

# Generating Canonical Sentences from Question-Answer Pairs of Deposition Transcripts

Maanav Mehrotra

Thesis submitted to the Faculty of the  
Virginia Polytechnic Institute and State University  
in partial fulfillment of the requirements for the degree of

Master of Science  
in  
Computer Science and Application

Edward A. Fox  
Michael Hsiao  
Hoda Eldardiry

August 7, 2020  
Blacksburg, Virginia

Keywords: Natural Language Processing, Deep Learning, Legal Tech, Legal Depositions

Copyright 2020, Maanav Mehrotra

# Generating Canonical Sentences from Question-Answer Pairs of Deposition Transcripts

Maanav Mehrotra

(ABSTRACT)

In the legal domain, documents of various types are created in connection with a particular case, such as pleadings, transcripts, written discovery documents, memos, and emails. Deposition transcripts are one such type of legal document, which consists of conversations between the different parties in the legal proceedings that are recorded by a court reporter. Court reporting has been traced back to 63 B.C. It has transformed from the initial scripts of “Cuneiform”, “Running Script”, and “Grass Script” to Certified Access Real-time Translation (CART). Since the boom of digitization, there has been a shift to storing these in the PDF/A format. Deposition transcripts are in the form of question-answer (QA) pairs and can be quite lengthy for common people to read. This gives us a need to develop some automatic text-summarization method for the same. The present-day summarization systems do not support this form of text, entailing a need to process them. This creates a need to parse such documents and extract QA pairs as well as any relevant supporting information. These QA pairs can then be converted into complete canonical sentences, i.e., in a declarative form, from which we could extract some insights and use for further downstream tasks. This work investigates the same, as well as using deep-learning techniques for such transformations.

# Generating Canonical Sentences from Question-Answer Pairs of Deposition Transcripts

Maanav Mehrotra

(GENERAL AUDIENCE ABSTRACT)

In the legal domain, documents of various types are created in connection with a particular case, such as pleadings, transcripts, written discovery documents, memos, and emails. Deposition transcripts are one such type of legal document, which consists of conversations between a lawyer and one of the parties in the legal proceedings, captured by a court reporter. Since the boom of digitization, there has been a shift to storing these in the PDF/A format. Deposition transcripts are in the form of question-answer (QA) pairs and can be quite lengthy. Though automatic summarization could help, present-day systems do not work well with such texts. This creates a need to parse these documents and extract QA pairs as well as any relevant supporting information. The QA pairs can then be converted into canonical sentences, i.e., in a declarative form, from which we could extract some insights and support downstream tasks. This work describes these conversions, as well as using deep-learning techniques for such transformations.

# Dedication

*To my beloved parents Vishal and Mayuri Mehrotra.*

# Acknowledgments

I would like to start by thanking my advisor Dr. Edward Fox for his guidance, support and efforts without which my thesis wouldn't be possible. I would also like to thank James and Susan Chapman who allowed me the opportunity to work on this project. James gave me an insight as to what and how attorneys look at a deposition transcript. I also thank Dr. Hsiao and Dr. Eldadiry for their efforts and suggestions. I thank Mayfair Group LLC for providing us with a dataset to use and the necessary funds to run MTurk experiments. Thanks go to the Virginia Tech Institutional Review Board (IRB) that ensured the protection of those experiments, which followed suitable protocols as shown in Appendix E. Thanks also go to Amazon, for providing research credits aiding work on AWS. I am indebted to the Statistics Department, for funding me through my Master's degree, and all the staff of the CS Department and Virginia Tech for assisting me when needed.

I would like to express my gratitude to Saurabh Chakravarty for his ideas, support, and work ethic, which has reflected on me since the past year. I also thank all of the undergraduate students who have supported my work in some way and all my co-authors for the papers published through this work. I am grateful to all the members of the DLRL – Liuqing, Amirsina, Prashant, Bipasha, and Satvik – who have helped me through my queries.

I am thankful to my family and friends in India as well as my “Galat Phamileee” here, for supporting me. A special thanks to Abhishek, Deeksha, Palakh, Pranav, Siddharth, Shravani, and Varun for being there every step of the way.

# Contents

- List of Figures** **xii**
  
- List of Tables** **xv**
  
- 1 Introduction** **1**
  - 1.1 Background . . . . . 1
  - 1.2 Problem Statement . . . . . 3
  - 1.3 Motivation . . . . . 4
  - 1.4 Research Questions . . . . . 5
  - 1.5 Hypotheses . . . . . 5
  - 1.6 Thesis Outline . . . . . 6
  
- 2 Literature Review** **8**
  - 2.1 Core NLP Libraries . . . . . 8
    - 2.1.1 Natural Language Toolkit . . . . . 8
    - 2.1.2 spaCy . . . . . 8
    - 2.1.3 Gensim . . . . . 9
  - 2.2 Parsing . . . . . 9
    - 2.2.1 PyPDF2 . . . . . 10

2.2.2	Apache Tika	10
2.2.3	Amazon Textract	11
2.2.4	OpenCV	11
2.3	Dialog Act Classification	11
2.4	Transformation of Sentences	14
2.4.1	Rule-based System	14
2.4.2	Other Systems	16
2.5	Deep Learning based Architectures	16
2.5.1	Machine Translation	16
2.5.2	Language Models	18
2.5.3	COPYNET	19
2.5.4	Pointer Generator Network	20
2.6	Sentence Correction	21
2.7	Evaluation Metrics	22
2.7.1	Word-Based Evaluation	22
2.7.2	Semantic Similarity	23
2.8	Extra Tools	23
2.8.1	GROBID on Documents	23
2.8.2	Adobe-OCR	24
2.8.3	Tesseract-OCR	24

2.8.4	OCRmyPDF	25
<b>3</b>	<b>Parsing</b>	<b>26</b>
3.1	Processing PDF Images	27
3.2	Splitting Larger Sentences	31
3.3	Extracting Page and Line Numbers	33
<b>4</b>	<b>Data</b>	<b>36</b>
4.1	Proprietary Dataset	36
4.1.1	Dataset Observations	37
4.2	Tobacco Dataset	40
4.2.1	Dataset Observations	40
4.3	Ground Truth Annotation	43
4.4	MTurk dataset	45
<b>5</b>	<b>Transformational Techniques</b>	<b>48</b>
5.1	Pre-processing of Questions	48
5.1.1	Processing QAs with Extra Statements	49
5.2	Chunking and Chinking method	51
5.3	Post-processing of Questions	53
5.4	Sentence Correction	56
5.4.1	Using Language Models and Heuristics	56

5.5	Transformation using Deep Learning . . . . .	58
<b>6</b>	<b>Experiments and Results</b>	<b>61</b>
6.1	Parsing . . . . .	61
6.2	Proprietary Dataset . . . . .	65
6.3	Tobacco Dataset . . . . .	69
6.4	Sentence Correction . . . . .	72
6.5	Transformation using Deep learning . . . . .	74
6.5.1	Single model approach . . . . .	75
6.5.2	Multi-model approach . . . . .	76
6.6	MTurk Study . . . . .	77
<b>7</b>	<b>Conclusion and Future work</b>	<b>82</b>
7.1	Conclusion . . . . .	82
7.2	Research Contributions . . . . .	83
7.3	Future Work . . . . .	85
	<b>Bibliography</b>	<b>87</b>
	<b>Appendices</b>	<b>103</b>
	<b>Appendix A User Manual</b>	<b>104</b>

<b>Appendix B Developer Manual</b>	<b>106</b>
B.1 deposition-summarization/scratch . . . . .	106
B.1.1 Parsing and Anonymization . . . . .	106
B.1.2 Classifier . . . . .	107
B.1.3 Transformation . . . . .	108
B.2 saurabc/sentence-correction . . . . .	109
B.3 maanav/eval-scratch . . . . .	110
<b>Appendix C Other Experiments</b>	<b>111</b>
C.1 Parsing born-digital PDF files . . . . .	111
C.1.1 GROBID on Documents . . . . .	111
C.1.2 Adding Text layer . . . . .	114
C.2 Transformations . . . . .	116
C.2.1 Next sentence prediction using BERT . . . . .	116
C.3 Evaluation . . . . .	117
C.3.1 Flair Embeddings . . . . .	117
C.3.2 Run Time Analysis . . . . .	118
C.3.3 Coverage Analysis . . . . .	118
<b>Appendix D Discussion</b>	<b>120</b>
<b>Appendix E IRB Approval and Supporting Files</b>	<b>125</b>

E.1	VT IRB Authorization Letter . . . . .	125
E.2	Online Recruitment . . . . .	128
E.3	Consent Form . . . . .	129
E.4	Sample Tasks . . . . .	130
<b>Appendix F Mturk Final Report</b>		<b>143</b>

# List of Figures

1.1	A page from a legal deposition . . . . .	2
1.2	Implementation Pipeline . . . . .	3
2.1	Dialog Act classifier performance [20] . . . . .	14
2.2	Mapping of QA pair into declarative sentence (QA2D) [28] . . . . .	15
2.3	Schematic view of machine translation [46] . . . . .	17
2.4	High-level Architecture of BERT [29] . . . . .	18
2.5	Architecture of COPYNET [38] . . . . .	19
2.6	Architecture of PGN [74] . . . . .	20
3.1	Samples sections in a deposition . . . . .	26
3.2	Types of deposition transcripts . . . . .	28
3.3	Splitting the 4 sub-pages in Fig. 3.2b using box detection . . . . .	29
3.4	Final text file obtained after processing the condensed PDF page given in Fig. 3.2b . . . . .	30
3.5	Long sentence division . . . . .	32
3.6	Sub-figures show different ways a transcript can be summarized. Some parts of the figures have been blacked out to preserve anonymity. . . . .	34
3.7	Final JSON file output showing page number and line number variables . . . . .	35

4.1	Visualizing the distribution of QA pairs of M10 dataset . . . . .	37
4.2	Distribution of QA pairs based on their length . . . . .	39
4.3	Visualizing the distribution of QA pairs of tobacco dataset . . . . .	41
4.4	Distribution of QA pairs based on their length for tobacco dataset . . . . .	42
4.5	Visualizing the distribution of MTurk dataset for Task 1 . . . . .	46
4.6	Visualizing the distribution of MTurk dataset for Task 2 . . . . .	47
5.1	Pipeline which we aim to implement . . . . .	49
5.2	Part-Of-Speech tagged question . . . . .	51
5.3	Identifying chunks in the question . . . . .	52
5.4	Canonical form of QA pair . . . . .	52
5.5	Shows how the code is structured internally [21] . . . . .	54
5.6	Example of scoring of words and sentence using BERT . . . . .	57
5.7	Showing small samples of the source (a) and target (b) files for machine translation . . . . .	60
6.1	Visualization of percentage of handled documents . . . . .	62
6.2	Parsing with old technique . . . . .	62
6.3	JSON output from parsing with new technique . . . . .	63
6.4	Coverage of the rule-based transformer . . . . .	65
6.5	Distribution of ratings for auto-generated canonical sentences via MTurk . . . . .	81

C.1	Output of page parsed in GROBID . . . . .	112
C.2	Output of another page parsed in GROBID . . . . .	113
C.3	OCR output of Abode-OCR (left) and OCRmyPDF (right) . . . . .	114
D.1	Change of accuracy across steps. . . . .	122

# List of Tables

2.1	Some question dialog acts. [20]	12
2.2	Some answer dialog acts. [20]	13
3.1	Example where the question consists of multiple questions and answers.	31
3.2	Processing extra long QA pair given in Table 3.1.	33
4.1	Top 20 Question-Answer pairs for the M10 dataset. [21]	38
4.2	Top 7 important DA classes w.r.t. summaries.	39
4.3	Top 20 Question-Answer pairs for the tobacco dataset. [52]	41
4.4	Annotated DA-pair classes for M10 dataset.	44
4.5	Annotated DA-pair classes for tobacco dataset.	45
4.6	Distribution of the MTurk dataset curated for Task 1.	45
4.7	Distribution of the MTurk dataset curated for Task 2.	47
5.1	Question-Answer pairs having noise (highlighted in bold).	49
5.2	Example of QA pairs with extra statements.	50
5.3	Explanation of some rules implemented for transformation.	53
5.4	Examples of QA pairs having “- -”.	55
5.5	Example of an N-gram based swap.	57

5.6	Example of one word deletion. . . . .	58
5.7	Example of one word replacement. . . . .	58
6.1	QA distribution with different parsing techniques. . . . .	64
6.2	Top 13 DA distribution on new parsing technique. . . . .	64
6.3	Evaluation results for M10 dataset. . . . .	66
6.4	Evaluation results of Tobacco Dataset. . . . .	69
6.5	Experimental setup for sentence correction. . . . .	72
6.6	ROUGE -1 scores for all models . . . . .	72
6.7	ROUGE -2 scores for all models . . . . .	73
6.8	Semantic Similarity scores for all models . . . . .	74
6.9	Deep learning result on mixed dataset . . . . .	75
6.10	Deep Learning results on mixed dataset with POS tagging . . . . .	76
6.11	Deep Learning results on M10 dataset . . . . .	77
6.12	Evaluation against MTurk annotated sentences . . . . .	78
6.13	Good and bad examples of grammatically correct sentences . . . . .	79
6.14	Good and bad examples of natural/readable sentences . . . . .	80
6.15	Good and bad examples of completeness of sentences . . . . .	81
C.1	Sample example where discontinuity is high. . . . .	116
C.2	Run time for each step in transforming QA pairs (in seconds). . . . .	118

C.3 Coverage of rules across DA pairs. . . . . 119

# List of Abbreviations

AMT Amazon Mechanical Turk

CRF Conditional Random Field

DA Dialog Act

DBN Deep Belief Network

HMM Hidden Markov Model

MTurk Amazon Mechanical Turk

NER Named-Entity Recognition

NLP Natural Language Processing

NLTK Natural Language Tool Kit

NMT Neural Machine Translation

OCR Optical Character Recognition

POS Part-of-Speech

QA Question-Answer

SVM Support Vector Machine

# Chapter 1

## Introduction

### 1.1 Background

Since the boom of digitization, there has been a drastic increase in the number and variety of documents available online. Trial testimonies, transcripts, video depositions, reports, and charts are some examples of documents which are used in legal proceedings and are nowadays available online for public use. There has been a great deal of progress made in the field of Natural Language Processing (NLP) and Machine Learning (ML) for processing and extracting information from this textual data. Advances have also been made in the field of summarization of documents, but these do not extend well to deposition transcripts.

