

Extracting Evidence Summaries from Detective Novels

Aditya Motwani
DSAC, IIIT Hyderabad
aditya.motwani@research.iiit.ac.in

Aayush Naik
DSAC, IIIT Hyderabad
aayush.naik@students.iiit.ac.in

Kamalakar Karlapalem
DSAC, IIIT Hyderabad
kamal@iiit.ac.in

Abstract

Literary novels are a mine of information stored in a narrative. Information Extraction (IE) on these novels can be directed by a specific aim to derive knowledge of interest. In this paper, we explore the problem of extracting evidence summaries for all characters in a detective novel. But in this domain, there is no standard annotated text which provides the ground truth for evidence and there exists no fixed metric of evaluation. For extracting summaries, we explore unsupervised methods of learning. We first assume that all characters are equally likely to be culprits and build evidence against each character. Given this evidence, we assess the likelihood of each character being the culprit, eventually arriving at the true culprit(s). Our contribution is the formulation of evidence summaries for characters in determining the culprit. Our experiments are based on a corpus of detective novels spanning the last hundred years. We evaluate these summaries for comprehension with the help of human readers and broadly categorize novels into three groups.

1 Introduction

Detective Novels are a mine of information embedded in a narrative. In this paper, we explore the problem of extracting evidence summaries from detective novels. A big difficulty and novelty of this problem can be attributed to the widely varying writing style of different authors. This makes it very difficult for standard Information Extraction (IE) based techniques to extract important sentences in novels across multiple books 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 this domain of detective novels, there is no existing annotated data, and there is no fixed metric of evaluation. Moreover, making such a corpus is not as simple as just labeling a sentence as clue or not a clue by just a cursory glance at 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 a valid clue. Therefore it is difficult to extract data in a supervised way. Therefore, our methods primarily focus on unsupervised techniques.

Copyright © 2019 for the individual papers by the paper's authors. Copying permitted for private and academic purposes. This volume is published and copyrighted by its editors.

In: A. Jorge, R. Campos, A. Jatowt, S. Bhatia (eds.): Proceedings of the Text2StoryIR'19 Workshop, Cologne, Germany, 14-April-2019, published at <http://ceur-ws.org>

These techniques exploit the underlying semantic structure in a novel using spacy’s [HM17] sentence vectors as mathematical representation which attempts to overcome the challenge of manual annotation of sentences.

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. 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. The standard approach to solve this problem is to first extract evidence and then deduce the likely culprit. However, without a concrete definition of valid evidence, this problem is difficult to address. So, our methods can be viewed as approaching the problem backwards. We extract and evaluate the possible evidence against every character in the novel. We only collect the evidence relating to the crime that was revealed before the culprit is declared. This gives us an evidence map of each character. An analysis of this map by human evaluators gives us the likelihood of the said character being the culprit.

We focus our analysis on the text before the aforementioned juncture. Given the juncture of revelation as a priori information, we define what constitutes as valid evidence (see Section 3) using different algorithms and metrics. Our first step is to use trends pertaining to the domain of detective novels to determine the important character (especially the culprit) in novels.

We need to extract the evidence relating to the crime that was revealed *before* the culprit is declared. Using heuristics defined in section 3.1 we find the approximate position of the culprit revealing sentence (defined in section 3.0.2) in the novel by narrowing down to the relevant paragraph. Since the paragraph contains only a few sentences, most of which are related to the culprit, it is feasible to find the revealing sentence.

We use our algorithm which refines text into potential evidence. We then validate our results by employing human readers given two summaries of each character - one with baseline summaries (described in section 3.3) and the other with summaries from our approaches - and comparing them in table 2. From our gathered evidence summaries and our responses from our human readers, we broadly classify the novels into three categories based on the extracted evidence.

In this paper, we talk about related work in section 2. We detail the overview of our solution and some necessary definitions in section 3. Next, we describe the exact methods and algorithms used in section 3.1 and 3.2. We provide and explain our results in section 4. We finally conclude and talk about future work in section 5.

2 Related Work

Summarization of large text documents is an important task in the NLP domain. However, summarization techniques for texts with a narrative structure are challenging to produce. [LE11] attempted a method for evidence discovery along with visual aid in understanding the characters and structure in detective novel genre. Moen et.al [MPH⁺16] generate extractive summaries on medical reports using word space models in distributional semantic modelling. Medical reports involves temporal patient history, and unpacks in similar ways to detective novels.

