case scenario of *Eye of the Needle*. In contrast, the second group presented
with our Augmented search reported a highest with 8/10 readers reporting CI (helpful in culprit
identification) for *Red Dragon*.

From the response of our readers presented with our Augmented A* summaries we draw
the following inferences -

1. Novels with a high percentage of readers who fall in *NA*, can be said to provide negligi-
   ble information about the culprit before revelation - *hard*.

2. Novels with high percentage of readers who fall in *CI* are the ones which make culprit
guessing easy and do not have sufficient obfuscation - *trivial*.

3. Novels with high percentage of readers who fall in *NCI* seem to provide only subtle
obfuscated information about the culprit - *elusive*.

In the *trivial* category, the highest *CI* percentage is 80% for *Eye of the Needle* and *Red
Dragon*. In *Red Dragon*, the author *Thomas Harris*, aims to show the culprit in a gruesome
fashion and how the culprit *Francis Dolarhyde* eludes the detective. The author does not aim
to hide the identity of the culprit.

In the *elusive* category, the highest percentage of readers in *NCI* is 70% indicating a certain
degree of obfuscation of evidence using narrative tricks. This is done by making it hard to sift
out the evidence using certain crafty tricks like addition of phony clues, misdirection by adding
spurious culprits and mention of seemingly irrelevant information which turns out to be key in
the near future. An astute reader can always identify the culprit in advance. Summaries for
*Agatha Christie* novels contain lines which readers overlook but are found to be of importance.
For example, in *The Murder of Roger Ackroyd* [4], since the reader does not expect the narrator
to be the culprit, many clues are left unnoticed; and in *Murder on the Orient Express* 12 out of
the 13 suspects are the culprits - something a reader would not expect apriori.

5.2 Baseline

We use the graph based summarization method Lexrank [8] as a baseline for creating ev-
idence based summaries. Lexrank works by building a graph representation of text. It then
computes sentence importance based on the eigenvector centrality and cosine similarity in the
graph representation of sentences. We then compute a connectivity matrix based on intra-
sentence cosine similarity. We find the closest sentences to the culprit sentence and create an
extractive summary of the character. The baseline output is a generic character summary of the characters, not explicitly focusing on evidence.

<table>
<thead>
<tr>
<th>Novel Title</th>
<th>Percentage Score</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Baseline Lexrank</td>
<td>Augmented Search</td>
</tr>
<tr>
<td></td>
<td>(CI) (NCI) (NA)</td>
<td>(CI) (NCI) (NA)</td>
</tr>
<tr>
<td>Adventure of Blue Carbuncle</td>
<td>0 10 90</td>
<td>10 10 80</td>
</tr>
<tr>
<td>Hound of Baskerville</td>
<td>0 10 90</td>
<td>0 10 90</td>
</tr>
<tr>
<td>A Study in Scarlet</td>
<td>0 0 100</td>
<td>0 0 100</td>
</tr>
<tr>
<td>The Sign of the Four</td>
<td>0 0 100</td>
<td>0 0 100</td>
</tr>
<tr>
<td>The ABC Murders</td>
<td>10 10 80</td>
<td>20 60 20</td>
</tr>
<tr>
<td>The Murder of Roger Ackroyd</td>
<td>0 10 90</td>
<td>20 70 10</td>
</tr>
<tr>
<td>Murder in Mesopotamia</td>
<td>10 10 80</td>
<td>10 60 30</td>
</tr>
<tr>
<td>And Then There Were None</td>
<td>10 20 70</td>
<td>10 40 50</td>
</tr>
<tr>
<td>The Mysterious Affair at Styles</td>
<td>0 20 80</td>
<td>30 50 20</td>
</tr>
<tr>
<td>The Girl with the Dragon Tattoo</td>
<td>20 0 80</td>
<td>40 50 10</td>
</tr>
<tr>
<td>The Maltese Falcon</td>
<td>10 10 80</td>
<td>20 20 60</td>
</tr>
<tr>
<td>The Moonstone</td>
<td>10 10 80</td>
<td>20 20 60</td>
</tr>
<tr>
<td>Eye of the Needle</td>
<td>30 10 60</td>
<td>80 10 10</td>
</tr>
<tr>
<td>Shock for the Secret Seven</td>
<td>10 10 80</td>
<td>30 10 60</td>
</tr>
<tr>
<td>Red Dragon</td>
<td>20 10 70</td>
<td>80 10 10</td>
</tr>
<tr>
<td>Killing Floor</td>
<td>10 10 80</td>
<td>20 30 50</td>
</tr>
<tr>
<td>Angels and Demons</td>
<td>10 20 70</td>
<td>30 20 50</td>
</tr>
<tr>
<td>The Lost Symbol</td>
<td>10 20 70</td>
<td>30 20 50</td>
</tr>
<tr>
<td>In the Woods</td>
<td>10 10 80</td>
<td>20 20 60</td>
</tr>
<tr>
<td>The Cuckoo’s Calling</td>
<td>10 0 80</td>
<td>10 60 30</td>
</tr>
</tbody>
</table>