A deposition transcript is a legal document, which records conversations between a lawyer and person involved in legal proceedings. This conversation is generally documented verbatim by a third party such as a court reporter. The main aim of a deposition is to question the deponent (witness) and extract important, relevant information regarding the legal case. The deponent may provide information through their account of events, testimony as an expert, etc. Since the attorney is fishing for information from the deponent, it generally follows a question-answer based format where the attorney asks a question, and the deponent answers it. This conversation loosely follows (English) language rules.

Fig 1.1 shows a page from a legal deposition. To conserve memory space, legal companies

Page 15

1           Lynch - Confidential

2 gone in this case?

3           A.    One notice would go to -- well, you

4 have my notices there. I could go through the

5 distribution.

6           Q.    I believe I have only one. I am

7 trying to find that one.

8           Let me ask you to take a look at

9 what's been marked as Plaintiff's Exhibit 28.

10           (Document marked Lynch Exhibit 28 for

11 identification, this date.)

12           Q.    While you are looking at it, I will

13 find a copy for counsel.

14           MS. CECIL: I'm sorry. What number is

15 this?

16           MR. KLONTZ: 28.

17           Q.    Do you recognize that document,

18 Mr. Lynch?

19           A.    Yes, I do.

20           Q.    Is this one of the five notices?

21           A.    Yes, that's correct.

22           Q.    And to whom did this notice go?

23           A.    This went to employees of Philip

24 Morris Companies, Inc., Philip Morris

25 Incorporated (PM USA), and Philip Morris

Esquire Deposition Services  
1-800-944-9454

3990005890  
3990005890

Source: <https://www.industrydocuments.ucsf.edu/docs/tryp0183>

Figure 1.1: A page from a legal deposition

introduced the condensed format, which has the content of 4 pages of a deposition transcript in a single page.

Since the deposition transcript is recorded word-for-word, it also includes conversational artifacts such as “um” or “uh” that signify that the speaker is thinking. There are also instances where the original conversation is interrupted, leading to incomplete sentences, or abandonment of a topic altogether. All of this introduces a lot of noise within the document, making it hard to process.

These documents are quite large. Analyzing such documents by hand becomes toilsome. Having some sort of summary of such documents will allow for ease of understanding the deposition topic, and the proceedings that took place.

## 1.2 Problem Statement

At a very high-level, we have the following problem statement:

*Can we process a given deposition transcript and transform the QA pairs within them into their canonical forms? Can we implement an end-to-end pipeline such as Fig. 1.2 to produce the canonical sentences?*

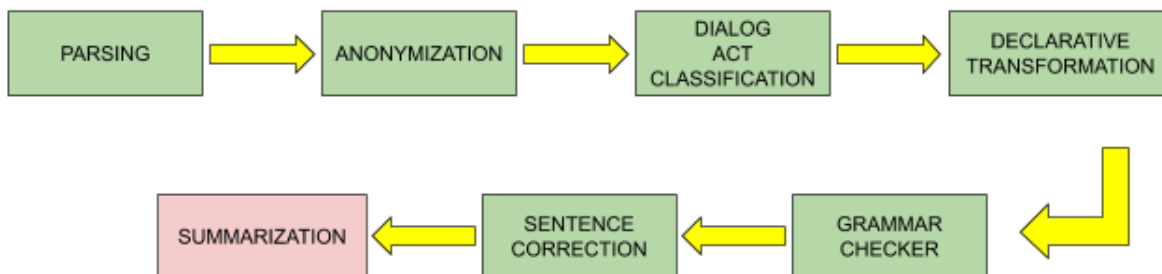


Figure 1.2: Implementation Pipeline

There is a deposition transcript that consists of questions and answers. We use the ontology and parsing techniques defined in [20] to parse and classify the questions and answers. Using the question-answer pairs as well as the respective dialog acts (DAs) we want to develop some method or pipeline to process them into suitable forms.

For solving the same, we divide the problem into 3 parts:

- **Declarative Transformation** - Developing rule-based and/or deep-learning-based NLP methods to transform a QA pair into its canonical form. These transformation

rules might be DA specific. Chapter 5 describes this in more detail while Section 5.5 covers the deep-learning based transformations.

- **Grammar Checking** - Developing methods or using libraries to identify and check whether a sentence makes grammatical sense or not. Section 5.3 has more details on this.
- **Sentence Correction** - Developing and/or implementing previously published methods to correct erroneous sentences. This may be by correcting spelling errors, swapping word positions, correcting tense of a verb, etc. More details on this can be found in Section 5.4.

### 1.3 Motivation

As part of the larger goal of summarizing documents with a question-answer based origin, parsing the documents, classifying the type of question or answer, and processing them, would be useful for researchers and commercial use as well. Processing of the QA pairs would allow a better understanding of the document and allow the identification of important instances or conversations.

Work done concerning question-answer pairs is generally in the field of answer generation, question generation, or natural language inference (NLI) [8, 28, 30]. In the domain of legal tech, there has been work done for parsing legal documents, semantic analysis of legal documents, and summarization of legal case judgments, but negligible work done to process deposition transcripts.

## 1.4 Research Questions

As part of this work, we plan to address the following research questions:

**RQ1:** Can we improve the parsing of deposition transcripts?

**RQ2:** Can we combine QA pairs into their canonical form?

**RQ3:** Can sentence correction be used to improve the form of sentences?

**RQ4:** Can deep learning methods be used to generate the canonical sentences from question and answer pairs.

**RQ5:** Can we evaluate the transformed sentences on quantitative metrics such as ROUGE and similarity score, using them to decide whether a transformation is good or bad?

**RQ6:** To what extent can external aid, such as from Amazon Mechanical Turk (MTurk), be useful in helping us identify issues in the developed transformation?

## 1.5 Hypotheses

We constructed the following hypotheses when we started our work:

1. The PDF files studied, including those provided as page images with 4 pages combined on a PDF page, can be accurately transformed into QA pairs, along with their page and line numbers.
2. Our linguistic transformation rules can achieve a ROUGE score of 0.6 when converting our QA pairs into what our judges consider roughly equivalent declarative statements.

3. Our grammar checker and sentence correction system will improve ROUGE-2 scores by 10% over the version produced by our linguistic transformation rules.
4. Deep learning methods, with the training data provided, will achieve a correctness score better than that of the linguistic transformation rules.
5. Our judges consider that the generated declarative statements with ROUGE-1 and -2 scores of more than 0.6, and a semantic similarity score of more than 0.85, demonstrate a good transformation.
6. Our experiment with AMT will document the need for improvement in at least 20% of the generated declarative statements.

## 1.6 Thesis Outline

- Chapter 1 outlines the basic aspects of our research problem, i.e., the motivation, hypotheses, and research questions.
- Chapter 2 presents prior and related work done that is relevant to this thesis. It also consists of some basic background information on Python libraries which have been used for development.
- Chapter 3 introduces the techniques implemented to improve the parsing of documents.
- Chapter 4 describes the datasets that we have used for our work.
- Chapter 5 introduces rule-based and deep learning based transformation techniques on QA pairs.
- Chapter 6 explains the experiments we ran and the subsequent results.

- Chapters 7 presents the conclusion, research contribution of this work, and possible future work.
- Appendix A is the user manual giving details on how to setup and run the code base.
- Appendix B is the developer manual explaining how the code was developed and information on each python script.
- Appendix C comprises of details regarding some preliminary exploration and studies done on this work.
- Appendix D is a discussion section which explains the need for improving the speed of processing, ensuring privacy, etc.
- Appendix E comprises of the IRB Authorization letter as well as the supporting documents for it.
- Appendix F outlines the final results of the Amazon Mechanical Turk experiment.

# Chapter 2

## Literature Review

In this chapter we discuss prior and related work done, relevant to our study. We also discuss different tools used for the study.

### 2.1 Core NLP Libraries

#### 2.1.1 Natural Language Toolkit

Natural Language Toolkit (NLTK) [13, 14] is an open-source library written in Python, developed by researchers and scholars to help create complex NLP functions. Its main aim was to help students explore ideas and support education. NLTK is a toolbox of NLP algorithms giving the ability to finely control NLP algorithms, including stemmers. This allows us to mix & match and develop the best solution for our task. Some lexical tasks handled by NLTK are word and text tokenizing, part-of-speech tagging, and named-entity recognition.

#### 2.1.2 spaCy

spaCy [41] is an open-source library written in Python and Cython. The main aim of spaCy is to provide the best way to do a particular NLP task. It is less complicated than NLTK

and is suitable for application development. spaCy developers keep their algorithms updated consistently. Even though its performance, time-wise, is much better than NLTK for many tasks, it also hogs memory. It can perform all the tasks done by NLTK but also supports integrated word vectors, neural network models, dependency parsing, and entity linking [43].

### 2.1.3 Gensim

Gensim [69] is an open-source library developed in Python. This package allows us to process text, work with pre-defined and custom vector models, and build topic models. It has been developed to handle large text files without having to load the entire file in memory. The package has implementations of some \*2vec algorithms along with Latent Semantic Analysis (LSA) and Latent Dirichlet Allocation (LDA).

## 2.2 Parsing

Parsing of legal depositions is not a simple task. There is in general lack of availability of legal documents in a structured, standard format. Work such as [48] introduces a methodology to produce a semantic representation of legal documents. The authors use Akoma Ntoso, a popular legal meta-schema, and re-purpose it to use for Greek legislation. They identify the structure of the legal documents and the most common parts such as the introduction, text body, and end part. They develop a new domain-specific language (DSL) enabling the implementation of syntactic rules and parsers.

Another work [66] introduces a pipeline from parsing to DA classification. For parsing of the different documents, they chose to use Apache Tika [58]. They identified different sections of a deposition transcript such as header, footer, page number, examination section, etc.

Within the examination section, they were also able to identify who the attorney asking the question was, as well as the questions and answers. The authors also introduced the ability to anonymize the deposition transcript for privacy purposes. A combination of regular expressions and a Named Entity Recognizer, specifically the Stanford NER [33], was used to identify email IDs, phone numbers, names, and other identifiable traits. They were then replaced with their anonymized versions.

The following sub-sections give us some more information and other alternatives to parsing of PDF files.

### 2.2.1 PyPDF2

PyPDF2 [78] is a Python-based library which is a super-set of the pyPdf library that was previously implemented. PyPDF2 has the ability to extract information and perform different operations on PDFs. For information extraction, it can extract metadata (title, author, number of pages, etc.) and the main textual contents of each PDF page. It can also be used to perform PDF operations such as merging, splitting, or cropping PDF pages. This is possible because this works on StringIO objects rather than file streams.

### 2.2.2 Apache Tika

Apache Tika [34, 58] is a parser library/toolkit developed in Java to detect document types and extract the contents. It can extract the metadata as well as the full-text within the document. It supports various file types (such as XML, HTML, PDF, RTF, etc.) as well as text extraction from images using Tesseract-OCR [76], making it a one-stop solution for parsing. Apache Tika uses existing specialized parser libraries for each document type. It can be useful for applications such as content analysis, digital asset management, document

analysis, and search engine indexing.

For this project we make use of Tika-Python [57], which is a Python port of the Apache Tika library, allowing Tika to be called natively by the Python community.

### 2.2.3 Amazon Textract

Amazon Textract [2] is a service provided by Amazon, which automatically extracts text and data from scanned documents. It not only can extract text similar to simple optical character recognition (OCR) systems but by leveraging black-box machine learning algorithms it can also identify forms or information stored in tables. This provides us with an alternative way to extract text from images or PDFs with high confidence even if the quality of the documents isn't good.

### 2.2.4 OpenCV

Open Source Computer Vision Library (OpenCV) [16] is a suite of programs written in C++ aimed at providing a common infrastructure for computer vision applications. OpenCV can be used for image and video analysis. For this work, we use OpenCV to divide a page of a PDF file to process its content more easily.

