View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by SOURCE: Sheridan Scholarly Output Undergraduate Research Creative Excellence

Sheridan College

SOURCE: Sheridan Scholarly Output, Research, and Creative Excellence

Faculty of Applied Science and Technology - Exceptional Student Work, Applied Computing Theses

Exceptional Student Work

12-2019

Automatic Description: A Novel Approach to Documenting Character Description for Consistency in Long – Form Prose

Samantha Akulick
Sheridan College

Follow this and additional works at: https://source.sheridancollege.ca/student_work_fast_applied_computing_theses

 Part of the Science and Technology Studies Commons

SOURCE Citation

Akulick, Samantha, "Automatic Description: A Novel Approach to Documenting Character Description for Consistency in Long – Form Prose" (2019). *Faculty of Applied Science and Technology - Exceptional Student Work, Applied Computing Theses*. 5.
https://source.sheridancollege.ca/student_work_fast_applied_computing_theses/5

This work is licensed under a Creative Commons Attribution-Noncommercial-No Derivative Works 4.0 License.
This Thesis is brought to you for free and open access by the Exceptional Student Work at SOURCE: Sheridan Scholarly Output, Research, and Creative Excellence. It has been accepted for inclusion in Faculty of Applied Science and Technology - Exceptional Student Work, Applied Computing Theses by an authorized administrator of SOURCE: Sheridan Scholarly Output, Research, and Creative Excellence. For more information, please contact source@sheridancollege.ca.

AUTOMATIC DESCRIPTION DETECTION: A NOVEL APPROACH TO DOCUMENTING CHARACTER DESCRIPTIONS FOR CONSISTENCY IN LONG-FORM PROSE

A Thesis

Presented to

The Faculty of Applied Science and Technology, School of Applied Computing

of

Sheridan College, Institute of Technology and Advanced Learning

by

Akulick, Samantha

In partial fulfillment of requirements

for the degree of

Honours Bachelor of Computer Science (Mobile Computing)

December 2019

© Samantha Akulick, 2019

ABSTRACT

AUTOMATIC DESCRIPTION DETECTION: A NOVEL APPROACH TO DOCUMENTING CHARACTER DESCRIPTIONS FOR CONSISTENCY IN LONG-FORM PROSE

Samantha Akulick
Sheridan College, 2019

Advisor:
Dr. El Sayed Mahmoud

Currently, continuity editing for narrative fiction is performed manually. Many hours of human effort are required to comb through written works for inconsistencies. This study investigates the use of syntactic patterns of descriptions in narrative text and subject identification techniques like named entity recognition (NER) and coreferent resolution in narrative text as a step toward automated continuity analysis. This investigation involved examining natural English language to identify patterns used in descriptions and using natural language processing (NLP) techniques to identify those patterns and sentence subjects programmatically. Results were assessed by using the content of well-known works of fiction and two algorithms developed to identify sentence subjects and descriptions, to promising results. With the fragmented, iterative cycle of writing long-form prose and the limitations of human memory and reading speed, maintaining a clear and consistent image of a character's appearance and personality is a difficult task for human authors and editors to complete manually. The results of this research provide a starting point to automate and improve the process writing and proofreading narrative works.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE

1. INTRODUCTION

1.1 The Problem Context

Continuity is a challenge for writers of long-form prose. In writing novels, which are often in excess of 40 000 words. A writer describes characters over and over in different scenes as they work through the story. Descriptions provide material for mental visualization and occasionally act as cues to indicate who is speaking or acting. This means discrepancies in these descriptions can make writing difficult to understand and thus less effective. It is recognized in the industry that one of the first and most important steps in editing a novel manuscript involves checking for errors in continuity (Schmidt, Maintaining Continuity: Tales from the Copy Editor, 2015) (Einsohn, 2011). The author and/or an editor will manually read and re-read the entirety of the 40 000+ word manuscript using human memory and manual notes in order to check for conflicts and mistakes, adjust details where necessary, and start again to ensure that the adjustments didn't create new problems. Overall, the process takes at least as long as required for a human to read through the text. Given the limitations of human memory, there is no guarantee that all mistakes will be located and corrected.

Writers share strategies to mitigate the number of continuity errors online, and these approaches are also shared by professional editors (Schmidt, On Writing: Creating Characters and Maintaining Continuity in Writing, 2015). These strategies tend to include keeping details on post-it notes, rereading a manuscript while writing to ensure details are correct before editing, and most commonly keeping a collection of character profiles in what is often called a “story bible” or "story binder" (Lehman, 2018). Story binders are

collections of documents (printed or digital) describing people, places, things, timelines, and all manner of details pertaining to a story, used for reference when writing in order to maintain a clear and consistent view of those details in the text. This form of documentation is the most prevalent means for keeping details straight, but it suffers from the expectation that the writer will constantly refer to it and update it while writing and editing. This means it can take as long to maintain a story binder as it would to edit out the mistakes the documentation is meant to prevent, and that it will be ineffective if not consistently updated.

Software exists to create and maintain story binders (Scrivener) (yWriter) (Manuskript) but does not automatically track details. Existing solutions still rely on manual input and updates for any form of documentation and thus serve more as electronic organizers than documentation assistants in terms of editing or reviewing manuscripts.

1.2 Terms and Definitions

Table 1. Terms and Definitions

Character	An individual person who is mentioned in a work of writing.
Coreferents	The different ways of referring to the same entity. “The man”, “he”, “Joe”, “Joe Patterson”, and “Mr. Patterson” could all be coreferents for the same person.
Genre	A category of written work characterized by similarities in form, style, or subject matter.

Lexical Dependency	The structure of sentences, as in which words refer to or rely on others to make sense of them. If we say “he ran quickly” the word “ran” depends on the word “he” (who?) and “quickly” (how?)
Narrative Text	Text describing connected events; writing in the form of a story.
Natural Language Processing (NLP)	A field of computer science concerned with the interaction between computers and human languages (also known as natural languages.)
NeuralCoref	An extension for the spaCy library which uses neural networks to resolve coreferents in text.
Novel	A work of creative writing, variable in length but generally many chapters and often 40 000+ words.
Parts of Speech (POS)	Types of words with different purposes in language, including (but not limited to): verbs (action words), nouns (things), pronouns (he/she/they) and adjectives (descriptive words)
Prose	Language in a natural form rather than a measured poetic form.
spaCy	A python library for NLP.
Syntactic Analysis	Analysis of the syntax of text, including details like POS, lexical dependency and other features.

1.3 Problem Statement

Maintaining character continuity in novel manuscripts requires manually reading long (40 000+ word) documents multiple times and building external documentation. The process is so labour intensive that it is a large component of the copyediting profession, representing a significant investment of time and money for any manuscript before publication. This work develops two algorithms: one to identify descriptions and second to attribute them to particular characters. These algorithms are the first step in building a system to extract details about given characters over a work of long-form prose. These details may allow writers and editors to compare descriptions for continuity without rereading entire manuscripts and may assist literary analysts in their studies of character development.

1.4 Purpose

This thesis examines the potential for natural language processing, particularly using semantic patterns, to identify characters and to associate descriptions and details with those characters across a work of narrative fiction. This work aims to facilitate the process of writing, editing and studying novels as related to character description consistency or evolution by drastically reducing the amount of time required to locate and review character details in long texts.

The ultimate goal of this research is to provide the basis for an automatic character profiling tool for authors, editors, and literary analysts to support and reduce the time requirements of the production of works of fiction.

1.5 Motivation

The primary motivation of this research is to reduce the human effort and time required to create long-form fiction in order to support the creation of more and higher-quality novels. The amount of time required for an author to repeatedly re-read their own work reduces the time spent working on new stories. The cost of hiring someone to edit a novel is similarly problematic due to the amount of time and work required. However, the act of editing for the creation of consistent stories is important for written clarity and human understanding, and not something that should be ignored if a story is to be widely accepted. Reading has the capacity to promote empathy (Koopman, 2015) (Johnson, 2012) and social understanding (Dodell-Feder & Tamir, 2018) in human beings. Certain stories also have the ability to spark interest in students that may enhance learning experiences or form career paths – whether in their original written form or when adapted for film (Laprise & Winrich, 2010). Overall, works of fiction have a profound impact on the human experience, and any obstacle to the timely production of such works, or their quality, presents a loss for humanity.

1.6 Process

The research process for this thesis involved 4 steps. The first step involved determining the semantic patterns used in describing characters in a narrative text. This phase involved the manual examination of text to find descriptive sentences, the use of a semantic analysis tool to extract the patterns of these phrases, and analysis of the extracted patterns to find effective components. For example, given the phrase below from J.K. Rowling's *Harry Potter and the Philosopher's Stone* and a visualization of linguistic analysis from Google Cloud Natural Language service:

Figure 1.1 Visualization of Sentence Analysis

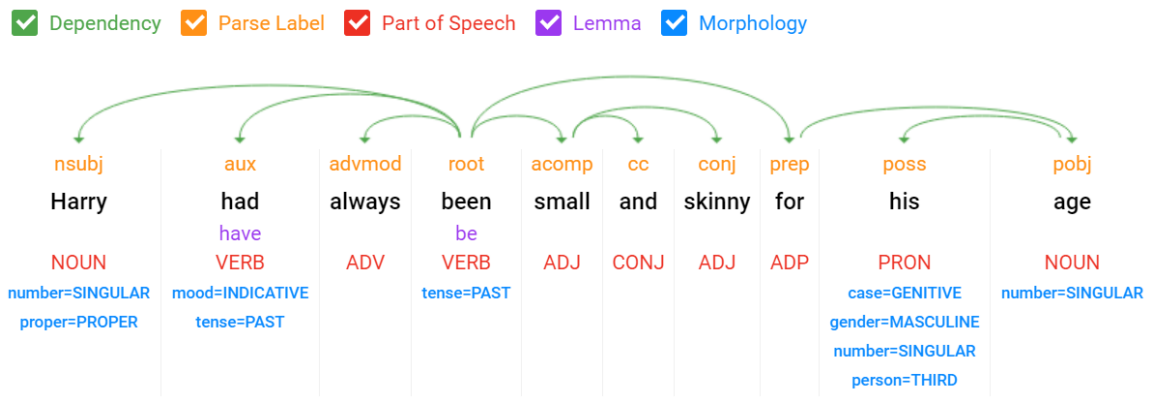


Figure 1.1 shows the variety of layers to the English-language sentences. By examining the patterns common to descriptive phrases, the significant components of analysis were identified. Parts of speech, lemma, parse labels and dependency form the meaningful patterns required for analysis. The second step was to codify those patterns into an algorithm to identify descriptions.