Table 5.1: Comparison of baseline vs our solution over 20 novels. CI is the percentage of readers who found the summaries useful in finding the right culprit. NCI is the percentage of readers who found the summaries as leading to a non-culprit character. NA is the percentage of readers who found the summary as not informative in labelling any character as the culprit.
Chapter 6

Conclusion

In this thesis, we explored the novel problem of understanding and extracting information ingrained in detective novels. We formulate a problem of extraction of evidence from the detective’s point of view, and gather sentences containing evidence against potential culprits, thereby creating an evidence summary. In Chapter 2, we explored some of the other works in the field of short text summarisation and segmentation, while in section 2.1.2 we explored the genre of detective novel fiction and clever techniques employed by authors.

In Chapter 3, we define the terminologies and model our problem into a sentence extraction problem from text. We describe a binning method used in Section 3.3, which is very useful in breaking the novel into two parts - the before-reveal and the after-reveal. In the following Section, we explore various techniques to extract sentences and establish logical boundaries in the text. Our method uses an augmented search algorithm which takes in an implicit graph representation and finds an optimal path to the culprit reveal sentences. From thereon, we extract the relevant sentences and employ our sentences in the study of writing styles in the detective novel genre.

In chapter 4, we evaluate the results of our algorithms and various styles of detective novels. We create a rubric for scoring sentences by the degree of relevance, corresponding to the quality of sentences. By extracting relevant information (evidence sentences, here), we reduce the search space for analysis. In this reduced and relevant search space, we empirically analyse the genre and specifically find how writers obfuscate clues for the readers.

Our results in chapter 5 show that we outperform the baseline method in extracting useful evidence sentences through a study with human evaluators. With how the evaluators categorised the effectiveness of evidence summaries, we categorised the different ways in which culprit related evidence obfuscation happens in detective novels.
6.1 Limitations and Future Work

This thesis has been an attempt to computationally understand and gather evidence in detective novels. We pick this genre as it presents us with an opportunity to solve for the question - who was the perpetrator of the crime? The methods that are explored in this thesis can be applied to a different genre like court case documents to extract all relevant details with respect to the crime. Our methods are inspired by critically reading and analyzing the detective novels to understand caveats in writing styles. A particular inference which forms the backbone of our method is, to divide the novel into before-reveal and after-reveal of the culprit and see it as critical point in the story. Our method is an extractive method, which does not lead to coherent summaries. Abstractive methods applied over the selected evidence sentence would give a more coherent understanding of the text. Studying other domains and genres of writing could similarly inform us of their critical points around which information is centered which can help us develop methods which reveal insights into them. While we utilise implicit graph representation of the novel text for generating evidence summaries and identifying sentences of interest, deep learning techniques could also be utilised for identifying salient patterns and distinctive writing styles of authors. Lately there have been improvements in generating better abstractive summarisation of a text. These techniques could be deployed to develop better evidence summaries as they may preserve coherence and maintain better coverage.