## 2.3 Dialog Act Classification

A Dialogue Act (DA) indicates the meaning of an utterance at the level of illocutionary force [56]. A dialogue act is the function of a sentence (or its part) in the dialogue. Research on DA classification has been going for quite some time, but it flourished in the 2000s, with the

growth in study of spoken dialog systems.

Dialog act classification is a complicated task. First, we need to define an ontology for sentences, then identify information within them, and finally, segment them for dialog act recognition. For identifying the context of the statement, dialogue lexical information, syntactic information, semantic information, prosody, and previous sentence context can be leveraged from sentences of the utterance.

Category	Description	Example
wh	This is a wh-* kind of question. These questions generally start with question words like who, what, where, when, why, how, etc.	What time did you wake up on the morning the incident took place?
wh-d	A wh-* kind of question which also consists of some extra sentences or information before the actual question.	You said generally wake up at 7:00 am in the morning. But what time did you wake up on the morning the incident took place?
bin	This is a binary question. These are questions that can be answered with a simple “yes” or “no”.	Is that where you live?
bin-d	In a binary-declarative question, the person who asks the question knows the answer but asks for verification.	That is where you live, right?
qo	This is an open question. These questions are general questions which are not specific to any context. These questions are asked to know the opinions of the person who is answering.	Do you think Mr. Pace made a good decision?
or	This is a choice question which offers a choice between several options. It is made up of two parts, which are connected by the conjunction “or”.	Were you working out for fun or were you into body building?

Table 2.1: Some question dialog acts. [20]

Initial works in dialog act segmentation and classification, such as [4, 32, 42, 54, 56, 79, 80] used the lexical & semantic features, along with prosodic cues, to develop and train models

for classification. Semantic classification trees, Support Vector Machines (SVMs), Hidden Markov Models (HMMs), and Deep Belief Networks (DBNs), are some of the techniques used for developing statistical models. Dialogue history is modeled using HMMs with word-based and prosodic features in [79]. [70] combines HMM and neural networks and compares the performance on the CallHome Spanish corpus [35]. Later works such as [4, 45, 49, 68] introduce the concept of Conditional Random Field (CRF) to label sequences of dialog acts.

Category	Description	Example
y	It is a category when a person answering the question means yes.	“yes”, “yeah”, “Of course”, “definitely it is”, “that’s right”, “I am sure”, etc.
y-d	It is a category when a person answering the binary question not only says yes but also explains the answer.	Yes. I play badminton because my doctor advised me to.
n	It is a category when a person answering the question means no.	“No”, “I don’t think so”, “certainly not”, “I am afraid not”, etc.
n-d	It is a category when a person answering the binary question not only says no but also explains the answer.	No. I am not interested in playing Cricket because it takes a lot of time
sno	It is a statement that has no-opinion. This is an informative statement made by the person answering the question.	I retired from my job in 2010.
ack	It is a response that indicates acknowledgment.	“Okay”, “Um-hum”, “I see”, etc.
dno	It is a response given when the person doesn’t know, or doesn’t recall, or is unsure about the answer to the question asked.	I don’t recall what happened that day

Table 2.2: Some answer dialog acts. [20]

[44, 55, 72] are some works that introduce deep-learning as a solution to this problem. These papers introduce architectures such as LSTM and RNN to achieve state-of-the-art results in DA classification. [24, 50] leverage Conditional Random Field dependencies along with deep learning architecture for achieving close to human annotation performance for some cases.

[20, 66] introduces a new ontology for datasets following a question-answer form, specifically focusing on legal depositions. The authors define 20 DAs which can be divided into two different sets, one for questions, the other for answers. Table 2.1 and Table 2.2 show some of the dialog acts defined. The authors adapt previously state-of-the-art classification methods such as CNNs (fine-tuned), LSTM (fine-tuned), and also utilize the BERT architecture for sentence classification. Across all 3 models they found that the classifier using BERT performs the best as seen in Fig. 2.1.

Classifier	F1-score
BERT	<b>0.84</b>
CNN	0.57
LSTM	0.71

Figure 2.1: Dialog Act classifier performance [20]

## 2.4 Transformation of Sentences

### 2.4.1 Rule-based System

[28] proposes a new method for automatically deriving NLI datasets from the growing abundance of large-scale question answering datasets. Initially, the authors talk about Natural Language Inference (NLI), its importance, and various datasets presently available, while explaining the advantages of deriving NLI from QA. Moving forward they then emphasize how their approach, called QA2D, hinges on learning a sentence transformation model which converts question-answer pairs into their declarative forms. As input, they have a passage, question, and answer as shown in Fig. 2.2. The transformed declarative sentence is then compared to the statements within the passage to identify whether it is true (positive

entailment) or false (negative entailment/contradiction). Through the categorization of the declarative sentences according to positive and negative entailment, they effectively generate a new NLI dataset.

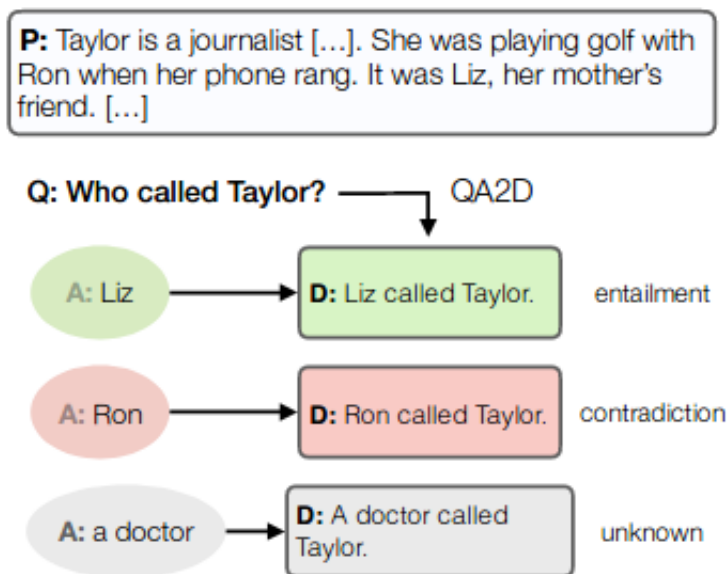


Figure 2.2: Mapping of QA pair into declarative sentence (QA2D) [28]

[28] explores three different ways to perform the declarative transformation: a rule-based system, crowd-sourcing, and a neural sequence model. For the rule-based system, there are a set of rules and specific steps followed, and it is highly dependent on part-of-speech tagging and parsing accuracy. For the crowd-sourced approach, turkers were asked to fix the declarative sentences. The authors selected SQuAD [67] to be the main source of QA pairs because of its large size, high quality, and syntactic diversity. From the crowd-sourcing, a dataset of (Q, A, D) tuples is available. This is used to learn a model of  $p(D|Q, A)$ , implemented with an encoder-decoder architecture. The inputs Q and A are each encoded using a bi-directional three-layer LSTM. D is then generated using a three-layer LSTM decoder equipped with one attention head for each question and answer, and a copy mechanism based on [38]. The authors observe that predictions of neural-based and rule-based approaches match 40% of

the time. The neural based implementation performed better, as it was also able to handle some semantic modification. The rule-based approach was more robust when considering longer sentences, while the neural-based approach worked better on shorter length sentences.

### 2.4.2 Other Systems

The concept of transformation is not limited just to transformation into declarative sentences. This can be used for question generating systems, paraphrasing systems [8, 30], and sentence simplification systems [22]. [22] introduces a new finite state grammar (FSG) and a super tagging model to produce dependency linkages, eventually aiming to reduce the complexity of sentences. Quarc [71] is a rule-based system that uses lexical and semantic heuristics to answer questions on a short reading comprehension test. It introduces some semantic classes – such as HUMAN, LOCATION, TIME, and MONTH – which are useful for answering who, where, and when type questions. The authors could observe a specific pattern for *what* questions, but *why* questions could be answered by identifying certain keywords in the comprehension.

## 2.5 Deep Learning based Architectures

### 2.5.1 Machine Translation

Transforming a QA pair into a canonical form can also be formulated as a machine translation problem. Though we have the same source and target languages, the input and output have different forms. Works like [25] employ an encoder-decoder based approach to translate text from one language to another. The idea is to encode the input sentence into a vector using an RNN and then apply the decoder to the encoded representation to yield a target output

sentence. Challenges with vanilla sequence-to-sequence models are that they are repetitive, and the decoder does not always know when to stop. Work in [6] addressed some of the challenges with the sequence-to-sequence models by adding an attention layer over each output of the input RNN cells. Through training, the system learned how to map a source language input word to an output word in the target language, based upon the context of the source word.

OpenNMT [46] is an open-source toolkit developed for advances in neural machine translation and neural sequence learning. It was designed with three main aims:

- Prioritize first training and test efficiency.
- Maintain model modularity and readability.
- Support significant research extensibility.

This toolkit has been implemented in the Lua/Torch mathematical framework and over the years extended to applications such as image/speech/video to text conversion. Fig. 2.3 presents a schematic view of how machine translation works.

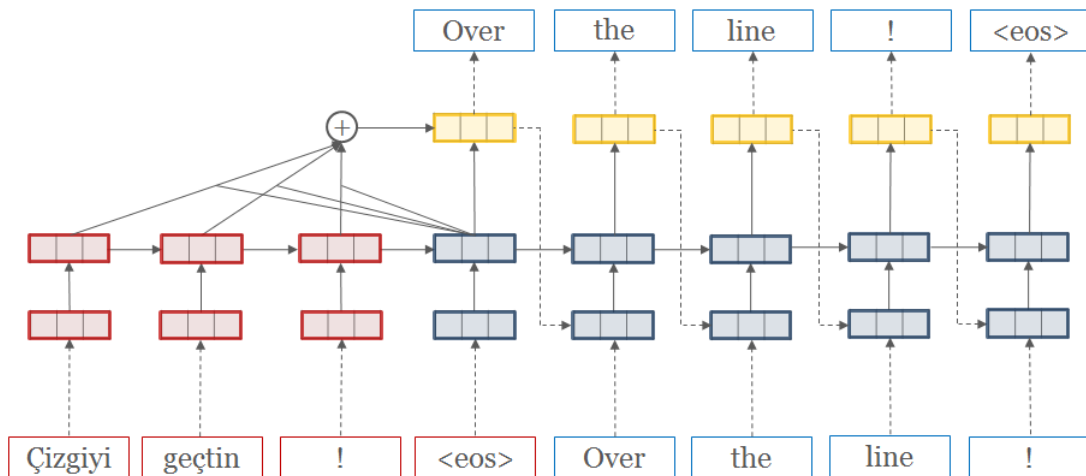


Figure 2.3: Schematic view of machine translation [46]

## 2.5.2 Language Models

Bidirectional Encoder Representations from Transformers (BERT) [29] is a pre-trained unsupervised language model. The main aim of developing BERT was to enable computers to understand the meaning of words and sentences using surrounding text as context. BERT had a transformer-based architecture which could read text bi-directionally allowing it to better understand contexts. The system used Masked Language models and Next Sentence Prediction strategies for training. It is observed that BERT was able to achieve state-of-the-art performance for 11 NLP tasks. Fig. 2.4 shows a high-level architecture of BERT and how its representation is conditioned on the left and right context.

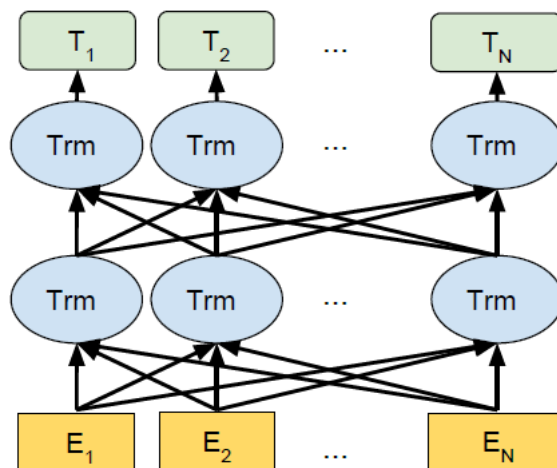


Figure 2.4: High-level Architecture of BERT [29]

Another quite popular language model is AI-GPT2. AI-GPT2 [65], the successor to OpenAI GPT, is an unsupervised transformer-based language model which has been trained to specifically predict the next word, given all the previous words. It is a generative model that has been trained on 8 million text documents with 1.5 billion parameters. It achieves state-of-the-art performance on many NLP tasks as well.

### 2.5.3 COPYNET

[38] is a sequence-to-sequence framework that has been developed to enable the possibility to copy contents from the input itself. The authors believed that the previously implemented encoder-decoder architectures were too reliant on the meaning and understanding of the sentence, which in some cases may not be required.