The third step was the examination of the same patterns in order to find the subject of a sentence, and the fourth step was the creation of a second algorithm to identify sentence subjects.

1.7 Thesis Statement

Using a combination of named entity recognition (NER), coreferent resolution, lexical and syntactic textual analysis, two algorithms were developed to identify descriptions and their subjects in narrative text. Semantic patterns in descriptive text are used to denote descriptions and attribute them to specific entities (or characters) in narrative text. These patterns could be used to identify sentences containing descriptive information about certain characters in narrative works. This research aimed to determine

whether descriptive sentences and sentence subjects could be identified algorithmically using recent natural language processing libraries (in this case spaCy and WordNet)

1.8 Contributions

This work shows how semantic patterns may be used to automatically identify descriptive sentences and determine the character or object to which they pertain. The contributions of this thesis include two algorithms: one which identifies descriptive sentences, and one which identifies the subject of a sentence.

1.9 Organization of Thesis

The structure of the thesis hereafter includes a literature review, a full explanation of the undertaken methodology, and the results obtained. The literature review focuses on prior research related to the analysis of fiction or narrative text, as well as natural language processing techniques (NLP). Pertinent literature and meta-analyses are discussed. The methodology chapter details the means of identifying semantic patterns and descriptive words, the construction of the identifier algorithms, and testing methods. Findings and analysis are discussed in the findings chapter, while conclusions, limitations and areas for future research are indicated in the results chapter.

CHAPTER TWO

2. LITERATURE REVIEW

Literary studies and the creation of literature itself predate the advent of computer science by a significant margin. Applications of computer science to augment research efforts into works of natural language narrative have been proposed since at least the 1990s with studies like (Wiebe, 1990) examining concrete methods for identifying the subjects of narrative sentences by using semantic analysis. More recent research applies more abstract forms of natural language processing such as word frequency and distribution to achieve a higher level understanding of narrative text features like character relationships (Elson, Dames, & McKeown, 2010) and emotions (Jhavar & Mirza, 2018) in works of narrative fiction.

The details extracted using these tools assist in the study of completed literary works, but make little contribution to the creation of new works. The use of natural language processing to assist writers of narrative fiction is a mostly unexplored area of computer science – despite recent studies establishing social benefits from the consumption of stories. Exposure to literature has a demonstrable positive impact on human empathy and charitable contributions (Koopman, 2015) and the abilities to participate in social behaviours and recognize the emotions associated with facial expressions (Johnson, 2012). Examining results across multiple studies, it's established that the positive social effects of reading fiction are significant compared to reading non-fiction (Dodell-Feder & Tamir, 2018) which tends to be the focus of natural language processing studies. The effects of fiction may be said to be more pronounced than those of non-fiction or academic writing. The concept is supported by evidence that when

adapted for film, works of fiction can influence the interest of youth towards the sciences and potentially inspire career choices (Laprise & Winrich, 2010).

Much research in the area of text mining and analysis has a focus outside of the area of narrative fiction. Natural, user-generated text as in social media posts (Akulick & Mahmoud, 2017) (Ghosh, et al., 2015) (Pozzi, Fersini, Messina, & Liu, 2016), newspaper articles (Chambers & Jurafsky, 2009) academic papers, and opinions (Al-sharman & Pivkina, 2018) are more frequently examined. Techniques have been developed to extract less explicit details from text. Sentiment analysis has been performed on social media posts by examining word choice and distribution, machine learning, and semantic analysis (Akulick & Mahmoud, 2017) (Ghosh, et al., 2015) (Pozzi, Fersini, Messina, & Liu, 2016).

Outside of Academia, efforts have been made to develop computing tools to aid in the process of writing and the creation of literature by digitally organizing details for reference (Manuskript) (Scrivener) (yWriter). It is recognized by professionals in literature – both authors and editors – that the organization of supporting documentation for a literary manuscript is necessary for the creation of consistent, believable stories (Schmidt, On Writing: Creating Characters and Maintaining Continuity in Writing, 2015) (Schmidt, Maintaining Continuity: Tales from the Copy Editor, 2015) (Lehman, 2018). It is also understood that most, if not all, current efforts to organize stories and characters are completed manually; established author J.K Rowling has publicly stated that documentation for her works exists as pen on paper (Rowling, 2018).

Due to most of the approaches recent studies have taken in abstracting clusters of words rather than trying to directly analyze the structure of language to discern meanings,

this research was more closely inspired by the older study (Wiebe, 1990) and the more rigid but conceptually similar project PATTY (Nakashole, Weikum, & Suchanek, 2012) which are focused on the discrete analysis of language. The idea was to extract specific, explicit details from an understanding of linguistic rules and structure rather than less concrete notions of character relationships or emotions implied by language in arguably subjective categories. The drive was to interpret specific language as in PATTY, identifying certain types of information and intelligently labelling individual sentences based on their subject and whether their primary purpose is to describe something.

Given existing efforts to improve the narrative editing process with digital tools, and the lack of narrative analysis tools focused on the extraction of specific, concrete details from narrative work, this area of automatic narrative analysis as a tool for the creation and analysis of literature is novel and the focus of this study.

2.1 Parts of Speech

Parts of Speech (POS) are categories of words that serve different purposes and form the building blocks of language. Each of the eight main POS in the English language has a purpose and a syntactic role that form a framework for every coherent sentence. While human understanding of language is often subconscious, the understanding of POS allows us to piece together the meanings of sentences. They allow attribution of actions, descriptions, and chronology to the right entities that we might infer the nature of unfamiliar words based on the patterns around them. The study of POS began centuries ago and is not limited to the English Language (Matilal, 2015), and as such will only be discussed here at a basic level to understand how they convey meaning

and why they are an important and useful labelling measure to extract meaning from sentences.

In basic discussions of parts of speech, there are eight major categories:

Nouns (people, places, things) specifying entities and ideas

Pronouns (I, you, we, it) which specify entities differently depending on context

Adjectives (tall, dark, bright, cold) describing entities and ideas

Verbs (run, speak, read, sleep, is) denoting actions or states of being

Adverbs (slowly, softly, hotly, solidly) which act as descriptions of verb actions

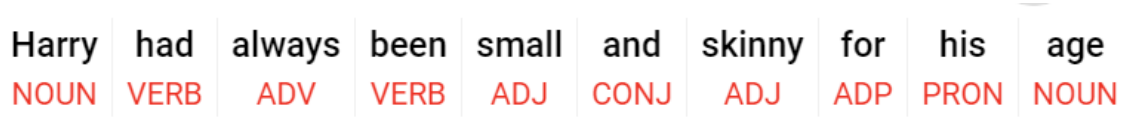
Prepositions (of, like, at, before) relating nouns and pronouns to other words

Conjunctions (and, or, because, so) joining ideas together to show connections

Interjections (Wow! Hey! Oh!) expressing emotions

The classification of inputted words into these categories is called *parts of speech tagging* and is a popular area of research. Patterns in POS are being used as components in the development of applications with a limited understanding of natural language. For example improving summary generation in n-gram based applications by eliminating blocks of text containing only prepositions, adverbs or other less meaningful types of words (Al-sharman & Pivkina, 2018). There also exist commercial applications and services that perform this function using the eight categories above along with more nuanced and specific categories, such as Google Cloud Natural Language API demonstrated in figure 2.1.

Figure 2.1 Example of Parts-of-Speech Tagging



2.2 Syntactic Patterns

While it is understood that the patterns formed by parts of speech (POS) according to grammar are not synonymous with the patterns formed logically in interpreting language and that deriving the logical meaning of a sentence from a grammatical expression is not yet an exact science, the two are strongly related (Szabó, 2015) and examining these patterns can yield usable results.

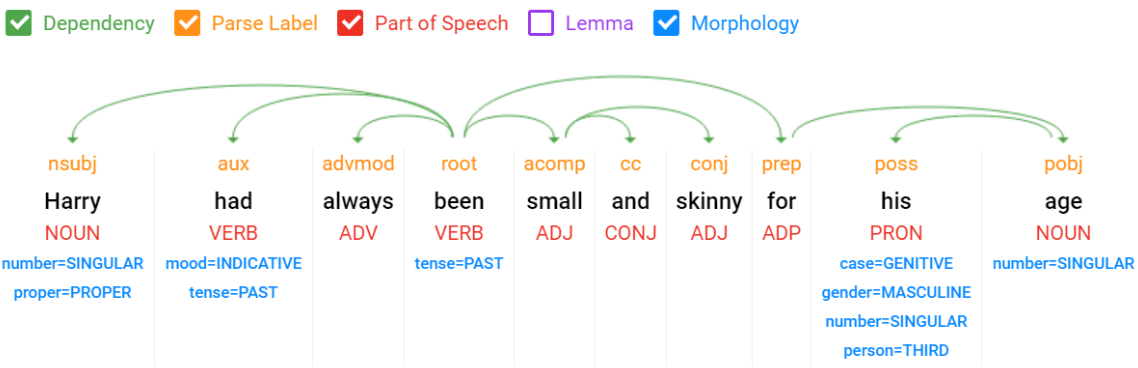
Patterns in natural language are, however, not limited to tagged sets of POS. The nature of certain words – particularly but not limited to conjunctions and prepositions – imply relationships between words. These relationships organize sentences by identifying subjects (the people, objects, ideas or events being discussed) and roots (the purpose of the sentence – describing the subject or explaining the action of a subject, among others).

These semantic and syntactic rules provide more structure to language and have been used to define rigid patterns exploring the relationships between two semantically defined concepts in a project called PATTY (Nakashole, Weikum, & Suchanek, 2012). For example, they defined patterns like "<politican> governor of <state>" where the <politican> identifies a specific individual in order to establish that person is the governor of a state. This is a very inflexible approach that requires scanning and labelling a large corpus to find exact phrases where the subject (the thing being spoken about, the

politician in the above example) and the object (the state in the example) can be removed to treat the rest of the text as a complete pattern.