As future work, we would try to find linguistic features and techniques which are widely used in this genre. We hope to explore the use of active/passive voice, characters’ narrative arcs, detectives’ statements to better understand this genre. We could also incorporate the use of knowledge graphs, which would provide context specific domain knowledge and a granular view of the domain. As our technique exploits the inherent semantic knowledge in a text, one can evaluate its feasibility in the domain of court documents, police reports and autopsy reports (documents with some pre-defined structure). Another interesting application would be in the domain of helper agents, which can discuss detective novels with humans. Such an application could be realized on a voice-activated agent like Amazon Echo or Google Assistant.
Appendix A

Sample Survey Question and Answer Set

Question 1) - Pick sentences from the detective novel given below, based on the following guidelines:

1. "Does this sentence point to an event or object which implicates any character in the story?"

2. "Does the event or object of interest in this sentence have anything to do with the culprit who is (eventually) revealed?"

3. "Does this sentence point to another clue or seemingly important bit of information?"

Evidence summaries picked from The Murder of Roger Ackroyd by Agatha Christie by one of the participants

2161 Incidentally this proved to me one thing, that both Ralph Paton and Ursula Bourne (or Paton) had the strongest motives for wishing Mr Ackroyd out of the way.

2162 And it also made one other point unexpectedly clear. It could not have been Ralph Paton who was with Mr Ackroyd in the study at nine-thirty.

2163 “So we come to another and most interesting aspect of the crime. Who was it in the room with Mr Ackroyd at nine-thirty? Not Ralph Paton, who was in the summer-house with his wife. Not Charles Kent, who had already left.

2166 Raymond, however, did not seem impressed, and lodged a mild protest ”I don’t know if you’re trying to make me out a liar, M. Poirot, but the matter does not rest on my evidence alone - except perhaps as to the exact words used.

2167 Remember, Major Blunt also heard Mr Ackroyd talking to some one. He was on the terrace outside, and couldn’t catch the words clearly, but he distinctly heard the voices.”
"Yet there must have been some reason for his thinking so," mused Poirot. "Oh! no," he held up his hand in protest, "I know the reason you will give - but it is not enough. We must seek elsewhere.

I will put it this way. From the beginning of the case I have been struck by one thing - the nature of those words which Mr Raymond overheard. It has been amazing to me that no one has commented on them - has seen anything odd about them.”

He paused a minute, and then quoted softly: "... The calls on my purse have been so frequent of late that I fear it is impossible for me to accede to your request. Does nothing strike you as odd about that?"

"Exactly," cried Poirot. "That is what I seek to arrive at. Would any man use such a phrase in talking to another? Impossible that that should be part of a real conversation. Now if he had been dictating a letter -"

"But why? We have no evidence that there was any one else in the room. No other voice but Mr Ackroyd’s was heard, remember.”

"You have all forgotten one thing,” said Poirot softly: "the stranger who called at the house the preceding Wednesday."

"But yes,” said Poirot, nodding encouragingly, "on Wednesday. The young man was not of himself important. But the firm he represented interested me very much.”

"The Dictaphone Company,” gasped Raymond. "I see it now. A dictaphone. That’s what you think?"

Table A.1: Evidence Summaries picked from The Murder of Roger Ackroyd by Agatha Christie by one of the participants. The participant was presented the entire book with corresponding line numbers.

**Question 2) - We present the evidence summaries from the algorithm. Compare them with the ones you picked to output and assign a score according to the rubric.**

I was obliged to go out to a case at some distance away, and it was past eight o’clock when I got back, to be greeted with a plate of hot dinner on a tray, and the announcement that Poirot and my sister had supped together at half-past seven, and that the former had then gone to my workshop to finish his reading of the manuscript.