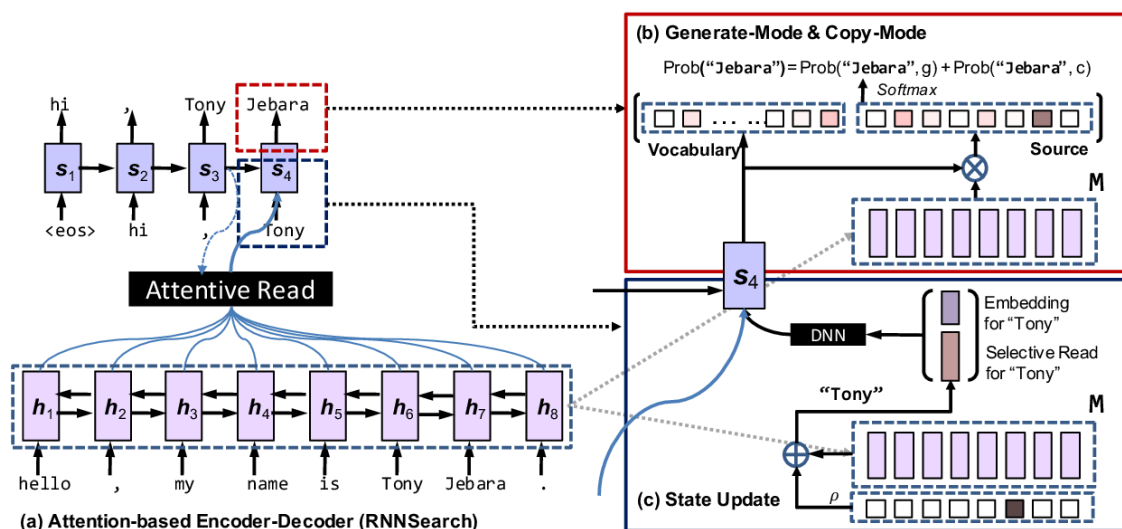


Figure 2.5: Architecture of COPYNET [38]

From Fig. 2.5, the architecture follows [6] apart from a few changes in prediction, state updating, and reading within the decoder. For fetching the content, [38] combines both content-based and location-based addressing. Looking more closely at the decoder, it uses two scoring mechanisms and two attention mechanisms. Generation score (similar to [6]) and copy score are the two scoring mechanisms implemented, while *selective read* and *attentive read* are the attention mechanisms. In attentive read, it follows the usual attention mechanism that produces a weighted sum over the encoded source states according to how similar each encoded source state is to the latest decoder state. In selective read, they produce the weighted sum over the encoded source states, but the weights are just the corresponding re-normalized copy probabilities from the previous time step [82]. This architecture allows

switching in-between generation and copying, allowing it to nicely integrate copied words along with generated words at proper places. The authors ran three different case-studies (dataset on simple patterns, text summarization, and single-turn dialogues) and observed a significant improvement for the single-turn dialogues dataset.

## 2.5.4 Pointer Generator Network

Conventional abstractive summarization systems suffer from problems such as the generation of inaccurate text, repetition of phrases/parts, and no pre-trained models. The work done in [74] aims to solve the same. A Pointer-Generator Network (PGN) [74] is an abstractive summary generation system, which aims to use the concept of “copying” for improving summary generation. Fig. 2.6 shows the architecture implemented.

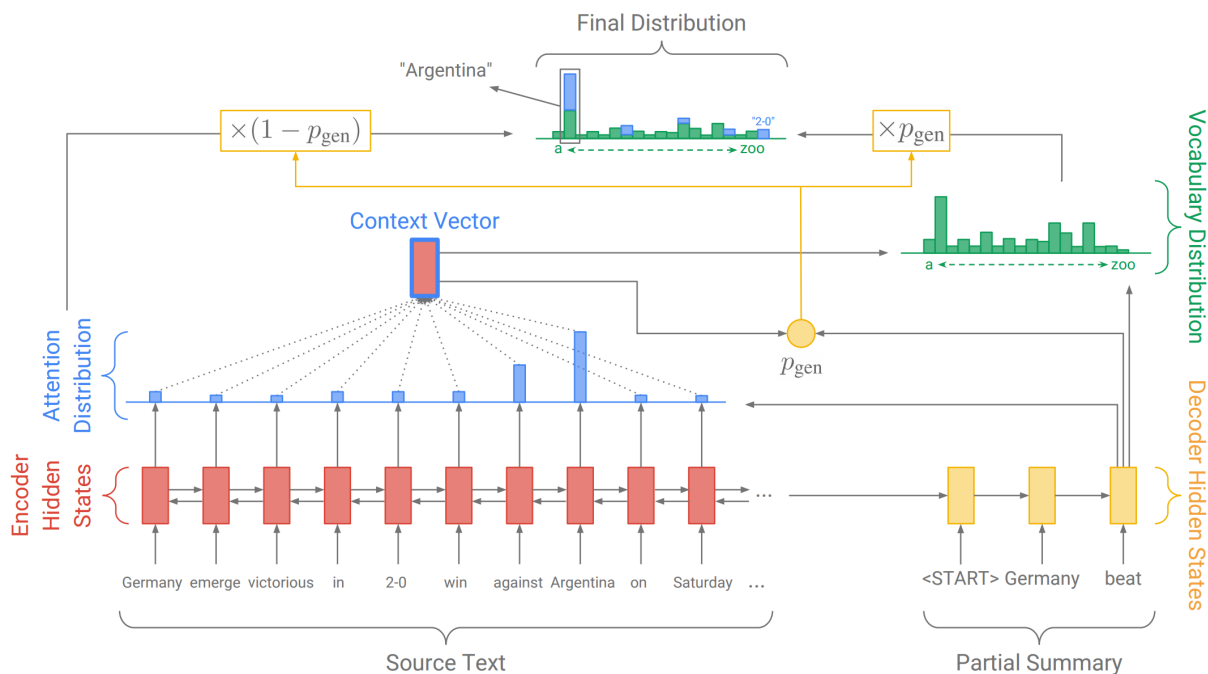


Figure 2.6: Architecture of PGN [74]

PGN uses a pointer which points to words which can be used for copying. It learns the ability

to generate, as well as learns when to copy. This allows the system to generate factually correct data in the summaries. Another technique implemented by the authors is called coverage. This allows the network to keep track of what words or phrases have been covered until that point in time. The network penalizes the score for repeatedly covering the same parts, allowing the system to reduce the possibility of repetition in summaries.

## 2.6 Sentence Correction

The task to correct the lexical, spelling, and punctuation errors within a sentence is called Grammatical Error Correction (GEC). GEC and sentence correction can be used interchangeably as within GEC we take an incorrect sentence and try and correct it. Work to develop such systems has been going on for quite some time [18, 19, 39, 60, 73]. [18, 60] employs a rule-based approach where pattern-matching was typically used to identify erroneous parts and use string replacement for correction. [39, 73] considers GEC as a multi-class classification problem. For error correction, there is a finite set of possible correction candidates from the class labels. These solutions, many a time, ignore dependencies between words and the sentences.

Newer GEC systems [11, 27, 36, 85] are focused on leveraging the advantages of machine translation, i.e., considering transformation from erroneous sentences to correct sentences. In Statistical Machine Translation (SMT), two models are developed: a language model that outputs the probability for a target sentence, and a translation model that outputs the conditional probability. In Neural Machine Translation (NMT) an encoder-decoder architecture is used, where the encoder encodes source words into vectors and the decoder re-transforms the vector into the target sequence. But, these systems are data-hungry and require a lot of data for better performance.

[37] uses synthetic data and a transformer-based architecture to solve the GEC task. For developing realistic synthetic data, the authors use the suggestions from the Aspell Spellchecker. The system is trained using synthetic and authentic sentences of the WMT News Crawl Corpus, and then fine-tuned. For better performance an ensemble of the transformer systems was used to generate the corrected sentences, to better address the sentence context.

[5] is a text-editing model that, instead of generating a target sequence of words, generates a sequence of edits. The authors think of GEC as a local sequence transduction problem, and develop operator type tokens – such as keep, delete, and add – as part of expressions used to process each sentence. This work may lack the ability to look at dependencies overall in a long sentence, but still yields results with comparable performance.

## 2.7 Evaluation Metrics

### 2.7.1 Word-Based Evaluation

Recall-Oriented Understudy for Gisting Evaluation (ROUGE) [53] is a word-based set of metrics used to evaluate automatic summarization systems and machine translation systems. It was developed opposite to Bi-lingual Evaluation Understudy (BLEU) [63] scores which are based on the brevity penalty. Metric for Evaluation of Translation with Explicit Ordering (METEOR) [7] is another evaluation metric that can be used for machine translation outputs. It aims at producing correlation at a segment level and not just at the corpus level like BLEU. The ROUGE scores can be calculated using precision, recall, or F-1 score with each having its implementation. Some variants of ROUGE include ROUGE-N, ROUGE-L, ROUGE-W, ROUGE-S, and ROUGE-SU. This evaluation metric is limited as it doesn't take into account the meaning of the sentences.

## 2.7.2 Semantic Similarity

Another quantitative way to evaluate sentences may be through calculating semantic similarity between them. Latent Semantic Indexing [62], Word Mover's Distance [51], and LDA with Jensen-Shannon distance [75] are some ways through which this can be done.

InferSent [26] is a sentence embedding developed by Facebook which provides representations for English sentences. There are two parts to producing an InferSent embedding. It consists of a sentence encoder which takes a word vector (GloVE [64] or FastText [15]) and encodes a sentence into its corresponding vector. The encoded vector is then passed through a NLI classifier which outputs a class (entailment, contradiction, and neutral). The authors ran experiments with different sentence encoder architectures but observed the best performance with BiLSTM with max/mean pooling. [84] is another alternative to calculate sentence similarity. It calculates relatedness by considering the depths of the two synsets in the WordNet taxonomies, along with the depth of the Least Common Subsumer. This score is normalized and continuous. A drawback is that this score is dependent on the quality of the graph generated. A variation of this scoring system can be used to calculate the similarity between sentences.

## 2.8 Extra Tools

### 2.8.1 GROBID on Documents

GeneRation Of Bibliographic Data (GROBID) is a machine learning library which can be used for extracting, parsing, and re-structuring raw documents such as PDF files into structured formats such as XML/TEI documents [1]. This library was developed keeping technical

and scientific publications in mind. It equipped us with a method to extract full-text as well as metadata information from the PDF documents. GROBID is light-weight and a fast way to process scientific documents.

### 2.8.2 Adobe-OCR

Adobe Acrobat is a family of application software and Web services developed by Adobe Inc. to view, create, manipulate, print, and manage files in Portable Document Format (PDF). [83] In the recent updates of Adobe Acrobat the company has introduced a PDF to text conversion system, which can be used to convert scanned files into editable/searchable documents. This is a black-box code for conversion.

### 2.8.3 Tesseract-OCR

Tesseract [76] is an Optical Character Recognition engine, whose main aim is to convert images of typed, handwritten or printed text into its machine-encoded form, electronically. Tesseract presently has support for over 100 languages and can identify text written from left-to-right or right-to-left. It can also identify the layout of the text and then accordingly parse the text. Many other OCR engines also use Tesseract [77] in some way or form. In the newest version, it introduced a new OCR engine based on LSTM networks, but there are still a few drawbacks such as poor quality scans resulting in poor OCR results, not being as accurate as some other commercial solutions, some gibberish being reported in the OCR output, and inability to process born-digital PDF files.

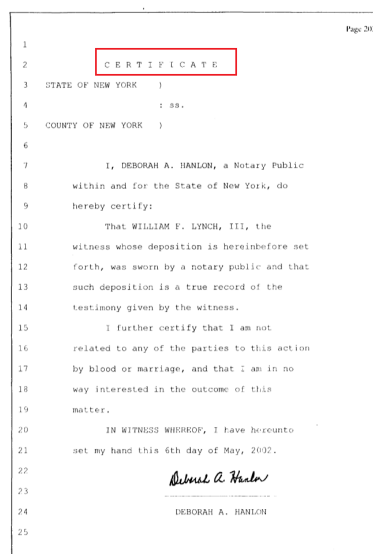
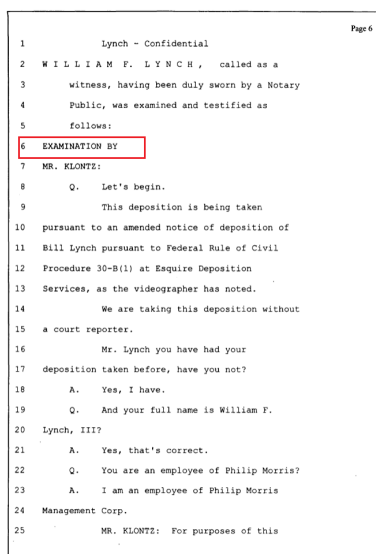
### 2.8.4 OCRmyPDF

This is a Python 3 library that adds an OCR text layer to PDF files, specifically. OCRmyPDF [9] analyzes each page of a PDF to determine the color-space and corresponding resolution needed to capture the information. It has some additional image editing features which allow it to better extract content than Tesseract-OCR. This library is highly dependent on Tesseract-OCR and has similar drawbacks as well. [9]

# Chapter 3

## Parsing

As mentioned in Section 1.1, legal depositions are a witness's sworn out-of-court testimony on a particular event, their expert opinion on a topic, etc. The main aim of a deposition is to extract information and understand the circumstances of the case. This information may be used for future trials.