The concept does form the basis of the idea for this research, however. Inspired by the incredibly specific patterns of PATTY, this research uses more aspects of syntactic labelling - including the variety of features visualized in figure 2.2 – to create a more flexible system to identify less specific instances of patterns and derive their meanings. Rather than explicitly looking for a word like "governor" modern language processing systems are able to determine when a noun is applied to a subject – meaning that a pattern of "<proper noun> <noun> <preposition> <noun>" could be processed instead, to a much larger array of meanings and with much less manual training overhead.

Figure 2.2 Multiple Features of Syntactic Analysis



2.3 Semantic Meaning and Synonyms

Semantic meaning is the actual meaning behind a word. For example, the word "skinny" in English is used as an adjective to describe a slim body type, a low-fat type of food, and minimalist margins (as in finance) or as a noun to describe information (to get the skinny on a celebrity relationship). The meaning of the word depends on its part of

speech (noun, adjective, verb), which is inferred based on the syntax of the sentence around it, and context, which can often be hard to determine. In this experiment, however, the context is limited to sentences based on syntactic patterns that are associated with descriptions about characters. More importantly, the semantic meanings of words are not often unique. When there exist multiple words to describe the same thing, they are a recognized feature in linguistics called synonyms – while the underlying idea called a lemma. When referring a slim body type, one might use the word “skinny” – or the synonyms “boney”, “scrawny”, “unweight”, or “weedy” – and these would indicate (roughly) the same thing.

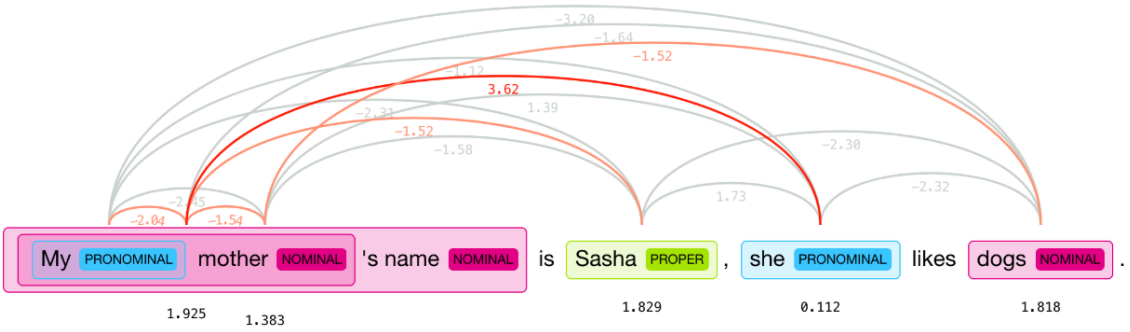
Humans understand semantic meaning by memory. We are taught the meaning of words and infer the meanings of others by how similar they are to words we know and by the context surrounding them. In NLP, a lexical reference system – sometimes called a dictionary or database of terms – such as WordNet (Princeton University, 2010) may be used to identify the meanings of words and disambiguate words or identify multiple instances of the same idea. Combining lemmas with synonyms reduces a wide variation of words into their core meaning for disambiguation and more robust matching against dictionaries of specific terms.

2.4 NLP Tools – spaCy and NeuralCoref

spaCy (spaCy, n.d.), an open-source NLP library was used in this project. spaCy provides tools for syntactic analysis, including named entity recognition, determination of lemmas, labelling sentences by parts-of-speech, determining dependencies of words, splitting text into sentences, and matching sequences of words to syntax-based patterns.

spaCy doesn't include any tools for resolving coreferents – where an entity is referred to by any words or terms other than their names - but there is an extension for spaCy called NeuralCoref (Wolf, 2017) which uses a trained neural network to predict the most likely entity. There is a visual example of NeuralCoref output in figure 2.3

Figure 2.3 Coreferent Resolution with NeuralCoref



2.5 Metrics

Both algorithms are ultimately classifiers, and there are two concrete questions that determine if their output is accurate. The description identifier algorithm is assessed based on whether the extracted information is accurately labelled as a descriptor (yes, it is a description or no, it is not a description). The subject identifier algorithm is assessed based on whether it has extracted the exact subject of the sentence (yes, it is the correct subject or no, it is some other subject). These questions were answered by manual verification of program output for each sentence. The overall effectiveness of the algorithms was assessed over multiple results from descriptions collected across multiple novels by using a confusion matrix, a common means of evaluation in classifer algorithms (Fawcett, 2006) based on the following:

True Positive (TP) – labelled as positive (yes) correctly

True Negative (TN) – labelled as negative (no) correctly

False Positive (FP) – labelled as positive (yes) incorrectly

False Negative (FN) – labelled as negative (no) incorrectly

The above are assessed based on the relationship between the output of the algorithms and the manual values assigned as a human interpretation of the data, offering the following four categories:

Actually Positive (AP) – human identified as positive

Actually Negative (AN) – human identified as negative

Labelled Positive (LP) – algorithm labelled as positive

Labelled Negative (LN) – algorithm labelled as negative

These results are visualized as follows for a single-class (one question) classifier, where ***n*** is the number of results examined and each cell corresponds to the labelled and actual conditions in its row and column. The values represented in figure 2.4 and all other values in section 2.5 are included as an example, and not representative of any results obtained for the purposes of this study.

Figure 2.4 Example Confusion Matrix

n = 100	**Labelled Positive (LP)**	**Labelled Negative (LN)**
Actually Positive (AP)	20 (TP)	30 (FN)

Actually Negative (AN)	10 (FP)	40 (TN)

These four results are used to calculate more meaningful values often used to evaluate the effectiveness of NLP applications: *recall, precision, f-score, accuracy,* and the *null accuracy rate (NAR).*

Recall is a measure of how many times a positive result was labelled correctly. For example, if someone were to look through old yearbook photos, recall would consider how many people that person recognized out of the set the people they had met. If that person knew 100 people in the book, but only recognized 50, then their recall rate would be 50/100 or 0.5. The general formula is: $\frac{TP}{AP}$

Precision is a measure of how often a positively identified result was correctly identified as a positive result. To continue the yearbook photo example, if the person thought they recognized 10 people in the yearbook, but had only met 8 of the people they had listed, their precision would be 8/10 or 0.8. The general formula is: $\frac{TP}{LP}$

The **F-score** is a weighted average of recall and precision, used to average the correctness of two metrics, providing a more general measure of performance when the other metrics may not agree. In examining their yearbook photos, someone could identify only one person they had known in a class of 100 people. They would be perfectly precise (1) but their recall would be very low (0.01). The F-score is calculated by the following formula: $2\left(\frac{precision*recall}{precision+recall}\right)$ (Powers, 2011)

Accuracy measures how often, overall, that a correct result occurs – whether it is a true positive or true negative, whether a person recognize someone they know or identify someone they don't. It is found by the following formula: $\frac{TP+TN}{n}$

The **null accuracy rate** (NAR) measures how often it would be right if the more likely option were always selected. For example, if the yearbook was for a school of 1000 people and it was known that someone had only ever met 50 of them, but had no idea who, how often would they be wrong if they chose the more likely option (that they hadn't met them) for every picture they saw? The formula is $\frac{max}{n}$ where ***max*** represents the number of instances of the more likely class – meaning in our example, that person's null accuracy rate would be $\frac{950}{1000}$ or 0.95 rate of correct labels.

The NAR can be compared to accuracy to determine if the classifier is more effective than always assuming the more likely choice. For example, if there is an accuracy of 0.5 and NAR of 0.9, classification using the algorithm is correct 50% of the time. However, assuming the more likely choice is right 90% of the time - meaning it is more accurate to assume the likely choice than to use the algorithm to determine classification.

The f-score and the relationship between NAR and accuracy show the effectiveness of the algorithm overall, while the other metrics provide more specific information about the strengths and weaknesses of the classifier.

CHAPTER THREE

3. METHODOLOGY

This chapter introduces the research methods used, including discovering patterns in descriptive text, the steps of the proposed system to identify descriptions and descriptive subjects, the testing strategy to evaluate the system functionality and the complexity of the system.

3.1 Discovering patterns in descriptive text

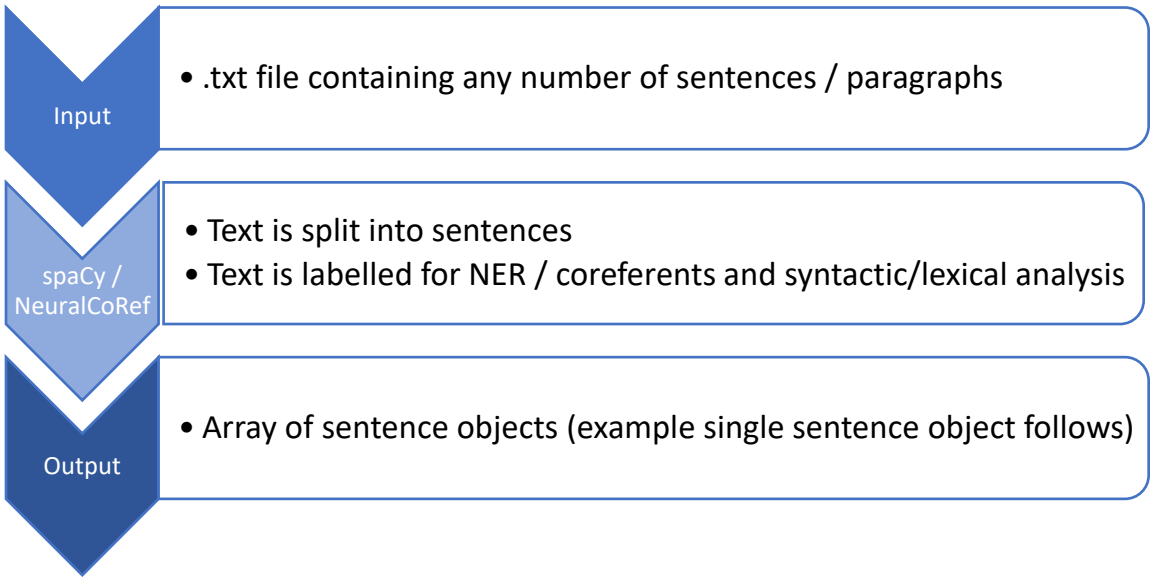
In order to discover patterns in descriptive text, existing descriptive text had to be analyzed. Initially, it was thought that examining a certain number of paragraphs from multiple sources of text would provide sufficient examples, though upon on examination of this text multiple issues became apparent. Paragraphs vary significantly in length, in particular between narrative works of different styles; the first five paragraphs of Harry Potter and the Philosopher's Stone contain 22 sentences across approximately 327 words, while the first five paragraphs of The Hobbit contain 27 sentences across approximately 802 words. By this paper's definition of a descriptive sentence, the first five paragraphs of Harry Potter contain only two sentences that could be considered descriptions of a character – and the Hobbit contains three. These numbers were not significantly improved over larger sample sizes, and so the content examined in order to find patterns was not constrained to specific works of narrative text, expanding instead to include general descriptive sentences and patterns from general reading, conversation, and other sources. Overall 593 sentences were used in training, including 50 artificial descriptive sentences that did not originate in published narrative works.