Some of the popular techniques for summarizations happen graph based. Algorithms such as [TDC10] has been attempted on different domains using different algorithms. Thakkar et al [TDC10] 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. In our corpus, we observe there is no correlation between PageRank of sentences and their importance with respect to evidence against the culprit. Aker et al [ACG10] 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.

3 Evidence extraction from detective novels

3.0.1 Evidence

Any piece of information that increases the confidence of the reader that a particular character is the culprit can be categorised as evidence. In the context of our problem, every sentence in list of sentences of the output of the search algorithm is a potential evidence sentence. In table 1, we show some examples of evidence sentences retrieved by our search algorithm for the novel *The ABC Murders* [Chr03] by *Agatha Christie*.

Table 1: Clues from The ABC Murders

Miss Grey here told us that she did not see or speak to any stranger on the day that Sir Carmichael Clarke was murdered.
There was the possibility that the footmarks might have been made by somebody else who happened to have the same kind of studs in his shoes.
According to the police theory, Ralph was wearing another pair of the same kind, and I found out that it was true that he had two pairs.

3.0.2 Culprit Sentence

The first sentence in the novel that makes it unambiguous and lucid who the true culprit is.

Very often, this is an assertion or deduction by the prime detective following which, the detective explains how he/she arrives at that deduction. Or sometimes, it’s a confession by the culprit.

3.1 Culprit Frequency Binning

We divide the text into buckets and glean information from aggregation of these buckets. The information we are looking for is the location in the novel where the culprit is revealed.

First we look for the approximate location (paragraph) where the culprit is revealed. To achieve this, we use a bucket optimization algorithm. We want to maximize the frequency of occurrence of the culprit in a paragraph. Using the bucketing algorithm, we optimize the size of a paragraph to get the best frequency. We group consecutive sentences into bins of sentences and count the frequency of the culprit in each bin. By varying bin size, we arrive at an optimal paragraph size where there is a sharp increase in the mentions of the culprit. Once we’ve identified the paragraph, our search space has reduced drastically. At the beginning of the culprit

Algorithm 1: Culprit Bucket Optimization

```

Function bucket Sentence[] doc, Integer min_bucket_size, Integer max_bucket_size
    Integer bucket_size = min_bucket_size
    Integer optimal_bucket_size = min_bucket_size
    Sentence[] culprit_bucket
    Integer max_mentions = 0
    while bucket_size ≤ max_bucket_size do
        Sentence[][] Buckets = emptyset()
        Sentence[] bucket = Buckets[bucket_size*i to bucket_size*(i+1)] : for i in range(number of buckets)
        Divide the list of sentences into buckets of size bucket_size
        Count the number of times the culprit is mentioned in each bucket
        Let mentions and bucket hold the values for the most mentions among all buckets and the bucket
        with most mentions respectively
        if mentions > max_mentions then
            max_mentions = mentions
            optimal_bucket_size = bucket_size
            culprit_bucket = bucket
        end
        bucket_size = bucket_size + 1
    end
    return culprit_bucket, optimal_bucket_size
end

```

paragraph, we add an asserting sentence which implicates the culprit. But such a sentence would not exist for a character who is not the culprit. For example, in the Murder of Roger Ackroyd by Agatha Christie, we add the sentence “Ursula Borne killed Roger Ackroyd.” as Ursula 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 that character was not the culprit with high certainty.

3.2 The Search Algorithm - Augmented A* Search

The search algorithm is a higher-order function that takes in a strategy function along with a starting sentence and a goal sentence. The strategy defines how and which sentences are picked. The aim is to find an optimal path, starting at the first sentence of the novel and ending at the culprit-revealing sentence.

Our Augmented A* algorithm in two parts - a general graph search algorithm [Algorithm (2)] that takes in a strategy as input and the augmented A* strategy [Algorithm (3)]. We use a variant of the A*- search (with additional hyperparameters) as the strategy function input to the higher-order graph search function.

The final output of the graph search is a list of sentences. This list of sentences is analogous to an extractive evidence summary. An example is shown in table 3.