(a) Deposition page showing start of Examination section (b) Deposition page showing start of Certificate section

Figure 3.1: Samples sections in a deposition

Deposition transcripts as seen in Fig. 3.1 typically follow a consistent format. It consists of segments such as “INDEX”, “EXHIBITS”, “EXAMINATION”, and “CERTIFICATIONS”

which can be separated. For this research, we focus on the “EXAMINATION” section. It consists of the conversation between the attorney(s) and the deponent. Legal depositions can come in various formats such as PDF, TXT, DOCX, and RTF. It becomes quite hard to implement a suite of programs to parse each format individually. For this reason, we use Apache Tika [58], which allows us to extract the meta-data and content of the files. [20] consists of more details on how the original parsing was done, but this work had some issues such as handling PDF files with images and handling larger sentences. In this section we aim to solve **RQ1**: *Can we improve the parsing of deposition transcripts?*

## 3.1 Processing PDF Images

Layouts for a deposition transcript generally follow one of the following two forms. One is the simple form in which each PDF page corresponds to 1 page only. In the condensed form, multiple pages of the PDF file are present in a single page. Fig. 3.2 shows an example of a page from simple PDF and condensed PDF files.

Since these PDF files are born digital, we can't select the text content within them directly through Apache Tika. We divided the task of parsing condensed PDF files into 3 parts: splitting original PDF document, splitting of pages within a single PDF page, and extracting text.

### Splitting original PDF document

In this step, we split each page of the condensed PDF document into separate pages. This was done with the help of pdf2image. pdf2image is a Python module that wraps the pdftoppm utility to convert PDF to PIL Image object [12]. We utilized the ‘convert\_from\_path()’

Deposition of		
<p>1 Lynch - Confidential 2 gone in this case? 3 A. One notice would go to -- well, you 4 have my notices there. I could go through the 5 distribution. 6 Q. I believe I have only one. I am 7 trying to find that one. 8 Let me ask you to take a look at 9 what's been marked as Plaintiff's Exhibit 28. 10 (Document marked Lynch Exhibit 28 for 11 identification, this date.) 12 Q. While you are looking at it, I will 13 find a copy for counsel. 14 MS. CECIL: I'm sorry. What number is 15 this? 16 MR. KLONTZ: 28. 17 Q. Do you recognize that document, 18 Mr. Lynch? 19 A. Yes, I do. 20 Q. Is this one of the five notices? 21 A. Yes, that's correct. 22 Q. And to whom did this notice go? 23 A. This went to employees of Philip 24 Morris Companies, Inc., Philip Morris 25 Incorporated (PM USA), and Philip Morris</p> <p style="text-align: center;">Equire Deposition Services 1 800 944-8424</p> <p>3990005890 3990005890 Source: https://www.industrydocuments.ucsf.edu/docs/fyp0183</p>	<p>10:57 1 Morris USA? 10:57 2 A. Yes, for 20 years. 10:57 3 Q. You anticipated my next question. And during that 10:58 4 20-year time frame, were you doing basically the same kinds 10:58 5 of work for Philip Morris USA directly that you're now doing 10:58 6 through the relationship between Altria Client Services and 10:58 7 Philip Morris USA? 10:58 8 A. Yes. I was also a senior principal scientist. I 10:58 9 was doing the same kind of work. It just got reorganized in 10:58 10 2006, so I started to do -- working for Altria Client 10:58 11 Services. 10:58 12 Q. Okay. And you explained to the jury that you have 10:58 13 a Ph.D. and the subject matter. Could you just give them 10:58 14 very briefly, your educational background leading up to 10:58 15 getting a Ph.D.? 10:58 16 A. I have a bachelor's of science degree in chemical 10:58 17 engineering from Washington University in St. Louis. And 10:58 18 after I finished that, I decided to get a Ph.D., and I went 10:58 19 to the University of California at Berkeley and got a Ph.D. 10:58 20 there. 10:58 21 Q. Okay. Now, Doctor, I've -- I've asked you -- you 10:58 22 know that this jury, last week, rendered a verdict in this 10:58 23 case -- in Phase I of this case, correct? 10:58 24 A. Yes. 10:58 25 Q. All right. Now, I want to focus your examination, VERITEXT REPORTING COMPANY 212-279-9424 www.veritext.com 212-495-3430</p> <p style="text-align: center;">0198</p> <p>10:59 1 year -- your testimony on the time frame beginning in -- 10:59 2 generally speaking, beginning in 1999 and coming forward to -- 10:59 3 present day, okay? 10:59 4 A. Okay. 10:59 5 Q. All right. Now, let me begin by asking you, as 10:59 6 the -- sort of one historical question. This jury has heard 10:59 7 the names of a number of people who were at Philip Morris 10:59 8 going back 30, 40, 50, 60 years. I'm going to call out some 10:59 9 names and just ask you if they're not at Philip Morris. 10:59 10 And if not, if there's been some name some name, okay? 10:59 11 A. Okay. 10:59 12 Q. The first is Robert Sotgiu. 10:59 13 A. He's not there. 10:59 14 Q. Myron Johnson. 11:00 15 A. Not there. 11:00 16 Q. Michael Washburn? 11:00 17 A. Not there. 11:00 18 Q. Bill Bush? 11:00 19 A. Not there. 11:00 20 Q. Tom Chalmers? 11:00 21 A. Not there. 11:00 22 Q. George Weinman? 11:00 23 A. Not there. 11:00 24 Q. Bill Gensler? 11:00 25 A. Not there.</p> <p style="text-align: center;">VERITEXT REPORTING COMPANY 212-279-9424 www.veritext.com 212-495-3430</p> <p>19 of 30 sheets Page 5138 to 5141 of 5183 Source: https://www.industrydocuments.ucsf.edu/docs/qjkd0224</p>	<p>11:00 1 Q. Howard Culham? 11:00 2 A. Not there. 11:00 3 Q. Dr. Joe Cullman. 11:00 4 A. No, not there. 11:00 5 Q. All done and all gone for quite some time? 11:00 6 A. Yes. 11:00 7 Q. All right. Can you tell us Philip Morris's 11:00 8 position on smoking and health issues and, in particular, 11:00 9 how Philip Morris makes statements about smoking and health 11:00 10 issues, including addiction to the addict today? 11:00 11 A. Well, you know, we agree that smoking is 11:00 12 addictive, that smoking causes disease, and we make those 11:00 13 statements on our corporate website. Philip Morris USA has 11:00 14 a website, and that's where you'll find our company 11:00 15 positions. 11:00 16 Q. All right. I think we have a -- 11:00 17 MR. KELLY: Michael, could you pull up PHU70333? 11:00 18 Q. And can you tell the jury if this is a screenshot 11:00 19 of the Philip Morris USA website? 11:00 20 A. Yes, it is. 11:00 21 Q. And that website readily available to anyone 11:00 22 who wants to access it on the internet? 11:00 23 A. Yes. 11:00 24 Q. Are there any restrictions on accessibility to that website. 11:00 25 A. There's no restrictions.</p> <p style="text-align: center;">VERITEXT REPORTING COMPANY 212-279-9424 www.veritext.com 212-495-3430</p> <p style="text-align: center;">0140</p> <p>11:01 1 Q. All right. Does the website contain Philip 11:01 2 Morris's position on health risks of smoking and whether 11:01 3 smoking is addictive? 11:01 4 A. Yes, it does. 11:01 5 Q. All right. Was Philip Morris required to create 11:01 6 this website? 11:01 7 A. No. This is not a requirement. Philip Morris 11:01 8 decided to do it in 1999. 11:01 9 Q. All right. I think -- can you tell the jury -- we 11:01 10 pulled out one of the statements from the website, is that 11:01 11 correct? 11:01 12 A. Yes. 11:01 13 Q. Is that readable to everybody? 11:01 14 Yes. You have it on your small screens. Okay. 11:01 15 Great. 11:01 16 Can you tell the jury what it says? 11:01 17 A. "There is no safe cigarette. Cigarettes are 11:01 18 addictive and cause serious diseases in smokers. For those 11:01 19 concerned about the health risks of smoking, the best thing 11:01 20 to do is quit." 11:01 21 Q. Has this been -- has this website been up and 11:01 22 running and available since 2007? 11:01 23 A. Yes. 11:01 24 Q. All right. Let me ask you this: Is there also a 11:01 25 site that provides the same information about whether or not</p> <p style="text-align: center;">VERITEXT REPORTING COMPANY 212-279-9424 www.veritext.com 212-495-3430</p> <p>07/05/2016 09:30:26 PM</p>

(a) Page from a normal deposition transcript (b) Page from a condensed deposition transcript

Figure 3.2: Types of deposition transcripts

method for the conversion.

## Splitting pages within a single PDF page

As seen in Fig. 3.2b each PDF page has 4 sub-pages. The location of these 4 sub-pages are the same for a particular deposition but may vary for different PDFs. We can leverage the borders of these sub-pages to identify each page and split them out. The box-detection algorithm mentioned in [81] is used to identify the horizontal and vertical lines, eventually a box with OpenCV [16]. The individual boxes sub-pages are stored as separate images.

5138

10:57 1 Morris USA?  
 10:57 2 A Yes, for 20 years.  
 10:57 3 Q You anticipated my next question. And during that  
 10:58 4 20-year time frame, were you doing basically the same kinds  
 10:58 5 of work for Philip Morris USA directly that you're now doing  
 10:58 6 through the relationship between Altria Client Services and  
 10:58 7 Philip Morris USA?  
 10:58 8 A Yes. I was also a senior principal scientist. I  
 10:58 9 was doing the same kind of work. It just got reorganized in  
 10:58 10 2008, so I started to do -- working for Altria Client  
 10:58 11 Services.  
 10:58 12 Q Okay. And you explained to the jury that you have  
 10:58 13 a Ph.D. and the subject matter. Could you just give them,  
 10:58 14 very briefly, your educational background leading up to  
 10:58 15 getting a Ph.D.?  
 10:58 16 A I have a bachelor's of science degree in chemical  
 10:58 17 engineering from Washington University in St. Louis. And  
 10:58 18 after I finished that, I decided to get a Ph.D., and I went  
 10:58 19 to the University of California at Berkeley and got a Ph.D.  
 10:58 20 there.  
 10:58 21 Q Okay. Now, Doctor, I've -- I've asked you -- you  
 10:59 22 know that this jury, last week, rendered a verdict in this  
 10:59 23 case -- in Phase I of this case, correct?  
 10:59 24 A Yes.  
 10:59 25 Q All right. Now, I want to focus your examination,  
 VERITEXT REPORTING COMPANY  
 212-279-9424 www.veritext.com 212-490-3430

(a) First subpage

5139

10:59 1 your -- your testimony on the time frame beginning in --  
 10:59 2 generally speaking, beginning in 1999 and coming forward to  
 10:59 3 present day. Okay?  
 10:59 4 A Okay.  
 10:59 5 Q All right. Now, let me begin by asking you, as  
 10:59 6 the -- sort of one historical question. This jury has heard  
 10:59 7 the names of a number of people who were at Philip Morris  
 10:59 8 going back 30, 40, 50, 60 years. I'm going to call out some  
 10:59 9 names and just ask you if they're still at Philip Morris.  
 10:59 10 And if not, if they've been gone quite some time. Okay?  
 10:59 11 A Okay.  
 10:59 12 Q The first is Robert Seligman.  
 10:59 13 A He's not there.  
 10:59 14 Q Myron Johnson.  
 11:00 15 A Not there.  
 11:00 16 Q Helmut Wakeham?  
 11:00 17 A Not there.  
 11:00 18 Q Bill Dunn?  
 11:00 19 A Not there.  
 11:00 20 Q Tom Osdene?  
 11:00 21 A Not there.  
 11:00 22 Q George Weissman?  
 11:00 23 A Not there.  
 11:00 24 Q Bill Gamble?  
 11:00 25 A Not there.  
 VERITEXT REPORTING COMPANY  
 212-279-9424 www.veritext.com 212-490-3430

(b) Second subpage

5140

11:00 1 Q Howard Cullman?  
 11:00 2 A Not there.  
 11:00 3 Q Or Joe Cullman.  
 11:00 4 A No, not there.  
 11:00 5 Q All done and all gone for quite some time?  
 11:00 6 A Yes.  
 11:00 7 Q All right. Can you tell us Philip Morris's  
 11:00 8 position on smoking and health issues and, in particular,  
 11:00 9 how Philip Morris makes statements about smoking and health  
 11:00 10 issues, including addiction to the public today?  
 11:00 11 A Well, you know, we agree that smoking is  
 11:00 12 addictive, that smoking causes disease, and we make those  
 11:00 13 statements on our corporate website. Philip Morris USA has  
 11:00 14 a website, and that's where you'll find our company  
 11:00 15 positions.  
 11:00 16 Q All right. I think we have a --  
 11:00 17 MR. REILLY: Michael, could you pull up PMU71031?  
 11:01 18 Q And can you tell the jury if this is a screenshot  
 11:01 19 of the Philip Morris USA website?  
 11:01 20 A Yes, it is.  
 11:01 21 Q And is that website readily available to anyone  
 11:01 22 who wants to access it on the Internet?  
 11:01 23 A Yes.  
 11:01 24 Q No restrictions on accessibility to that website.  
 11:01 25 A There's no restrictions.  
 VERITEXT REPORTING COMPANY  
 212-279-9424 www.veritext.com 212-490-3430

(c) Third subpage

5141

11:01 1 Q All right. Does the website contain Philip  
 11:01 2 Morris's position on health risks of smoking and whether  
 11:01 3 smoking is addictive?  
 11:01 4 A Yes, it does.  
 11:01 5 Q All right. Was Philip Morris required to create  
 11:01 6 this website?  
 11:01 7 A No. This is not a requirement. Philip Morris  
 11:01 8 decided to do it in 1999.  
 11:01 9 Q All right. I think -- can you tell the jury -- we  
 11:01 10 pulled out one of the statements from this website; is that  
 11:01 11 correct?  
 11:01 12 A Yes.  
 11:01 13 Q Is that readable to everybody?  
 11:02 14 Yes. You have it on your small screens. Okay.  
 11:02 15 Great.  
 11:02 16 Can you tell the jury what it says?  
 11:02 17 A "There is no safe cigarette. Cigarettes are  
 11:02 18 addictive and cause serious diseases in smokers. For those  
 11:02 19 concerned about the health risks of smoking, the best thing  
 11:02 20 to do is quit."  
 11:02 21 Q Has this been -- has this website been up and  
 11:02 22 running and available since 2000?  
 11:02 23 A Yes.  
 11:02 24 Q All right. Let me ask you this: Is there also a  
 11:02 25 site that provides the same information about whether or not  
 VERITEXT REPORTING COMPANY  
 212-279-9424 www.veritext.com 212-490-3430

(d) Fourth subpage

Figure 3.3: Splitting the 4 sub-pages in Fig. 3.2b using box detection

## Extracting Text

Now the task is simplified to OCR conversion. There are a variety of OCR systems such as Tesseract-OCR, GOCR, and OCRopus, but they did not perform well on the given image. In some cases, due to the relatively poor quality of the image, the system couldn't extract words/letters, making the statements erroneous. The OCR systems also found it hard to process each line within the image, i.e., it couldn't recognize the "line number, Q./A., Statement" format, making the parsed output unusable.