The above data was run through spaCy and neuralCoref before being examined for commonalities in the structure and labels of the sentences in question. Two algorithms were written to identify the presence of patterns found through this analysis, and refined by using variations with the same core features.

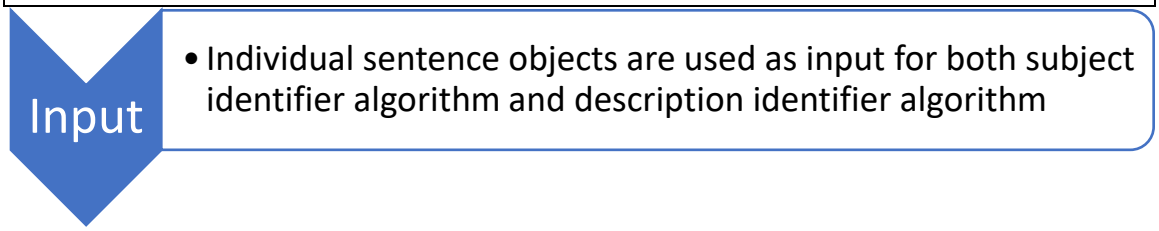
3.2 System Steps

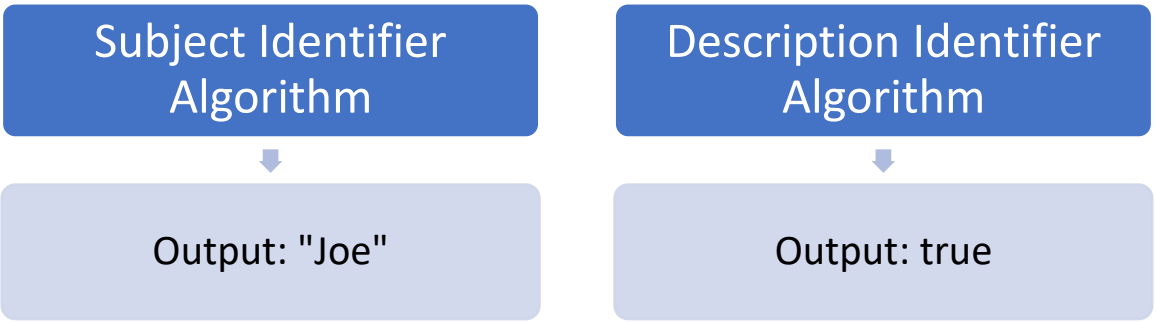
This research involved 3 things: preprocessing data with spaCy and NeuralCoref, identifying subjects with the Subject Identifier Algorithm, and identifying descriptions with the Description Identifier Algorithm. Figure 3.1 is a walkthrough of input/output through the system used, with an example sentence object provided.

Figure 3.1 System walkthrough



```
text: “Joe was tall”
root
   .text: “was”
   .lemma: “be”
   .children:
     [
         token[0]
              .text: “Joe”
              .dep: “nsubj”
              .lemma: “Joe”
              .pos: “PROPN”
              .ent_type: “PERSON”
              .in_coref: “false”
              .coref_clusters: []
              .children: []

         token[1]
               .text: “tall”
               .dep: “acomp”
               .lemma: “tall”
               .pos: “ADJ”
               .ent_type: “”
               .in_coref: “false”
               .coref_clusters: []
               .children: []
     ]
```





3.2.1 Preprocessing

The data examined by the system is exclusively narrative text and must be sanitized to some degree. The format of electronic text is inconsistent across eBooks and other electronic sources, and narrative text often includes decorative variations of certain typographic characters (such as ‘curly quotes’ – directional quotation marks “ ” as opposed to the neutral quotation familiar in software development " or long dashes) that were inconsistently interpreted by spaCy’s (spaCy, n.d.) English language model in delimiting sentences. In order to remove these inconsistencies, special typographic characters were replaced by their neutral counterparts.

For the most part, the system examines individual sentences to determine their meaning – and spaCy is capable of splitting text into sentences itself - but for the purposes of pronoun or coreferent resolution (section 3.2.2.4) text is input in the form of sequential paragraphs, and each source is processed in isolation as its own file. The text is analyzed using spaCy to add the following layers of information for use in later steps: lemma, parts of speech, and dependency relationships (which include parse labels). The patterns used by both algorithms rely most heavily on dependency labels and the dependency tree, which expresses the relationships between words in a sentence, while the other labels are used to support those patterns.

3.2.2 Identifying Subjects

The value of identifying descriptive sentences in narrative work is the potential to examine all descriptions for each character. In order to attribute any sentence to a particular individual, the sentence subject must be identified; this is the purpose of the subject identifier algorithm.

Figure 3.2 Subject Identifier Algorithm

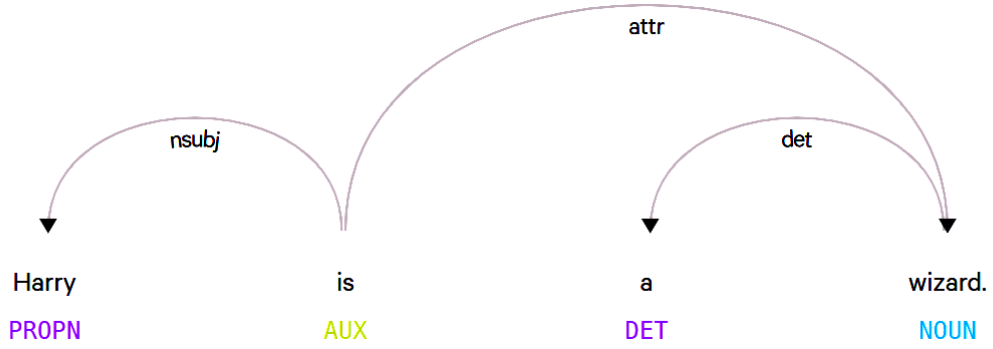
```
sentence = someSentenceObject #Per figure 3.1
aspects = [ "face", "hair", "height" ...] #Set per section 3.2.2.2

subjectIdentifier(sentence) : string
    foundSubject = null
    foreach token in sentence.root.children
        if token.dep == "nsubj"
            if token.pos == "PROPN" or token.ent_type == "PERSON"
                foundSubject = token;
            else
                foreach subtoken in token.children
                    if subtoken.dep == "poss"
                        synsets = token.wordnet.synsets()
                        if intersection(synsets,aspects).size > 0
                            foundSubject = subtoken
                        else
                            foundSubject = token
                        break;

            if foundSubject != null and foundSubject.in_coref
                return foundSubject.coref_clusters[0].text
            else if foundSubject != null
                return foundSubject.text

    return "No Subject"
```

Three techniques have been applied to make this determination, in the following order.

3.2.2.1 Sentence Subject Recognition

The subject of a sentence is not always a character – it may be a book, a rock, or in this case the subject of a sentence itself. The subject of a sentence is easily determined using a lexical dependency tree, as represented by the overarching arrows in figure 3.3

Figure 3.3 Example sentence analysis for subject recognition

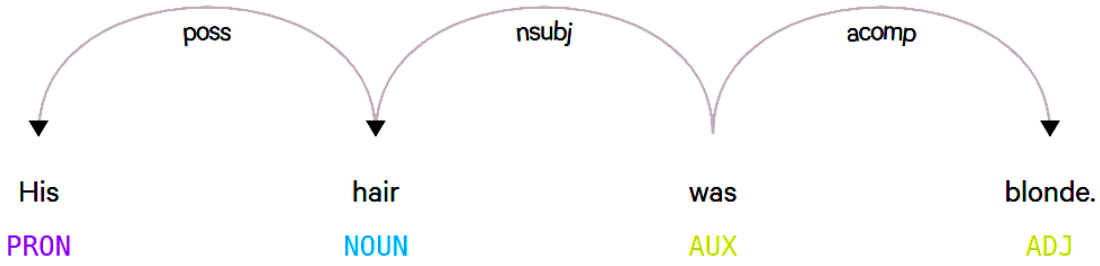


In this sentence, *'is'* is the root word, and *'Harry'* is the noun-subject (nsubj) of the root word – making him the subject of the sentence. If the subject is a proper noun, it refers to a specific individual (usually a person or a location) and the subject has been identified. In many cases, the subject is a coreferent term like *'he'* or an aspect of someone like *'his hair'*, meaning further examination is necessary. This first step is important in understanding aspect recognition however.

3.2.2.2 Aspect Recognition

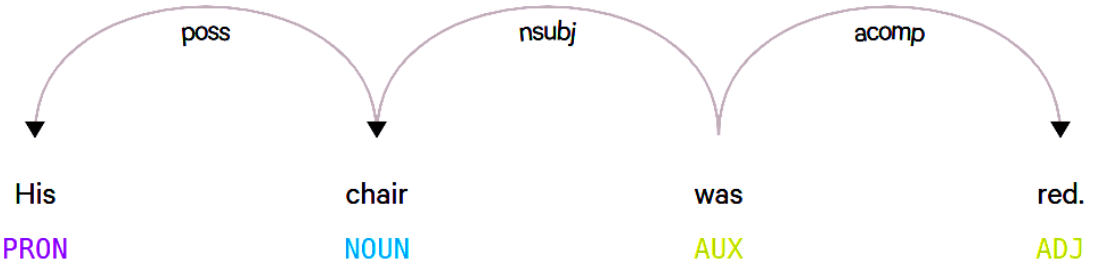
If the subject of a sentence is not a proper noun, it may still be a noun – a thing, which may be a part of a character. For example, in the sentence "His hair was blonde" we are describing '*him'* – specifically '*his hair'*, which is an aspect of his appearance. Aspect recognition is the process of determining if an object is a part of a character, and is a technique developed during this process and specific to this application. The patterns denoting aspects are possessive – "*His* hair was blonde" describes something that belongs to him, even though it's a part of his body. When labelled for analysis, the dependency tree appears as in figure 3.4

Figure 3.4 Example sentence analysis for aspect recognition



Notice that the subject of the parse tree (*hair*) is the noun directly linked to the root (*was*). Therefore, we know the sentence is about *hair*, but not who it belongs to. Following the next dependency link (labelled *poss*, for possessive modifier) explains that the hair belongs to a pronoun (*his*) which is a reference to a person. Intuitively, this pattern makes sense: the description is of '*his hair'*, which is a part of *'his'* body, and therefore it is a description of '*him'*. The pattern is deceptively simple, however, because the sentence in figure 3.5 appears identical if dependencies and parts of speech are the only layers of information in use.