Algorithm 2: Graph Search

```

Function search Sentence[] doc, Sentence start, Sentence goal, Strategy strategy
| Sentence current = start
| Set(Sentence) frontier =  $\emptyset$ 
| Set(Sentence) visited =  $\emptyset$ 
| while current != goal do
|   | yield current
|   | visited = visited  $\cup$  {current}
|   | Set(Sentence) new_nodes = Set(neighbors(current)) - visited - frontier
|   | frontier = frontier  $\cup$  new_nodes
|   | next_node, dist_from_start = strategy(doc, current, frontier, goal, dist_from_start)
|   | current = next_node
|   | frontier = frontier - {current}
| end
end

```

Algorithm 3: A* Strategy

```

Strategy astar Sentence[] doc, Sentence current, Set(Sentence) frontier, Sentence goal, Map
distance_from_start, Float  $\alpha$ , Float  $\beta$ , Function goal_heuristic, Function distance
| if distance_from_start is empty then
|   | distance_from_start = sent :  $\infty$  forall sent in doc
|   | distance_from_start[current] = 0
| end
| forall Sentence neighbor in neighbors(current) do
|   | dist_from_start[neighbor] = min(dist_from_start[neighbor], dist_from_start[current] +
|   | distance(current, neighbor))
| end
| Sentence next_node = argminnode {node | f(node) = minnode'  $\alpha$ .dist_from_start[x] +  $\beta$ .goal_heuristic(goal,
|   x)}
| return next_node, dist_from_start
end

```

3.3 Baseline

We use the graph based summarization method Lexrank [ER04] as a baseline for creating evidence 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.

4 Results

Evidence sentences are obfuscated in the narrative of a novel. There is no standard metric of evaluation or a baseline as such. We present a best-effort scenario by comparing our methods with the baseline and evaluating with human readers.

We evaluated our evidence summaries with 10 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 are presented with the summaries from our baseline method and then they are presented with evidence summaries from our Augmented AStar. 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*)” and “The evidence summaries are not informative in labelling *any* character as culprit (*NA*)”.

From table 2, we see that the Augmented A* Search has a higher *CI* percentage across novels, as compared to the first group. The highest *CI* score for the baseline lextank 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 A* reported a highest with 8/10 readers reporting *CI* (helpful in culprit identification) for *Red Dragon*.

We also categorize the novels into three types based on the rules detailed by vans et al [VD15] on writing detective stories and our intuition into - *hard*, *elusive*, and *trivial*. *Hard* novels are the ones in which most of the twenty rules given by [VD15] 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”.

This makes it impossible for the reader to deduce who the culprit is. *Elusive* novels follow most of the rules, and thus it is possible for a keen reader to deduce the culprit. *Trivial* novels may or may not follow the rules, but they just fail at hiding the culprit properly or don’t even aim to do so in the first place.

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 negligible 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.

Stories that falls into the *hard* category typically surprise the reader by hiding information. In *Hound of Baskerville* and *The Adventure of the Blue Carbuncle*, *Sherlock Holmes* withholds critical information until revealing the culprit, but it leaves the reader in awe of the detective who could draw such amazing deductions. This makes guessing who the culprit very difficult.

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* [Chr26], since the reader does not expect the

Novel Title	Author(s)	Percentage Score						Revelation Category
		Baseline Lexrank			A* Search			
		(CI)	(NCI)	(NA)	(CI)	(NCI)	(NA)	
Adventure of Blue Carbuncle	Arthur Conan Doyle	0	10	90	10	10	80	hard
Hound of Baskerville	Arthur Conan Doyle	0	10	90	0	10	90	hard
A Study in Scarlet	Arthur Conan Doyle	0	0	100	0	0	100	hard
The Sign of the Four	Arthur Conan Doyle	0	0	100	0	0	100	hard
The ABC Murders	Agatha Christie	10	10	80	20	60	20	elusive
The Murder of Roger Ackroyd	Agatha Christie	0	10	90	20	70	10	elusive
Murder in Mesopotamia	Agatha Christie	10	10	80	10	60	30	elusive
And Then There Were None	Agatha Christie	10	20	70	10	40	50	hard
The Mysterious Affair at Styles	Agatha Christie	0	20	80	30	50	20	elusive
The Girl with the Dragon Tattoo	Stieg Larsson	20	0	80	40	50	10	elusive
The Maltese Falcon	Dashiell Hammett	10	10	80	20	20	60	hard
The Moonstone	Wilkie Collins	10	10	80	20	20	60	hard
Eye of the Needle	Ken Follett	30	10	60	80	10	10	trivial
Shock for the Secret Seven	Enid Blyton	10	10	80	30	10	60	hard
Red Dragon	Thomas Harris	20	10	70	80	10	10	trivial
Killing Floor	Lee Child	10	10	80	20	30	50	hard
Angels and Demons	Dan Brown	10	20	70	30	20	50	hard
The Lost Symbol	Dan Brown	10	20	70	30	20	50	trivial
In the Woods	Tana French	10	10	80	20	20	60	hard
The Cuckoo's Calling	J. K. Rowling	10	0	80	10	60	30	elusive

Table 2: 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.