Out of the tried OCR systems, we found that Amazon Textract (see Section 2.2.3) gave the best performance. So we uploaded the images on Amazon S3 servers and processed them through Amazon Textract. This returned to us a text file containing the contents of the image. All these text files are appended together to get the whole content of the deposition. Some post-processing is required on the file to convert it to the required format. Fig. 3.4 shows the final text file we obtained.

The screenshot displays a text editor window with three panes. The left pane shows lines 1 through 28, the middle pane shows lines 29 through 63, and the right pane shows lines 61 through 104. The text consists of a series of questions (Q) and answers (A) regarding Philip Morris USA, including topics like the company's history, website accessibility, and health issues related to smoking. The text is formatted with line numbers on the left and Q/A labels at the start of each line.

```

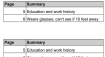
1 5138
2 1 Morris USA?
3 2 A Yes, for 20 years.
4 3 Q You anticipated my next question. And during
  that
5 4 20-year time frame, were you doing basically
  the same kinds
6 5 of work for Philip Morris USA directly that
  you're now doing
7 6 through the relationship between Altria Client
  Services and
8 7 Philip Morris USA?
9 8 A Yes. I was also a senior principal scientist. I
10 9 was doing the same kind of work. It just got
  reorganized in
11 10 2008, so I started to do -- working for Altria
  Client
12 11 Services.
13 12 Q Okay. And you explained to the jury that you
  have
14 13 a Ph.D. and the subject matter. Could you just
  give them,
15 14 very briefly, your educational background
  leading up to
16 15 getting a Ph.D.?
17 16 A I have a bachelor's of science degree in
  chemical
18 17 engineering from Washington University in St.
  Louis. And
19 18 after I finished that, I decided to get a
  Ph.D., and I went
20 19 to the University of California at Berkeley
  and got a Ph.D.
21 20 there.
22 21 Q Okay. Now, Doctor, I've -- I've asked you --
23 22 know that this jury, last week, rendered a
  verdict in this
24 23 case -- in Phase I of this case, correct?
25 24 A Yes.
26 25 Q All right. Now, I want to focus your
  examination,
27 5139
28 1 your -- your testimony on the time frame
  beginning in --
29 2 generally speaking, beginning in 1999 and
  coming forward to
30 3 present day. Okay?
31 4 A Okay.
32 5 Q All right. Now, let me begin by asking you, as
33 6 the -- sort of one historical question. This
  jury has heard
34 7 the names of a number of people who were at
  Philip Morris
35 8 going back 30,40, 50, 60 years. I'm going to
  call out some
36 9 names and just ask you if they're still at
  Philip Morris.
37 10 And if not, if they've been gone quite some
  time. Okay?
38 11 A Okay.
39 12 Q The first is Robert Seligman.
40 13 A He's not there.
41 14 Q Myron Johnson.
42 15 A Not there.
43 16 Q Helmut Wakeham?
44 17 A Not there.
45 18 Q Bill Dunn?
46 19 A Not there.
47 20 Q Tom Osciene?
48 21 A Not there.
49 22 Q George Weissman?
50 23 A Not there.
51 24 Q Bill Gamble?
52 25 A Not there.
53 5140
54 1 Q Howard Cullman?
55 2 A Not there.
56 3 Q Or Joe Cullman.
57 4 A No, not there.
58 5 Q All done and all gone for quite some time?
59 6 A Yes.
60 7 Q All right. Can you tell us Philip Morris's
61 8 position on smoking and health issues and, in
  particular,
62 9 how Philip Morris makes statements about
  smoking and health
63 10 issues, including addiction to the public today?
61 8 position on smoking and health issues and, in particular,
62 9 how Philip Morris makes statements about smoking and health
63 10 issues, including addiction to the public today?
64 11 A Well, you know, we agree that smoking is
65 12 addictive, that smoking causes disease, and we make those
66 13 statements on our corporate website. Philip Morris USA has
67 14 a website, and that's where you'll find our company
68 15 positions.
69 16 Q All right. I think we have a --
70 17 MR. REILLY: Michael, could you pull up PM071031?
71 18 Q And can you tell the jury if this is a screenshot
72 19 of the Philip Morris USA website?
73 20 A Yes, it is.
74 21 Q And is that website readily available to anyone
75 22 who wants to access it on the Internet?
76 23 A Yes.
77 24 Q No restrictions on accessibility to that website.
78 25 A There's no restrictions.
79 5141
80 1 Q All right. Does the website contain Philip
81 2 Morris's position on health risks of smoking and whether
82 3 smoking is addictive?
83 4 A Yes, it does.
84 5 Q All right. Was Philip Morris required to create
85 6 this website?
86 7 A No. This is not a requirement. Philip Morris
87 8 decided to do it in 1999.
88 9 Q All right. I think » can you tell the jury » we
89 10 pulled out one of the statements from this website; is that
90 11 correct?
91 12 A Yes.
92 13 Q Is that readable to everybody?
93 14 Yes. You have it on your small screens. Okay.
94 15 Great.
95 16 Can you tell the jury what it says?
96 17 A "There is no safe cigarette. Cigarettes are
97 18 addictive and cause serious diseases in smokers. For those
98 19 concerned about the health risks of smoking, the best thing
99 20 to do is quit."
100 21 Q Has this been » has this website been up and
101 22 running and available since 2000?
102 23 A Yes.
103 24 Q All right. Let me ask you this: Is there also a
104 25 site that provides the same information about whether or not
  
```

Figure 3.4: Final text file obtained after processing the condensed PDF page given in Fig. 3.2b

## 3.2 Splitting Larger Sentences

The efficacy of the parser is largely dependent on the quality of the OCR system and the quality of the PDF. The tobacco documents (see Section 4.2) have the aforementioned problems. These PDFs are already OCRed, but the OCR system has identified text haphazardly. It identifies line numbers first and then the set of questions and answers. In some cases due to the poor quality of the PDF, it misdiagnoses characters making it even more difficult to parse the content, causing some sets of QA-pairs to be quite long. In such instances, we observed that there are many sets of questions and answers but they do not have the identifiers “Q.” or “A.”.

Table 3.1: Example where the question consists of multiple questions and answers.

 <p><b>Question</b></p>	<p>Now, you testified in June last year about your duties with Philip Morris. Have your duties changed Esquire Deposition Services Lynch - Confidential substantially in your employment since June 2001? June prepare certain today’s No. You testified also that prior to that deposition, you met with counsel to for this deposition, and you reviewed documents. Do you recall testifying about that? Yes. Did you meet with counsel prior to deposition? three and the firm. twice I Yes. And how many times and for how long? I believe four or five times, between five hours each time. And with whom were you meeting? Cynthia Cecil and one other member of Of her firm? Yes. Do you remember who that was? The name – it was only once or don’t recall the name. Where were those meetings? Esquire Deposition Services Lynch - Confidential</p>
<b>Answer</b>	At the Hunton, Williams offices in New York
<b>Question-DA</b>	bin-d
<b>Answer-DA</b>	sno

Splitting just on “?” and “.” is not enough as a question-answer may have a combination of both punctuation marks. We identified that the most common pattern is “sentence. question? answer.”. Table 3.1 shows an example where the parser fails. There are multiple sets of QA pairs within the question itself.

To solve this issue, we tokenize each statement in the question and identify the “sentence. question? answer.” pattern. Fig. 3.5 shows the identified QA pairs within the question itself. For the last question, we append the actual answer to complete it. Each of the highlighted QA pairs in Fig. 3.5 was broken up into individual QA pairs as can be seen in Table 3.2. This allowed us to extract more information than what was originally present.

Now, you testified in June last year about your duties with Philip Morris. Have your duties changed Esquire Deposition Services Lynch - Confidential substantially in your employment since June 2001? June prepare certain today's No. You testified also that prior to that deposition, you met with counsel to for this deposition, and you reviewed documents. Do you recall testifying about that? Yes. Did you meet with counsel prior to deposition? three and the firm. twice I Yes. And how many times and for how long? I believe four or five times, between five hours each time. And with whom were you meeting? Cynthia Cecil and one other member of Of her firm? Yes. Do you remember who that was? The name -- it was only once or don't recall the name. Where were those meetings? Esquire Deposition Services Lynch - Confidential At the Hunton, Williams offices in New York.

Figure 3.5: Long sentence division

An alternative to this solution is to use Amazon Textract [2]. The vast algorithms of Amazon Textract allow it the ability to clearly understand and recognize the characters in the document. We upload the PDF file to an Amazon S3 server and run it through Textract, subsequently obtaining a text file that has the file contents. This text file is processed to prepare the contents in a format accepted by the pipeline, through which further processing may be done.

Table 3.2: Processing extra long QA pair given in Table 3.1.

Question	Answer
Now, you testified in June last year about your duties with Philip Morris. Have your duties changed Esquire Deposition Services Lynch - Confidential substantially in your employment since June 2001?	June prepare certain today's No.
You testified also that prior to that deposition, you met with counsel to for this deposition, and you reviewed documents. Do you recall testifying about that?	Yes.
Did you meet with counsel prior to deposition?	three and the firm.
twice I Yes. And how many times and for how long?	I believe four or five times, between five hours each time.
And with whom were you meeting?	Cynthia Cecil and one other member of Of her firm?
	Yes.
Do you remember who that was?	The name – it was only once or don't recall the name.
Where were those meetings?	At the Hunton, Williams offices in New York.

### 3.3 Extracting Page and Line Numbers

Depositions transcripts come in varying sizes. Some can be quite small – under 50 pages – while others can be very long such as 250 pages. Even for smaller depositions, there is a great deal of content to be read and understood by attorneys. This process may sometimes be cumbersome and time-consuming. One solution is that attorneys may read a summarized version of the deposition, giving them the crux of the matter.

There are various ways in which a deposition can be summarized. One is by condensing the whole transcript into a few paragraphs or pages pertaining to the crux of the topic. Another common form of summarization is the page and line number summary, which is a page-by-page summary of a transcript having page numbers and important line numbers. Another alternative is the page only summary, in which there are page numbers and the summary of

that page. Fig. 3.6 shows examples of different types of summaries.

He was vague in his description of how he was hurt and testified to details of the vessel that are completely incorrect, apparently having confused the [REDACTED]. He was plainly evasive when questioned about [REDACTED] though this is not likely to come up at trial. However, unless we can develop solid independent evidence of [REDACTED] close proximity to when he was working for the insured, or independent corroboration that he only handled soft lines on the job, it is probable that a jury would conclude that [REDACTED] resulted from his work [REDACTED].

(a) Snippet of a basic summary

Page	Summary
5	Education and work history
6	Wears glasses; can't see if 10 feet away.

(b) Snippet of page summarized deposition [31]

Page	Line	Summary
5	10-20	Graduated from high school in 1975. Took welding classes at votech school. Worked as a welder from 1977 to present.
6	1-5	Wears glasses; can't see anything if more than ten feet away.

(c) Snippet of page & line number summary [31]

Figure 3.6: Sub-figures show different ways a transcript can be summarized. Some parts of the figures have been blacked out to preserve anonymity.