Figure 3.5 Example sentence analysis for complications in aspect recognition



The purpose of this research is to identify sentences that are descriptions of characters, meaning a difference must be established between these two types of cases.

The distinguishing factor is what exactly the possessed object is – specifically whether it is a part of a person's body. In order to make that determination, the system contains a dictionary of terms used to refer to the body and various aspects of it. The subject identifier algorithm makes use of synsets (sets of synonyms) generated by WordNet, finding all synonyms for the object identified and comparing that list to the dictionary of terms. If the word is, or is synonymous with, a word in the dictionary, it is considered a part of the body. In that case, the subject of the sentence is whoever owns the object rather than the object by itself.

3.2.2.3 Named Entity Recognition

Named Entity Recognition (NER) is an NLP process by which certain entities may be identified (Sekine & Nadeau, 2017). spaCy provides tools for NER by which people specifically may be identified. In the example sentence, "Harry had always been small and skinny for his age." NER identifies Harry as a named entity under the label 'person,' which is effective for differentiating between subjects which are people and subjects which are places or organizations, all of which go by proper nouns.

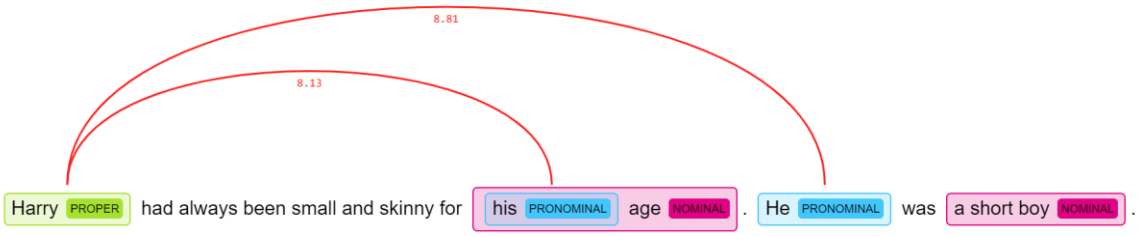
The application of this technique is limited, however, by the use of pronouns. For example, if we had the text "Harry had always been small and skinny for his age. He was a short boy." NER may be able to identify Harry in the first sentence, but he is never explicitly mentioned in the second. Intuitively, humans understand that Harry is the 'he' being discussed. More information is required to algorithmically understand this.

3.2.2.4 Coreferent Resolution

Pronoun or Coreferent Resolution is the process of examining a pronoun or other reference to a character that is not their primary name (a coreferent) and determining to

whom it refers. If the subject of a sentence is a person as identified by a pronoun, this technique may be used to find out who the subject is. Techniques for manual extraction using lexical parse trees have existed for decades (Hobbs, 1977) but the tool used in this case is NeuralCoref (Wolf, 2017) NeuralCoref is a neural network coreference resolution system for spaCy. Given the sentences in the previous example, NeuralCoref is able to determine who ‘he’ is in the demo visualization in figure 3.6

Figure 3.6 NeuralCoref Resolution



Coreferent resolution is limited by the set of coreferents used throughout the input text. If Harry’s name were never used in the example, it would be possible to determine that each ‘he’ referred to the same person – but not whom it was, by name. Coreferent resolution also uses a limited range of words around the referent term in question. Therefore the benefits of using it may be limited when applying this technique to longer text samples.

3.2.3 Identifying Descriptions

The purpose of the second algorithm is to determine whether a sentence is a description or not.

Figure 3.7 Description Identifier Algorithm

```
sentence = someSentenceObject #Per figure 3.1
aspects = [ "face", "hair", "height" ...] #Set per section 3.2.2.2

descriptionIdentifier(sentence) : boolean
    if sentence.root.lemma == "be"
        foreach token in sentence.root.children
            if token.dep in ["acomp", "attr", "prep"]
                return true

    if sentence.root.lemma == "have"
        foreach token in sentence.root.children
            if token.dep == "dobj"
                synsets = token.wordnet.synsets()
                if intersection(synsets, aspects) > 0
                    return true
    return false
```

The algorithm examines the root word of a sentence and operates as an expert system, examining the labels on a sentence and comparing them to the patterns described from 3.2.3.2 – 3.2.3.5. These details are used to determine if the sentence contains a description according to the definition in section 3.2.2.2.

3.2.3.1 Definition of a Descriptive Sentence

For the purposes of this classifier, a descriptive sentence is a description whose core and explicit purpose is to present a description of a person or object. This includes descriptions of specific aspects or features of a person or object (someone's hair, the surface of a table) but not actions associated with them (the way someone walks, the activity they are currently performing.) This means that incidental descriptions that form a part of other sentences do not qualify the entire sentence of which they are a part as a description. In human understanding, there is rarely a conscious distinction between these

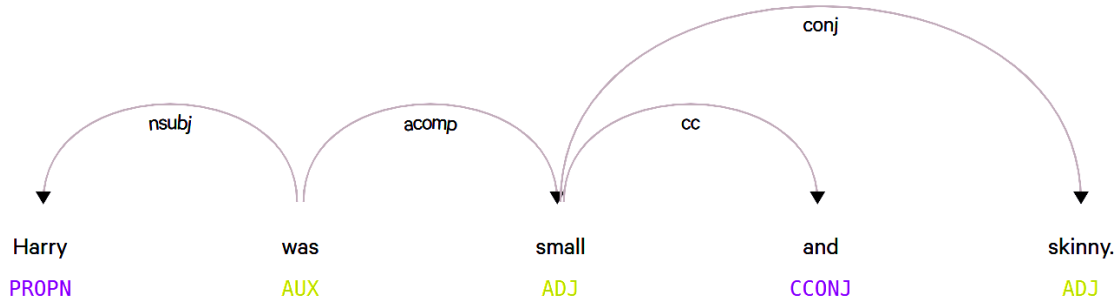
types of sentences, as the same idea may be conveyed through different types of sentences. For example: “All of the women in town thought she was beautiful” is not a sentence about the beauty of the subject in question, it is a sentence about what all of the women in town thought – a detail which gives the reader context and information, but in a non-explicit way that does not meet our definition. “She raised her thin hands above her head” is a sentence expressing that she raised her hands – the ‘*thin*’ descriptor is incidental in this case. “She had thin hands raised above her head” is a description of her state at the time – that her hands were raised – which is a description by this definition.

Patterns within this definition were examined for the purposes of this project, and are explained below along with the process by which they are algorithmically identified in code. The names of the patterns are terms used for this project and not reflective of any broader linguistic categorization system.

3.2.3.2 Basic Adjective Description

The most basic form of description in the English language is literal. “Joe is tall,” “Anna is fast,” “the sky is blue” – these sentences follow a simple pattern that is easy to detect, even when expanded to include multiple details as in figure 3.8

Figure 3.8 Adjective description pattern example



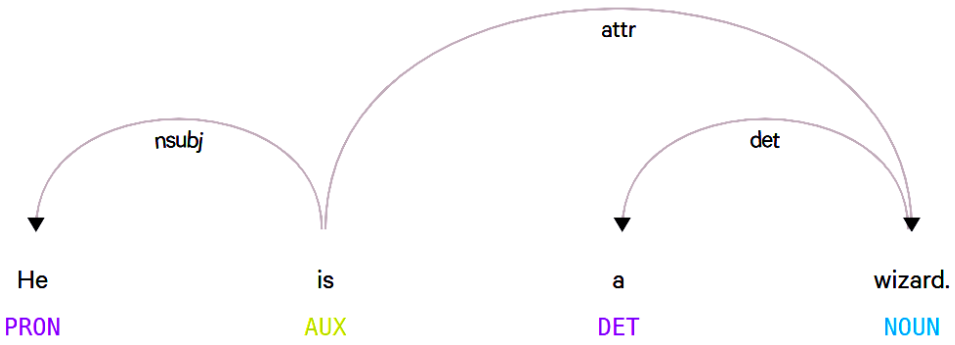
This pattern, as well as the others that follow, begins with the root word. Here the word is 'was'. Using a tool called a *lemmatizer* in spaCy, the algorithm finds the root version of the word – which is '*be*', a lemma which will identify all possible conjugations of a verb (for "be" variations include is, am, are, were, was) to allow the pattern to match the root regardless of tense (past, present, future) and subject (singular, plural, self as in I or we). As the sentence describes the subject, "was" is the root, connecting via dependency the subject and the descriptive terms - in this example an adjectival complement (*acomp*), which is a descriptive adjective. By programmatically navigating the dependency tree, the algorithm is able to check these features against our pattern.

In summary, this pattern checks for two things: is the root word a lemma of '*be*' and does the root word have a direct dependency that is an adjectival complement?