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 Conclusion and Future Work

In this paper, we explore the problem of extracting evidence summaries for characters in a novel. As there is no existing annotated data, we explore an unsupervised method of learning. We use an augmented A* algorithm which takes in an implicit graph representation and finds an optimal path to the culprit reveal sentence. The results show that the evidence summaries of the culprit contain much more information than the summaries generated for characters who are innocent of the crime. Furthermore, we evaluate our evidence summaries by presenting them to human readers and classify the novels into three categories based on the ease in which they were able to identify the culprit.

As future work, we would like to explore abstractive summarization as a technique to generate coherent evidence summaries. An interesting application would be in the domain of voice-activated agents, which would provide culprit narratives for humans. This agent would have the capability to extract facts from text and discuss detective novels with humans. Such an application could be realized on a voice-activated agent like Amazon Echo or Google Assistant.

References

- [ACG10] Ahmet Aker, Trevor Cohn, and Robert Gaizauskas. Multi-document summarization using a* search and discriminative training. In *Proceedings of the 2010 conference on empirical methods in natural language processing*, pages 482–491. Association for Computational Linguistics, 2010.
- [Chr26] Agatha Christie. The murder of roger ackroyd. *Collins*, 1926.
- [Chr03] A. Christie. *The ABC Murders: A Hercule Poirot Mystery*. Hercule Poirot Mysteries. HarperCollins, 2003.
- [ER04] Günes Erkan and Dragomir R Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research*, 22:457–479, 2004.
- [HM17] Matthew Honnibal and Ines Montani. spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. *To appear*, 2017.
- [LE11] A.L. Louis and A.P. Engelbrecht. Unsupervised discovery of relations for analysis of textual data. *Digital Investigation*, 7(3):154 – 171, 2011.

- [MPH⁺16] Hans Moen, Laura-Maria Peltonen, Juho Heimonen, Antti Airola, Tapio Pahikkala, Tapio Salakoski, and Sanna Salanterä. Comparison of automatic summarisation methods for clinical free text notes. *Artificial intelligence in medicine*, 67:25–37, 2016.
- [TDC10] Khushboo S Thakkar, Rajiv V Dharaskar, and MB Chandak. Graph-based algorithms for text summarization. In *Emerging Trends in Engineering and Technology (ICETET), 2010 3rd International Conference on*, pages 516–519. IEEE, 2010.
- [VD15] S.S. Van Dine. *Twenty Rules For Writing Detective Stories*. Bookclassic, 2015.

6 Appendix

Table 3: Evidence Summary of the A.B.C murders [Chr03] (last 10 lines)

<p>A newspaper might not print the first letter, but by the time the second crime took place, A.B.C. could have been assured of all the publicity the press could give.</p> <p>The great idea of such a killer is to hide his tracks - not to advertise them When we consider the four victims selected - or at any rate three of them (for I know very little of Mr. Downes or Mr. Earlsfield), we realize that if he had chosen, the murderer could have done away with them without incurring any suspicion.</p> <p>I may say that his description, as given me by Miss Grey, did not quite correspond with my own picture of the man who strangled Betty Barnard "I will pass over the next stages quickly.</p> <p>A fourth murder was committed - the murder of a man named George Earlsfield - it was supposed in mistake for a man named Downes, who was something of the same build and who was sitting near him in the cinema. "</p> <p>"Supposing, my friends, that while Cust committed three of the crimes - the A, C and D crimes - he did not commit the B crime."</p> <p>"Up to the time of the Barnard murder, no facts about the A.B.C. murders had been made public. It therefore followed that whoever killed Betty Barnard must have had access to facts known only to certain persons - myself, the police, and certain relations and neighbours of Mrs Ascher.</p> <p>"I had now only to review the various crimes and find the possible guilty person.</p> <p>Also I learned that he had his holiday early in August, which rendered it unlikely that he had anything to do with the Churston crime.</p> <p>The man I had known a long time in my secret mind was the same as the man whom I had known as a person.</p>
--