Extracting line numbers and page numbers can help in performing summarization tasks, down the line. For each QA pair, we store 4 extra parameters in the JSON file [20]: starting line number, ending line number, starting page number, and ending page number. From the deposition transcript (Fig. 1.1) we observed that each line starts with a line number and that the line numbers do not exceed 25. So, we defined a regular expression that identifies one and two-digit numbers, coming at the start of a line followed by “Q.” or “A.”. The starting line number stores the line number when the question starts, while the ending line number stores the line number when the answer ends. Similarly, the starting page number stores the page number when the question starts and the ending page number stores the page number when the answer ends.

```
{
  "question": "All right. Now, let me begin by asking you, as
the -- sort of one historical question. This jury has heard
the names of a number of people who were at Philip Morris
going back 30, 40, 50, 60 years. I'm going to call out some
names and just ask you if they're still at Philip Morris.
And if not, if they've been gone quite some time. Okay?",
  "answer": "Okay.",
  "start_line_number": 5,
  "start_page_number": 10,
  "end_line_number": 11,
  "end_page_number": 10
},
{
  "question": "The first is Robert Seligman.",
  "answer": "He's not there.",
  "start_line_number": 12,
  "start_page_number": 10,
  "end_line_number": 13,
  "end_page_number": 10
},
{
  "question": "Myron Johnson.",
  "answer": "Not there.",
  "start_line_number": 14,
  "start_page_number": 10,
  "end_line_number": 15,
  "end_page_number": 10
},
{
  "question": "Helmut Wakeham?",
  "answer": "Not there.",
  "start_line_number": 16,
  "start_page_number": 10,
  "end_line_number": 17,
  "end_page_number": 10
},
}
```

Figure 3.7: Final JSON file output showing page number and line number variables

Handling page numbers is much more tricky than line numbers. There isn't any specific pattern or form in which page numbers can be identified. We have previous knowledge that the question-answer part of the deposition starts in the "EXAMINATION" section, so we count the number of lines till then and divide by 25 to identify the initial page number. Typically, it is within the first 5 pages of the document. Using this initial page number we increased the page number when we find that the previous starting line number is less than the subsequent starting line number. Fig. 3.7 shows a snippet of the final JSON file used to process the QA pairs. Each QA pair has its values for starting and ending line numbers & page numbers.

# Chapter 4

## Data

This chapter describes the different datasets used in this work. We utilize the parsing techniques and classification methods implemented in [20].

### 4.1 Proprietary Dataset

This dataset is provided to us by Mayfair Group LLC. It consists of a collection of around 350 depositions, each related to an accident or injury case. We use a small subset of this dataset for performing the tasks mentioned in the subsequent sections.

#### **M3 dataset**

This dataset consists of 3 long depositions which consist of an average of 1500 QA pairs. We mainly use this dataset to develop the initial rules for transformation.

#### **M10 dataset**

This dataset consists of 10 randomly selected depositions of varying sizes. Combined they contain approximately 4000 question-answer pairs. This dataset is used for training deep learning models and evaluating the generated rules.

### 4.1.1 Dataset Observations

Table 4.1 shows the top 20 question-answer DA pairs in the M10 dataset. The top 20 class pairs account for about 70% of the dataset, showing how varied are the types of questions and answers. Fig. 4.1 represents the same data in the form of a pie chart. From the visualization, we observe that the top 3 classes are [wh, sno], [bin, y] and [bin-d, y], which account for approximately 30% of the dataset.

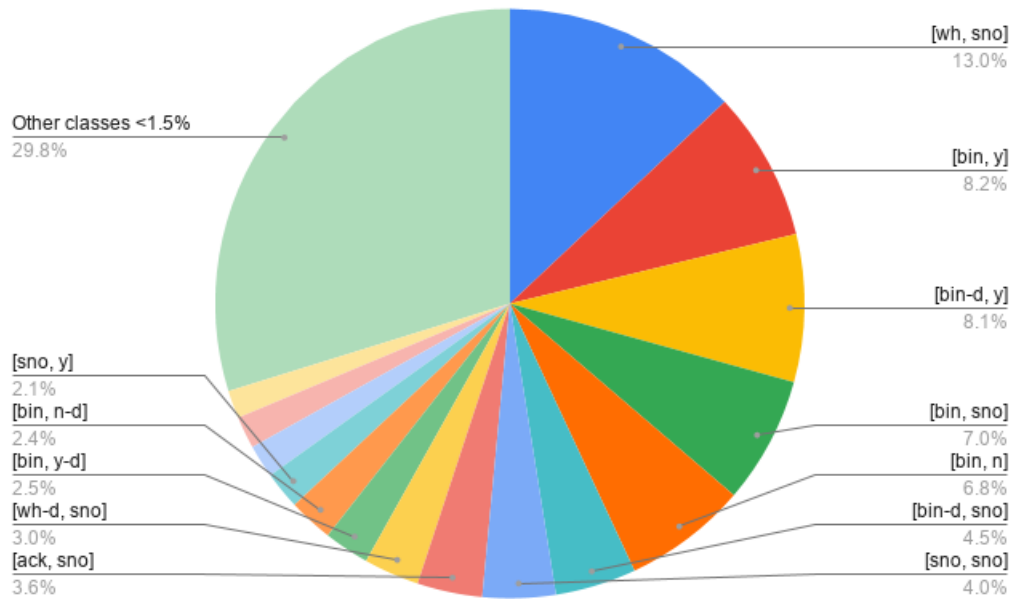


Figure 4.1: Visualizing the distribution of QA pairs of M10 dataset

Typically, in accident and injury cases, there are questions on personal information and prior physical conditions. These are accompanied with questions regarding the incident that occurred (time, date, cause, etc.) and the extent of the injury. There may also be questions regarding the medical bills, as well as long-term physical and fiscal impact. Many such questions are asked to corroborate the deponent’s story and validate all of the details of the accident. Due to this there are many “yes”, “no” type questions. Questions pertaining to the incident are more detailed, asking specific points such as “when”, “where”, and “how”,

falling under the wh category of questions. This is why we observe such a high number of samples in such DA classes. Due to the extensive nature of classes defined for questions and answers, there are many DA pair combinations possible, which crop up in depositions. From the “Other classes” we observe that there are many DA pairs surfacing, which generally are less frequent in nature.

Table 4.1: Top 20 Question-Answer pairs for the M10 dataset. [21]

Question-DA	Answer-DA	# of samples	% of Total
wh	sno	517	13.00
bin	y	326	8.20
bin-d	y	322	8.10
bin	sno	277	6.96
bin	n	270	6.79
bin-d	sno	177	4.45
sno	sno	159	4.00
ack	sno	142	3.57
wh-d	sno	121	3.04
bin	y-d	99	2.49
bin	n-d	97	2.44
sno	y	82	2.06
bin	dno	72	1.81
y	sno	70	1.76
wh	dno	61	1.53
bin-d	ack	53	1.33
sno	ack	53	1.33
bin-d	y-d	51	1.28
nu	sno	45	1.13
nu	y	40	1

Questions and answers can be of various lengths, which may affect the transformation quality. Typically the average length of a sentence is around 15 to 20 words. For this work, we consider sentences having 10 words or less to be short statements. Sentences having 10 to 25 words are medium-length statements. Finally, sentences longer than 25 words are long sentences. We find the approximate word count in questions and answers individually, by checking the number of “spaces”. These numbers and the aforementioned categories are used to classify

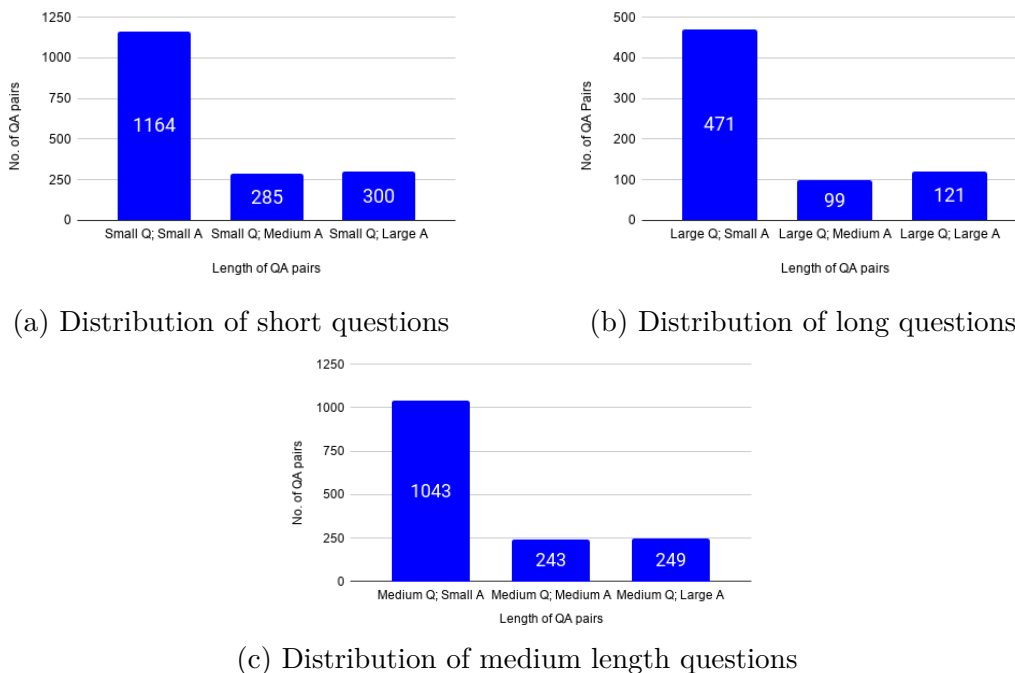


Figure 4.2: Distribution of QA pairs based on their length

the QA pair into categories seen in Fig. 4.2. It was observed that most of the answers are smaller in nature (check small A in Fig. 4.2), which accounts for 67% of the dataset. The number of medium and long answer questions are quite similar. Considering the length of questions, long questions only account for about 17% of the dataset seen in Fig. 4.2b.

Table 4.2: Top 7 important DA classes w.r.t. summaries.

Question-DA	Answer-DA
wh	sno
bin-d	y
bin	y
bin	n
bin	sno
sno	sno

Along with this work, there has been parallel research on the summarization of legal depositions. For that work, some paralegals were hired to annotate QA pairs and rate the importance of each with respect to the summary. This rating was between 1 to 5. We

analyzed these ratings to identify which DA classes are more important, namely counting the number of rating greater than or equal to 4. We observed that [wh, sno] QA pairs are most important. Table 4.2 shows the top 7 important classes.

## 4.2 Tobacco Dataset

Truth Tobacco Industry Documents [52] is an archive of 14 million documents related to tobacco companies about their advertising, manufacturing, marketing, scientific research, and political activities. It also consists of legal documents between the US states and the 7 major tobacco companies. This archive was created and has been maintained by the University of California, San Francisco. Out of the 14 million documents, around 20,000 documents are related to deposition transcripts and/or trial transcripts. We use a small subset of these documents to perform experiments and analyze the results. We selected 8 random depositions giving us a total of around 3300 QA pairs to work with.

### 4.2.1 Dataset Observations

Table 4.3 shows the top 20 question-answer DA pairs in the tobacco dataset. The top 20 class pairs account for about 80% of the dataset, showing how varied the types of questions and answers are. Fig. 4.3 represents the same data in the form of a pie chart. From the visualization, we observed that the top 3 classes are [bin-d, sno], [bin, sno], and [wh, sno]. Together these account for approximately 35% of the dataset.

There is a large variety of depositions available in this dataset. It ranges from people with health issues to expert witness depositions. Similar to injury and accident conditions, in such depositions, there are questions pertaining to personal information, long-term health

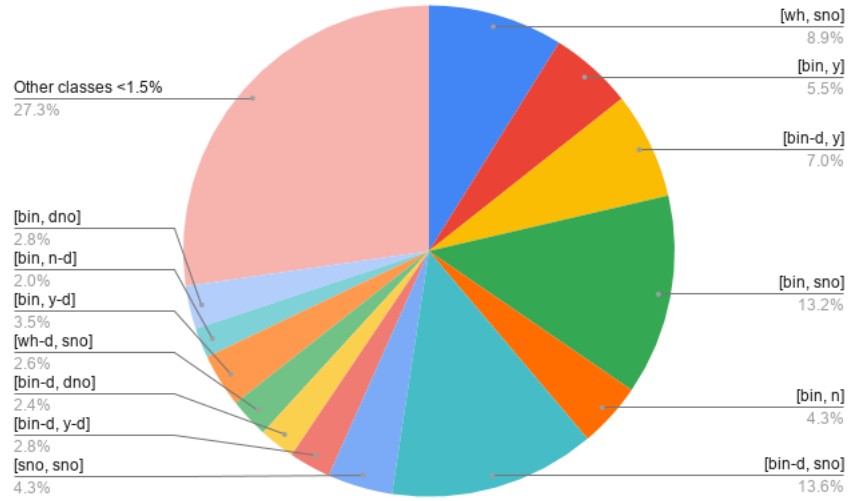


Figure 4.3: Visualizing the distribution of QA pairs of tobacco dataset

Table 4.3: Top 20 Question-Answer pairs for the tobacco dataset. [52]

Question-DA	Answer-DA	# of samples	% of Total
bin-d	sno	454	13.58
bin	sno	441	13.19
wh	sno	297	8.88
bin-d	y	235	7.02
bin	y	183	5.47
bin	n	143	4.27
sno	sno	143	4.27
bin	y-d	118	3.52
bin	dno	95	2.84
bin-d	y-d	92	2.75
wh-d	sno	87	2.60
bin-d	dno	79	2.36
bin	n-d	66	1.97
bin-d	n-d	47	1.40
wh	dno	45	1.34
ack	sno	35	1.04
bin-d	bin-d	34	1.01
bin-d	n	32	0.95
bin	so	29	0.86
bin	bin-d	28	0.83

impact, and medical bills. Depositions of expert witnesses follow a slightly different path. The deponent is an expert on a specific subject matter, whose expertise may be relevant in the legal proceedings. For example, in legal proceedings against tobacco companies, expert witnesses were people knowing the contents of smoking products, such as cigarettes. In such instances the deponent is generally asked for specific details on the subject matter, such as “Is nicotine harmful to the body?” The deponent replies with their opinion and findings. In many cases, they simply have to confirm whether the condition or case presented by the questioner is possible or not. This causes many questions to be of the “bin” or “wh” type form.