3.2.3.3 Attribute Description

An attribute description is similar to an adjective description in that it has "was" as a root word, but instead of offering information in the form of an adjective (acomp) it is presented through an attribute (attr) as in figure 3.9

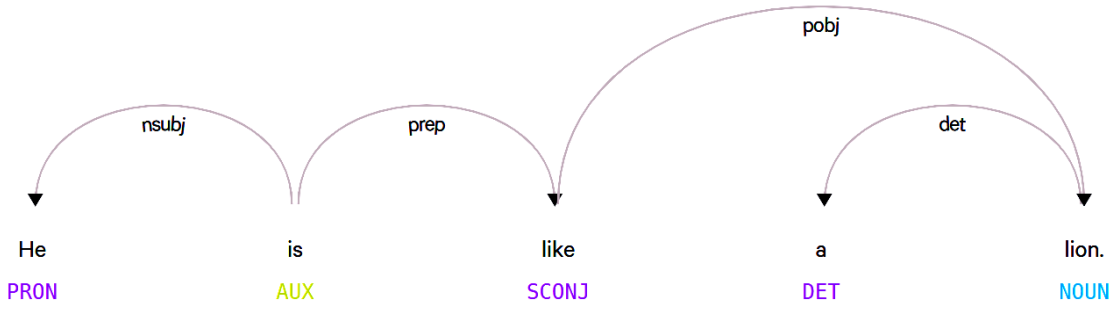
Figure 3.9 Attribute description pattern example



3.2.3.4 Comparative Description

A comparative description offers information about one thing by comparing it to another. It also uses a '*be*' root, but via dependency it connects the subject to a preposition (*prep*; like, beside, above, past, by, near, from) and then to the prepositional object (*pobj*) which it's being compared with. One example is demonstrated in figure 3.10

Figure 3.10 Comparative description pattern example

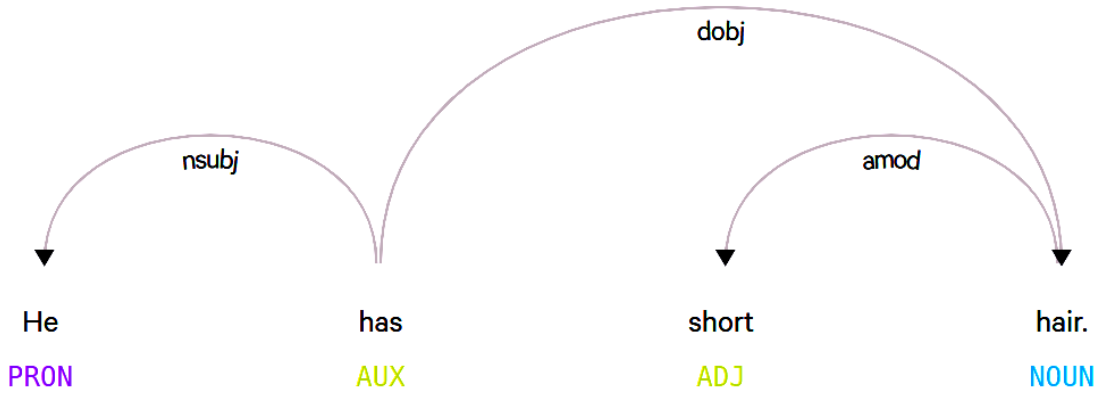


3.2.3.5 Descriptive Possession

One common form of description offers information about a person based on aspects of their appearance. By aspects of a person's appearance, we refer to the same

aspects in section 3.2.2.2 – parts or features of an individual that are intrinsically connected with their owner (as in a body). Consider the example in figure 3.11

Figure 3.11 Descriptive possession pattern example



This is the only identified pattern that doesn't use the '*be*' root word. Instead, it uses the possessive '*has*' (has, had, have) and connects the subject with a direct object (*dobj* – the thing/aspect they have), which may or may not have an adjective to describe it. For instance, the sentence "He has hair" presents information without necessarily adding the word 'short'. This pattern has the same concern as the aspect recognition problem in 3.2.2.2 – a direct object can be any noun, therefore we use the same technique of finding every synonym for the object and comparing them against the same list of terms to ensure the description is about a body and not a piece of clothing or other possession.

In summary: the algorithm checks if the root word is of the lemma 'has', whether it has a direct dependency of the type 'dobj'. If it does, the algorithm requests a list of all synonyms for the lemma of the dobj and compares those against the provided dictionary of terms relating to the body. If there is a match, the sentence is a description.

3.3 Testing Strategy

The testing strategy is to evaluate the performance of both algorithms in the system in identifying descriptions and descriptive subjects using narrative text that was not used to discover the descriptive patterns.

3.3.1 Testing data

Test data was collected after the development of the system in the form of paragraphs of character descriptions from various narrative texts in order to allow for an unbiased examination of the resulting system. From each source, one or two passages that contained descriptions were selected, and the text containing the descriptions as well as a few preceding/following sentences were selected for test data. The data was sourced from the following works:

Table 2. Novels used in testing data

Title	Author	Year
A Hitchhiker's Guide to the Galaxy	Douglas Adams	1979
A Separate Peace	John Knowles	1959
Ender's Game	Orson Scott Card	1985
Harry Potter and the Philosopher's Stone	J.K. Rowling	1997
Holes	Louis Sachar	1998
I Know Why the Caged Bird Sings	Maya Angelou	1969
Lord of the Rings: The Fellowship of the Ring	J.R.R. Tolkien	1954
No Longer a Stranger	Joan Johnston	2005
Soulless	Gail Carriger	2009
The Hunger Games	Suzanne Collins	2008
The Neon Rain	James Lee Burke	1987
The Orphan and the Thief	M.L. LeGette	2013
Twilight	Stephanie Meyer	2005

These works span in publication from 1959 to 2013, offering a 54 year variation in literary trends and style. They vary between books written for younger audiences such

as *Harry Potter and the Philosopher's Stone,* and *Holes* to those written for older audiences like *The Neon Rain* and *I Know Why The Caged Bird Sings.* Some texts, like *I Know Why the Caged Bird Sings, The Neon Rain* and *A Separate Peace* are in first person, while others are in third person. There are also differences in written style, as *The Lord of the Rings: The Fellowship of the Ring* and *Soulless* both make use of unusually 'flowery' or 'decorative' ways of speaking and narrating, *A Hitchiker's Guide to the Galaxy* has a non-character narrator with opinions and personality, and other titles are more neutral in their narrative voice. The titles also span genres between high fantasy, science fiction, dystopian, and romance. The wide variation in style and genre offer a small but reasonably representative data set.

The test data was labelled after human detection of a descriptive passage, meaning that while all selected text contained descriptions, not all of those descriptions met the specific description of a descriptive sentence outlined in section 3.2.3.1. Of the final selected text, there were 33 descriptive sentences and 89 non-descriptive sentences after manual assessment, for a total of 122 sentences.

Furthermore, while the raw text was selected in the form of full sentences, the 'sentences' as counted in the process of testing (and referred to above) were sentences as detected by spaCy. spaCy's '*sentencsizer'* splits text consistently but does not always produce the same 'sentences' a human would interpret in written text. For instance, it tends to consider dialogue as a sentence of its own, '"Hello!" he said.' might be broken into '"Hello!"' and 'he said.' Separately. In order to have consistent data for comparison, spaCy was allowed to split the sentences before they were manually labelled – including some sentence fragments or unintroduced dialogue as testing entries.

3.3.2 Metrics

The two algorithms developed in for this research are functionally independent of one another and were evaluated separately/

3.3.2.1 Evaluating the Subject Identifier Algorithm

In identifying the subjects of sentences, the subject identifier algorithm could output one of two things: a potential subject (which could be the direct subject of a sentence or a co-referent) or "No Subject" if no subject was detected. Evaluating the correctness of these labels is more complex than evaluating the description identifier, as there are more than two possible responses, and the correctness of the overall result cannot be evaluated exactly the same way in every case.

Despite efforts to capture text around known descriptions in sample data, not all sample data contains proper names for all subjects. While some passages clearly name characters, which may then be subjected to a coreferent like 'he' or 'she' that the system may use to label each sentence by the proper name, other passages never refer to some characters by name and as a result present 'he' or 'she' as the subject by itself – which is not incorrect, given the amount of information at hand. Furthermore, there are some instances where a character has multiple coreferents but the system was not able to resolve them, and therefore some sentences belong to the character by name, and other sentences (regarding the same character) are attributed to their coreferent – which is still correct, if suboptimal. For the purposes of this evaluation, subjects are considered correct if they are any clear coreferent of the appropriate character. If someone is called by "he" and "joe" in a certain passage, a sentence about this person is correct under either label, assuming that character is the only one in the passage called "he". If there are two people

called by “he” in a passage who are also called by other names, because it would be unclear to which “he” the sentence is attributed, it is considered incorrect as a subject. Some sentence subjects are not characters, which can be correct, and some sentences (notably the sentence fragments discussed in 3.3.1) do not have subjects.

This creates three categories of result: the correct subject, an incorrect subject, and no subject at all. Subject identification was measured by using these three metrics in a confusion matrix with the metrics explained in section 2.5 – with the exception of NAR. The subject of each sentence was different and the test data was processed as a series of small discrete files. Given there is no true, continuous, dominant class, NAR would not be helpful.

Accuracy assesses the effectiveness of the system on the particular test data used, while recall, precision, and F-score are more representative of its overall effectiveness. Recall for the subject identifier represents how often sentences with a subject were identified correctly. Precision represents the percentage of subjects correctly labelled out of all sentences which contained subjects. The F-score is a mixed representation meant to weigh the benefits of both recall and precision together.

3.3.2.2 Evaluating the Description Identifier

In identifying descriptions, the description identifier algorithm presents a true-or-false label – either a sentence is a description, or it isn’t. The algorithm can, therefore, be considered a binary classifier, and every sentence in the test data is labelled in this way by the system and manually according to the precise definition of a description as presented in section 3.2.3.1. There are descriptions of other types that do not match our definition in the test data, but as testing labels were made to adhere to the given definition

a failure to recognize these alternative descriptions is not currently considered by the system (result set A). An alternative analysis based on a more human interpretation of a description is provided (result set B) to consider the results outside of the scope of the description patterns sought out for this research. The limitations of this system are discussed in the later Findings chapter.

The true/false labels were applied to a confusion matrix as described in section 2.5 and figure 2.4, using the metrics there described: recall, precision, F-Score, accuracy, and also NAR. The meaning is similar for the description identifier as for the subject identifier, but as it is truly a binary classifier they are not perfectly identical. Recall for the description identifier represents how often sentences which were actually descriptions were identified as descriptions by the system. Precision represents the percentage of sentences the system identified as descriptions which were correctly identified as descriptions. The F-score is a mixed representation meant to weigh the benefits of both recall and precision together.

Accuracy represents the effectiveness of the system on the particular data used for this test, while the NAR is the expected accuracy if one were to assume every sentence were the more likely case. The majority of the test sentences were not descriptions, so the NAR represents how often one would be correct if we assumed none of the sentences were descriptions, in order to compare the result of the system against an unstructured approach or pure chance.

3.4 Complexity of the system

The system relies on data labelled with the following information: POS tags, dependency relationships (including sentence root word) and lemma – which are provided in this implementation by spaCy. It also uses coreferent data provided by NeuralCoref through the spaCy pipeline. The precise time complexity of these systems could not be determined based on their current documentation. However, these systems extract the initial details required for this system to function, but the data they provide is not unique to these systems (which is to say, another NLP system could be used with a different runtime) and therefore they aren't intrinsically tied to the speed of the system developed here.