Come, we must go over to my house and set the stage for my little performance. The lamps were arranged in such a way as to throw a clear light on the side of the room where the chairs were grouped, at the same time leaving the other end of the room, where I presumed Poirot himself would sit, in a dim twilight.

"Ralph was in a corner and took the only way out.

There was a suggestion in all this as of a trap - a trap that had closed.
That suggested at once to my mind a taker of drugs - and one who had acquired the habit on the other side of the Atlantic where sniffing 'snow' is more common than in this country.

Then I learnt that Ralph Paton had been seen coming up the path which led to the summer-house at twenty-five minutes past nine, and I also heard of a certain conversation which had taken place in the wood near the village that very afternoon - a conversation between Ralph Paton and some unknown girl.

Poirot leaned forward and shot the last words triumphantly at us, drawing back afterwards with the air of one who has made a decided hit. Raymond, however, did not seem impressed, and lodged a mild protest.

He was on the terrace outside, and couldn’t catch the words clearly, but he distinctly heard the voices."

no," he held up his hand in protest, "I know the reason you will give - but it is not enough.

His conscious mind was occupied with something quite different - the white figure he had caught a glimpse of.

When I was sufficiently master of myself to be able to realize what was going on, Ralph Paton was standing by his wife, her hand in his, and he was smiling across the room at me.

At first he refused to take me into his confidence, but later he told me about his marriage, and the hole he was in.

As soon as the murder was discovered, I realized that once the facts were known, suspicion could not fail to attach to Ralph - or, if not to him, to the girl he loved.

The thought of having possibly to give evidence which might incriminate his wife made him resolve at all costs to - to -" I hesitated and Ralph filled up the gap.

Yes, at one of them a patient was brought there by the doctor himself early on Saturday morning.

That was for a reason.

For a moment I was inclined to think that the scene I had just witnessed was a gigantic piece of bombast - that he had been what he called "playing the comedy" with a view to making himself interesting and important.

"I satisfied myself that the call could not have been sent by anyone in the house, yet I was convinced that it was amongst those present on the fatal evening that I had to look for my criminal.

For instance, it was something that the murderer had not been able to take away with him at the time that he committed the crime.

A person who was on the scene straightaway, but who might not have been if the crime had been discovered the following morning.
A person who was at the Three Boars earlier that day, a person who knew Ackroyd well enough to know that he had purchased a dictaphone, a person who was of a mechanical turn of mind, who had the opportunity to take the dagger from the silver table before Miss Flora arrived, who had with him a receptacle suitable for hiding the dictaphone - such as a black bag, and who had the study to himself for a few minutes after the crime was discovered while Parker was telephoning for the police.

Table A.2: A sample result from Murder of Roger Ackroyd by Agatha Christie. The user was was to assign a score to the summary based on the rubric in table A.3

<table>
<thead>
<tr>
<th>Score</th>
<th>Guideline</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>contains no overlap between the extracted sentences and readers’ set</td>
</tr>
<tr>
<td>1</td>
<td>The extracted sentences contain some of the major clues from readers’ set, but a lot of unnecessary information too which are not a good fit.</td>
</tr>
<tr>
<td>2</td>
<td>The extracted sentences contain most of the major clues from the readers’ set but contain a fair amount of unnecessary sentences</td>
</tr>
<tr>
<td>3</td>
<td>There is a major overlap between the extracted sentences and the readers’ set but the method extracts more sentences than readers’, some of which are not useful</td>
</tr>
<tr>
<td>4</td>
<td>There is a very significant overlap between the extracted sentences and the readers’ set and are very similar</td>
</tr>
</tbody>
</table>

Table A.3: A rubric for scoring sentences by degree of relevance, corresponding to quality of evidence.

The above participant assigned a score of 3 to the summary extract.
Related Publications

1. Extracting Evidence Summaries from Detective Novels, Aditya Motwani, Aayush Naik, Kamalakar Karlapalem, 2nd Text2Story Workshop, European Conference on Information Retrieval (ECIR), 2019
Bibliography