Excluding the time required for the aforementioned labelling, the system developed for this research runs in $\boldsymbol{O(n^2 + n)}$ time, where $\boldsymbol{n}$ is the number of words in a sentence. By themselves, the subject identifier algorithm runs in $\boldsymbol{O(n^2)}$ time and the description identifier algorithm runs in $\boldsymbol{O(n)}$ time. For context, the average number of words in a sentence ($\boldsymbol{n}$) in English language writing is approximately 25-30 in scientific literature (Moore, 2011) and less in prose (Vigen, n.d.).

CHAPTER FOUR

4. FINDINGS

4.1 Subject Identification

Figure 4.1 Confusion Matrix for Subject Identifier

n = 122	**Labelled Correct Subject (LC)**	**Labelled Other Subject (LW)**	**Labelled No Subject (LN)**	
Actual Correct Subject (AC)	89 (TP)	13 (FP)	10 (FN)	112
Actual No Subject (LN)		2 (FP)	8 (TN)	10
	89	15	18	

Table 3 Results for Subject Identifier

Recall	0.795
Precision	0.856
F-Score	0.824
Accuracy	0.795

The subject identifier algorithm correctly identified the subjects of approximately 80% of the sentences in the test data, which is in line with its F-score of 0.824. It scored more effectively on precision (0.856) than recall (0.795), meaning that when the system labelled a sentence with a subject, it was often correct – but that it also indicated a number of false positives. It incorrectly identified 13 subjects and incorrectly determined there to be no subject in 10 cases, indicating that the likelihood of choosing the wrong subject was similar to the likelihood of erroneously finding no subject in a sentence at all.

A close examination of the sentences where the subject identifier failed suggested the following types of issues.

4.1.1 Sentence Fragments and Dialogue

The subject identifier struggles with sentence fragments and dialogue. This is understandable, given that sentence fragments are by definition incomplete sentences that may not contain full ideas or complete structure for analysis, and dialogue is representative of spoken word which is often less formal and structured than narrative text.

Given the purpose of the system is to extract information from the narrative text as a source of explicit fact, removing dialogue text – which represents the words of a character and not the facts of the story - from consideration would have no negative impact. Sentence fragments are difficult to avoid or process by nature and exist unpredictably in narrative text – although some of the sentence fragments produced were a result of the way spaCy split the input text into sentences. Using alternative means to split text to reflect full sentences might result in fewer sentence fragments and thus fewer mistaken subjects.

4.1.2 Compound Sentences

Other incorrect classifications existed in the opposite situation. Sentences that expressed multiple complete ideas about different characters or different features of characters by combining phrases were frequently misidentified. For example, from *Harry Potter and the Philosopher's Stone*: "His face was almost completely hidden by a long, shaggy mane of hair and a wild, tangled beard, but you could make out his eyes, glinting like black beetles under all the hair." Or from *Lord of the Rings: The Fellowship of the*

Ring: "His hair was dark as the shadows of twilight, and upon it was set a circlet of silver; his eyes were grey as a clear evening, and in them was a light like the light of stars." In the first example the system selected no subject despite a number of aspect terms, and in the second it attributed the sentence to another character, seemingly because both mentioned eyes (they were picked up as a coreferent) but one of the characters was not identified by name, and thus possibly not considered as a separate entity.

It is possible that the dependency relationships that connect ideas in these longer sentences produce different patterns. However, given the number of different attributes described in each sentence, it's possible that attempting to determine one subject given the entire sentence is less effective in these cases and some means should be devised in order to split such long sentences into phrases to be considered separately. The algorithm as it was developed looks for the first subject at the highest level of the dependency tree, but it is possible that these sentences must consider more than one subject.

4.1.3 Contextual Information

The final source of confusion appeared to be short passages with two characters of the same gender where one is never referred to by name. The quote in 4.2.1 from *The Lord of the Rings* presents one such example, where one character was mistakenly labelled as another because they had no unique coreferent terms and were only called "he" – which was also a coreferent of someone else in the scene. A passage selected from *No Longer A Stranger* provides another. It describes a girl named Reb and her resemblance to her sister, who is not named in the selected grouping of sentences. All but one of the sentences in the passage refer either to 'Reb' or 'She' as the subject – and in fact, all sentences refer to Reb, but the coreferent resolution did not resolve any sentences

where Reb was not referred to by her proper name. It seems likely this is the result of the other female referent (the sister) which NeuralCoref may not have been able to resolve without more information.

4.2 Description Identification

Figure 4.2 Confusion Matrix for Given Definition of a Description (Result A)

n = 122	**Labelled Description (LP)**	**Labelled Negative (LN)**	
Actual Description (AP)	27 (TP)	7 (FN)	34
Actual Negative (AN)	6 (FP)	82 (TN)	88
	33	89	

Figure 4.3 Confusion Matrix for Human Interpretation of a Description (Result B)

n = 122	**Labelled Description (LP)**	**Labelled Negative (LN)**	
Actual Description (AP)	29 (TP)	18 (FN)	47
Actual Negative (AN)	4 (FP)	71 (TN)	75
	33	89	

Table 4. Results for Description Identifier

	Result Set A	**Result Set B**

Recall	0.794	0.617
Precision	0.818	0.879
F-Score	0.806	0.725
Accuracy	0.893	0.820
NAR	0.721	0.615

4.2.1 Analysis of Result Set A

Result set A uses data labelled strictly according to the given definition of a descriptive sentence, in order to establish the effectiveness of the system in identifying precisely those types of descriptive sentences.

4.2.1.1 Precision and Recall

The description identifier algorithm had a recall score of 0.794, suggesting that it detects approximately 4/5 descriptive sentences, and a similar precision of 0.818, suggesting that 4/5 of the sentences it labels as descriptions are true descriptions with about 1/5 being false positives. The F-Score meant to weight both kinds of performance is 0.806, which doesn't impart much of significance because the two values are already similar.

4.2.1.2 Apparent Verbs

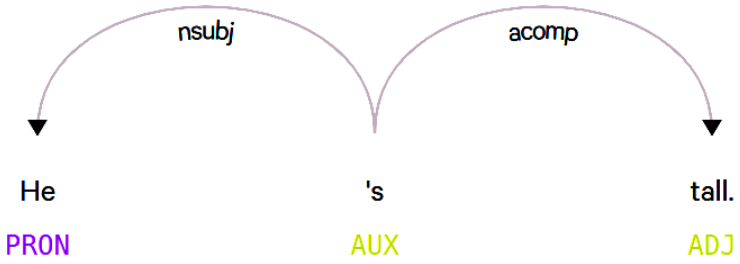
Something not accounted for in the patterns but which appears in many of the incorrectly labelled sentences are what we will refer to as 'apparent verbs'. The root word in all of the patterns used in the testing system is either "has" or "is", which are absolute verbs by themselves. However, in the same patterns, they can be replaced by certain other verbs intended to mean almost the same thing. "Joe is tall" is an absolute description of Joe and matches our existing patterns. "Joe looked tall" is a description of how he looked or was perceived – but looked, and likewise appeared, seemed, resembled – are not absolute. Including these terms in patterns might be as simple as adding alternative

lemmas to the core of some patterns but their inclusion adds an element of subjectivity to the resulting descriptions beyond the general bias of the narrator, which may require some adjustment to the definition of a descriptive sentence. As it is, their inclusion may be an indication of a lack of clarity in the definition as written.

4.2.1.3 Contractions

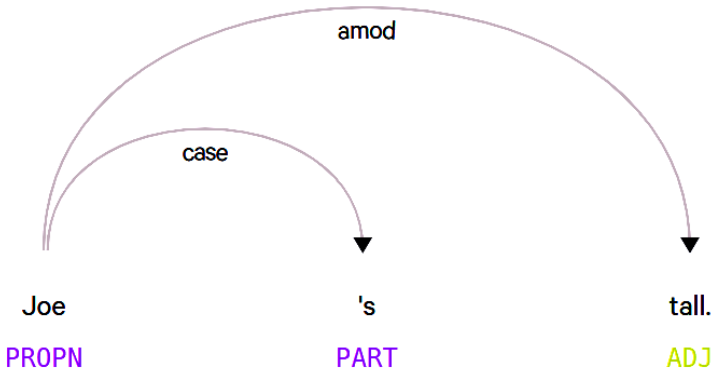
Something not accounted for in the patterns discovered is the different ways contractions can appear in the dependency tree – particularly possessive " ***'s*** " in a sentence. If a sentence resembles "He's tall" the system can interpret "He 's tall", treating the ***'s*** as the 'is' that was intended. The ultimate meaning is "He is tall" as in figure 4.4

Figure 4.4 Contraction with a pronoun



However, if the sentence contains a name, it may mark the ***'s*** as a case marker, interpreting the sentence, for instance, as being about Joe without offering a recognizable root word as used by all of the other patterns.

Figure 4.5 Contraction with a proper noun



This is an issue of ambiguity in the use of ‘s in general, as they can indicate both possession and being. It may be possible to correct for this by checking if the subject is also the root (as in the example in figure 4.5) and/or whether the subject has only amod and case dependencies.

4.2.1.4 Accuracy and Test Data

The NAR for the particular set of test data used in result set A was 0.721, meaning that if we had assumed the more likely case every time – that all sentences were not descriptions – we would identify 72.1% or approximately 88 sentences correctly. The description identification system classified 109 sentences or 89.3% correctly, a difference of 17% and 21 sentences. Along with the precision and recall rates, the difference suggests that the classifier’s performance is superior to a random result.

4.2.2 Analysis Result Set B

Result set B uses data labelled less strictly according to the given definition of a descriptive sentence. It’s a more natural human understanding of the definition of a sentence, where certain actions are understood to be descriptive. For example in the

sentence "his hair covered his face" cover is an action, but is here interpreted as a description.

There were 13 sentences otherwise classified as non-descriptions labelled differently in this set of results, which is more reflective of the system's performance as applied to descriptions in general. Overall this result set exposed the same difficulties found for section 4.3.1, alongside the coverage presented by the current patterns.

4.2.2.1 Precision and Recall

The algorithm compared with the broader definition of a description had a recall score of 0.617, suggesting that it detects approximately 6/10 descriptive sentences – lower than set A, which is to be expected when the result set contains descriptions of a sort it was not explicitly designed to detect. Its precision measured 0.879, suggesting that approximately 9/10 of the sentences it labels as descriptions are true descriptions with about 1/10 being false positives – which is an improvement over result set A, meaning the system identified some descriptions correctly despite them being outside of its intended scope. The F-Score meant to weight both kinds of performance is 0.725, reflecting the overall worse performance against the alternative definition.

4.2.2.2 Accuracy and Test Data

The NAR for the particular set of test data used in result set A was 0.615, meaning that if we had assumed the more likely case every time – that all sentences were not descriptions – we would identify 61.5% or approximately 75 sentences correctly. The description identification system classified 100 sentences or 82% correctly, a difference of 20.5% and 25 sentences. Along with the precision and recall rates, the difference

suggests that the classifier's performance is superior to a random result, even using the broader definition of a description.

CHAPTER FIVE

5. CONCLUSION

5.1 Conclusion

This thesis provides a promising starting point for the discovery and extraction of descriptions in narrative text through natural language processing techniques and linguistic patterns. The description identifier algorithm was able to achieve an f-score of 80.6% against a defined class of descriptions, and 72.5% against general descriptive sentences. The subject identifier algorithm achieved an 82.4% f-score. The results support the concept that direct linguistic patterns can be used for automated linguistic interpretation, using the system developed for this research as a proof-of-concept. With additional work and examination of further patterns, these algorithms can form the base of a means of automatically locating and extracting descriptions in written work.

5.2 Future Work

5.2.1 Refinement of Explicit Patterns In Description Identifier Algorithm

The patterns discovered over the course of this study were effective, but as noted in the examination of the results they could be improved. Given the amount of time required to comb through narrative text for examples of descriptive sentences, the amount of true narrative text used in the process of finding and extracting descriptive patterns, in this case, was limited. Iterative testing and examination of a larger set of formal training have the potential to build on the understanding of the given patterns, and the dictionaries required for these or other patterns like the descriptive adjectives mentioned in result set B.

5.2.2 Identifying Incidental Descriptions

Throughout the analysis of the narrative text for this research it became apparent that while explicit descriptive sentences are the most concrete and objective sources of description in text, they are not the most frequent. Implied descriptions require a level of understanding not present in the tools used to build the system described in this study, but incidental descriptions – that is, descriptions which appear in sentences whose primary purpose is not to convey a description – are common and potentially discoverable through the use of dependency trees and parts of speech.

5.2.3 Detail Extraction

The patterns used to identify descriptions and subjects clearly identify descriptive words. Adjectives and verbs used as descriptors could be extracted to form cumulative profiles of characters over a passage of text, although further patterns would need to be understood to do this correctly, including negation and the existence of multiple details or potential subjects in one sentence.

5.2.4 Detail Extraction

This research presents two algorithms for use in identifying subjects and descriptive sentences separately. In the future, these algorithms may be integrated in order to obtain both results for use in a system such as automatic character profiling or continuity checking tools.

5.3 Limitations

5.3.1 Collection and Prevalence of Data

The largest challenge in developing and testing the system was location appropriate data. While descriptions are an important part of written works, most descriptions in the texts originally selected for examination were implicit or incidental. Given the limited scope of time allowed for the study, a general pool of descriptive language not limited to narrative text was used.

The same limitation required the format of the testing data – short passages across a number of works which contained descriptions. The final testing data contained 30-40% descriptive sentences, which is not representative of the ratio found in the novels examined, where explicit descriptions were clustered with the introductions of new characters but otherwise vastly outnumbered by sentences conveying actions, setting and dialogue.

5.3.2 Defining a Description

The next challenge is in defining a description. The intended use of the system developed for this research was to extract physical descriptions of characters – but as in most natural language, what exactly is considered a description varies greatly by context and subject. Should immediate descriptions be considered – whether someone is tired, wet, fast? Does a person's description include their clothing (which may also vary scene to scene)? Do outside opinions – what others think they look like – qualify as descriptions? The answers to these questions rely on the purpose of the final application.

The specific definition of a description has some impact on the patterns used – though some examination suggests it would largely be adding words that are acceptable

as root verbs (is, has, wears, feels, thinks) and the specification of dictionaries of terms (physical attributes, pieces of clothing) which, using WordNet, do not need to be exhaustive to be effective. This is to say that these algorithms, by their nature, cannot be fully generic and must be at least somewhat tailored to a given definition of a subject and a description.

REFERENCES

Akulick, S., & Mahmoud, E. (2017). Intent Detection through Text Mining and Analysis. *Future Technologies Conference (FTC) 2017.* Vancouver, Canada.

Al-sharman, N., & Pivkina, I. V. (2018). Generating Summaries through Selective Part of Speech Tagging. *ICEMIS '18 Proceedings of the Fourth International Conference on Engineering & MIS 2018.* Istanbul, Turkey: ACM.

Chambers, N., & Jurafsky, D. (2009). Unsupervised learning of narrative schemas and their participants. *ACL '09 Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2* (pp. 602-610). Suntec, Singapore: Association for Computational Linguistics.

Dodell-Feder, D., & Tamir, D. I. (2018). Fiction Reading Has a Small Positive Impact on Social Cognition: A Meta-Analysis. *Journal of Experimental Psychology: General*, *147*(11), 1713-1727.

Einsohn, A. (2011). *The Copyeditor's Handbook: A Guide for Book Publishing and Corporate Communications* (3rd ed.). Los Angeles, California: University of California Press.

Elson, D. K., Dames, N., & McKeown, K. R. (2010). Extracting Social Networks from Literary Fiction. *ACL '10 Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics* (p. 138=147). Uppsala, Sweden: Association for Computational Linguistics.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters, 27*, 861-874.

Ghosh, A., Li, G., Veale, T., Rosso, P., Shutova, E., Reyes, A., & Barnden, J. (2015). Sentiment Analysis of Figurative Language in Twitter. *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)* (pp. 470-478). Denver, Colorado: Association for Computational Linguistics.

Hobbs, J. R. (1977). Pronoun resolution. *ACM SIGART Bulletin*, p. 28.

Jhavar, H., & Mirza, P. (2018). EMOFIEL: Mapping Emotions of Relationships in a Story. *WWW '18 Companion Proceedings of the The Web Conference 2018* (pp. 243-246). Lyon, France: International World Wide Web Conferences Steering Commitee.

Johnson, D. R. (2012, January). Transportation into a story increases empathy, prosocial behavior, and perceptual bias toward fearful expressions. *Personality and Individual Differences, 52*(2), 150-155.

Koopman, E. M. (2015, June). Empathic reactions after reading: The role of genre, personal factors and affective responses. *Poetics, 50*, 62-79.

Laprise, S., & Winrich, C. (2010). The Impact of Science Fiction Films on Student Interest in Science. *Journal of College Science Teaching, 40*(2), 45-49.

Lehman, D. E. (2018, June 26). *Organizing Your Fiction Series*. (Medium) Retrieved January 20, 2019, from The Writing Cooperative: https://writingcooperative.com/organizing-your-fiction-series-d28dae0472ed

Manuskript. (n.d.). Retrieved from www.theologeek.ch/manuskript

Matilal, B. K. (2015). *The Word and the World: India's Contribution to the Study of Language.* Oxford University Press.

Moore, A. (2011). The long sentence: A disservice to science in the Internet age. *Bioessays, 33*, pp. 193-193.

Nakashole, N., Weikum, G., & Suchanek, F. (2012). PATTY: A Taxonomy of Relational Patterns with Semantic Types. *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (pp. 1135-1145). Jeju Island, Korea: Association for Computational Linguistics.

Powers, D. M. (2011). Evaluation: From Precision, Recall, and F-Measure to ROC, Informedness, Markedness, & Correlation. *Journal of Machine Learning Technologies*, 37-63.

Pozzi, F. A., Fersini, E., Messina, E., & Liu, B. (2016). *Sentiment Analysis in Social Networks.* Morgan Kaufmann Publishers.

Primative, S., Spinelli, D., Zoccolotti, P., De Luca, M., & Martelli, M. (2016). Perceptual and Cognitive Factors Imposing "Speed Limits" on Reading Rate: A Study with the Rapid Serial Visual Presentation. *PLoS ONE, 11*(4). doi:http://dx.doi.org.library.sheridanc.on.ca/10.1371/journal.pone.0153786

Princeton University. (2010). *About WordNet*. Retrieved from WordNet: wordnet.princeton.edu

Rowling, J. K. (2018, January 30). *I plan a lot*. Retrieved from J.K. Rowling on Twitter: https://twitter.com/jk_rowling/status/958382036556886016

Schmidt, M. (2015, November). *Maintaining Continuity: Tales from the Copy Editor*. (Penguin Random House) Retrieved January 20, 2019, from News for Authors: http://authornews.penguinrandomhouse.com/maintaining-continuity-tales-from-the-copy-editor/

Schmidt, M. (2015, October). *On Writing: Creating Characters and Maintaining Continuity in Writing*. (Penguin Random House) Retrieved January 20, 2019, from News for Authors: http://authornews.penguinrandomhouse.com/on-writing-creating-characters-and-maintaining-continuity-in-writing/

Scrivener. (n.d.). Retrieved from www.literatureandlatte.com/scrivener

Sekine, S., & Nadeau, D. (2017). A survey of named entity recognition and classification. *Linguisticae Investigationes, 30*(1), 3-26. doi:https://doi.org/10.1075/li.30.1.03nad

spaCy. (n.d.). Retrieved from spaCy: https://spacy.io/

Szabó, Z. G. (2015). Major Parts of Speech. *Erkenntnis: An International Journal of Scientific Philosophy*, *80*, 3-29.

Vigen, T. (n.d.). *Literature Statistics*. Retrieved from tylervigen.com: http://www.tylervigen.com/literature/statistics

Wiebe, J. M. (1990). Identifying subjective characters in narrative. *COLING '90 Proceedings of the 13th conference on Computational linguistics*, *2*, pp. 401-406. Helsinki, Finland.

Wolf, T. (2017, July 7). *State-of-the-art neural coreference resolution for chatbots*. Retrieved from Medium: https://medium.com/huggingface/state-of-the-art-neural-coreference-resolution-for-chatbots-3302365dcf30

yWriter. (n.d.). Retrieved from http://www.spacejock.com/yWriter6.html