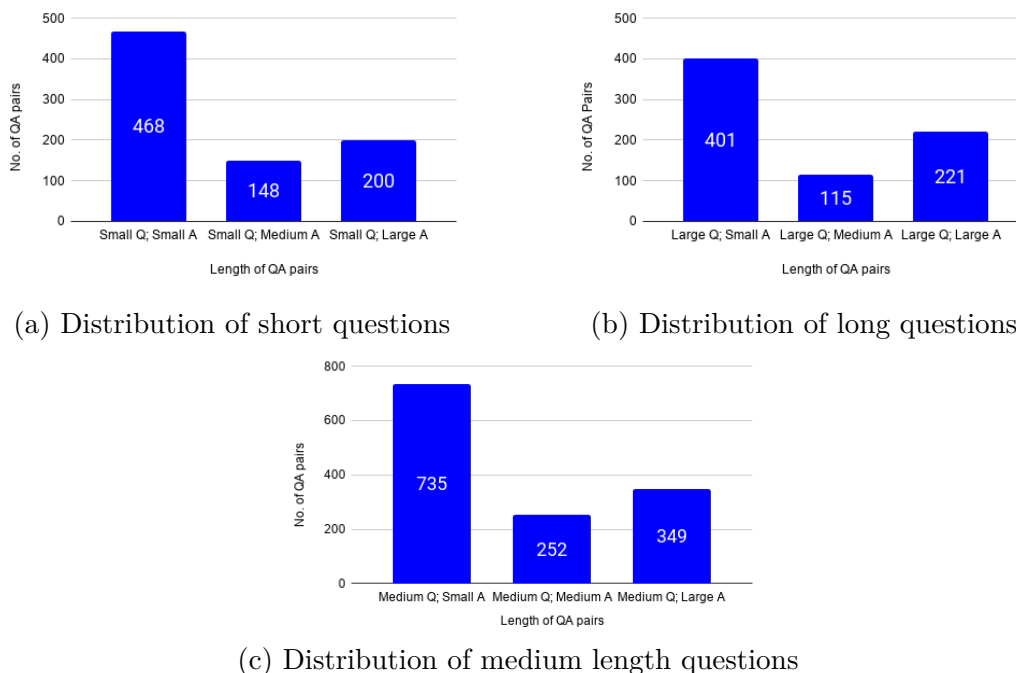


Figure 4.4: Distribution of QA pairs based on their length for tobacco dataset

We observed that the quality of these documents wasn’t great, such as when it was hard for the system to read and recognize characters. The classifier we used to identify the DA classes of each question and answer did have some difficulty in recognizing the [bin-d, sno] class, and we found some cases of mis-classification. Due to the variation within the [bin-d,

sno] class we do not use these QA pairs for the remainder of this work.

Similar to the M10 dataset, we organize the questions and answers, individually, based on length. Statements under 10 words are considered short, while statements longer than 25 words are considered as long sentences. All other sentences are considered as medium-length sentences. We find the approximate word count in questions and answers individually, by checking the number of “spaces”.

We create 9 different categories based on the combination of the length of questions and answers and categorize them. Fig. 4.4 shows the distribution of the categories. We noticed that nearly 50% of the questions fell under the medium-length category across the 8 depositions, seen in Fig. 4.4b. We also noticed that most of the answers by the deponents were short, similar to the M10 dataset. This could be expected due to the larger number of binary questions asked. The questioner possibly asked questions that didn’t need any extra explanation by the deponent. From Fig. 4.4, we can also observe that the distribution of shorter and longer questions are quite similar to the M10 dataset. It can also be noted that the distributions across the length of the answers are alike.

### 4.3 Ground Truth Annotation

Three different datasets were annotated, M3 dataset, M10 dataset (see Section 4.1), and the tobacco dataset (see Section 4.2). The M3 dataset was annotated by a group of paralegals who were selected by Mayfair Group LLC. They had the requisite knowledge of the relevant proceedings, and were asked to combine the QA pairs, with their best judgement. We spotted many instances where the paralegals cross-referenced some previously discussed topics or subject-matters in their annotated sentences. I, along with my co-authors in [21] and others selected by Mayfair, annotated the other two datasets. This involved reading the question-

answer pairs and suitably combining them into their canonical form. While annotating we considered each QA pair to be independent of others, i.e., without any cross-reference of information. These annotated sentences were considered to be the ground truth during evaluation.

Table 4.4: Annotated DA-pair classes for M10 dataset.

Qstn-DA	Ans-DA	Qstn-DA	Ans-DA	Qstn-DA	Ans-DA
ack	sno	bin	nu	qo	sno
bin	ack	bin	sno	sno	ack
bin-d	ack	bin	y	sno	dno
bin-d	dno	bin	y-d	sno	n
bin-d	n	dno	sno	sno	sno
bin-d	n-d	nu	n	sno	y
bin	dno	nu	sno	wh	dno
bin-d	y	nu	y	wh-d	sno
bin-d	y-d	or	sno	wh	n
bin	n	or	y	wh	sno
bin	n-d	qo	n	y	sno

Table 4.4 and Table 4.5 exhibit the DA pairs that have been annotated for the proprietary and tobacco dataset, respectively. We did more extensive work on the M10 dataset, which can be seen by the number of DA pair files that have been annotated. The DA pair files were decided based on the combination of frequency and importance in the summary. We limited the number of DA pairs annotated to the most important DA pairs in the tobacco dataset as it was mainly used to check the generalizability of the transformation rules. As mentioned previously, the quality of the depositions is not good, which causes some content to be misread by the parser. Many times the “Q.” or “A.” delimiters are not identified. This causes some QA pairs to combine with others, confusing the DA classifier, so inputs are misclassified. We found quite a few instances of such cases within the [bin-d, sno] class. Introducing a new parser could help resolve this issue.

Table 4.5: Annotated DA-pair classes for tobacco dataset.

Qstn-DA	Ans-DA	Qstn-DA	Ans-DA	Qstn-DA	Ans-DA
ack	sno	bin	n	sno	sno
bin	dno	bin	sno	wh-d	sno
bin-d	y	bin	y-d	wh	sno
bin	n-d	bin	y		

## 4.4 MTurk dataset

In addition to the experiments done with the proprietary and tobacco datasets with our manual annotation, we also wanted to integrate the observations and results from a layman’s perspective. For this, we utilize turkers on Amazon Mechanical Turk. They were given two different tasks to perform: (1) Annotation of QA pairs and (2) Rating the Canonical Sentences. For both of these tasks, two different datasets were curated. More details can be found further in this section as well as in Section 6.6 and Appendix E.

Table 4.6: Distribution of the MTurk dataset curated for Task 1.

Qstn-DA	Ans-DA	# of samples
wh	sno	209
bin	y	144
bin	sno	82
bin	n	65
sno	sno	57
bin-d	y	54
ack	sno	43
bin	y-d	36
bin	dno	31
wh	dno	28
bin-d	sno	26
sno	y	25
nu	sno	21
bin	n-d	20
.	.	.
<b>Total</b>		<b>1116</b>

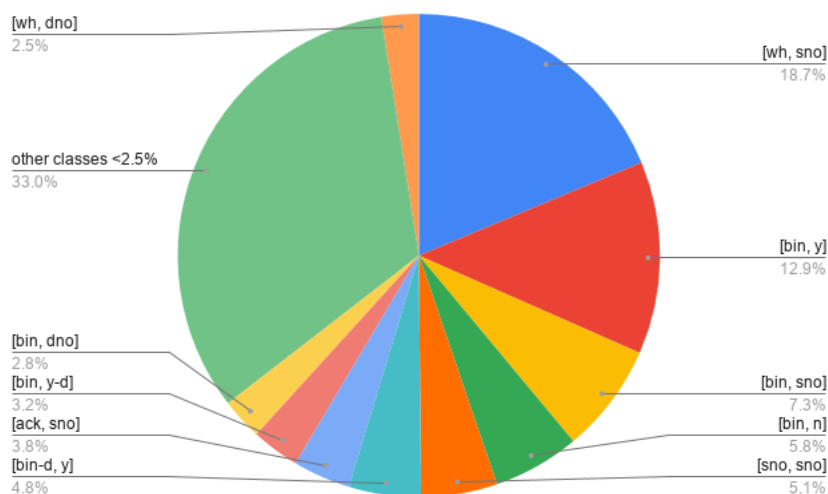


Figure 4.5: Visualizing the distribution of MTurk dataset for Task 1

Table 4.6 and Fig. 4.5 display the distribution of the DA class pairs of the questions and answers used for annotation on AMT. For this task, we collect 1116 samples from turkers. We curate a majority of the sentences from the M3 dataset with a small number of sentences from the tobacco dataset. We observed that the distribution is again quite similar. There are many [wh, sno] questions and many binary questions that have a simple “yes” or “no” answer.

Table 4.7 and Fig. 4.6 display the distribution of the DA class pairs of the questions and answers used for rating the canonical sentences. For this task, we collect 1200 samples across 400 QA pairs from turkers. The 400 QA pairs were curated from the M3 dataset. We observed that this dataset consists of a large percentage of “wh” type questions with long answers. We also observed that in the curated dataset, we have more “yes” answers than “no” ones. As this dataset is curated from the same original dataset as the previous dataset for Task 1, here also, there are a large number DA class pair combinations with smaller frequencies. Such pairs collectively account for 30% of the dataset, which is around 120 QA pairs.

Table 4.7: Distribution of the MTurk dataset curated for Task 2.

Qstn-DA	Ans-DA	# of samples
wh	sno	71
bin	y	60
bin	sno	42
bin-d	y	34
bin-d	sno	31
ack	sno	15
bin	y-d	9
sno	sno	9
wh	dno	9
bin	n	7
nu	y	7
or	sno	7
wh-d	sno	7
.	.	.
<b>Total</b>		<b>400</b>

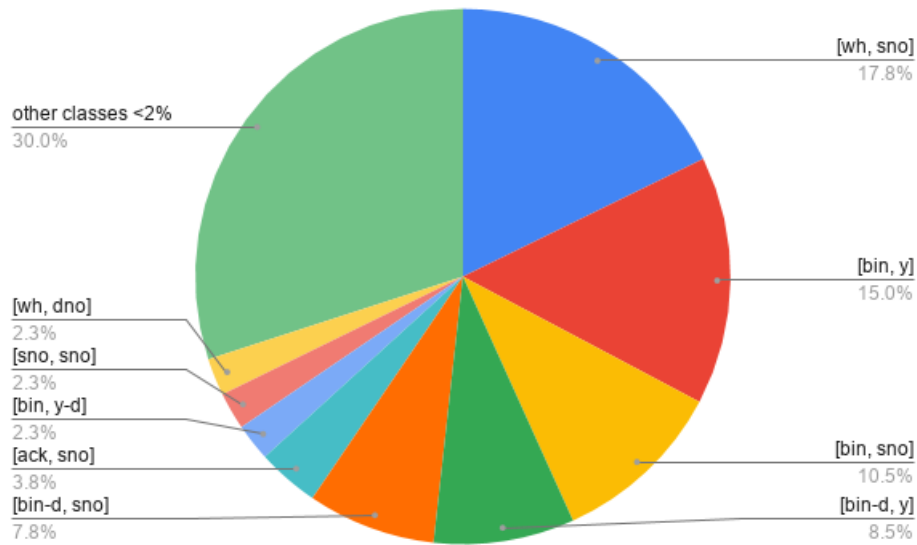


Figure 4.6: Visualizing the distribution of MTurk dataset for Task 2

# Chapter 5

## Transformational Techniques

We aim to combine the important parts of the question and answer coherently in a lexically correct manner. In doing so, identifying dialog acts (DAs) is an important step, which allows us to identify and group similar types of QA pairs. These grouped QAs can then be analyzed and grammatical rules can be developed accordingly. For this work, we have used the ontology defined in [20] to classify DAs in our dataset. The work in this section explains how we developed rules to combine the questions and answers. We call this as transformation to declarative or canonical form.<sup>1</sup> In the later part of this chapter, we describe how we aim to use deep learning for such transformations. In earlier sections we aim to solve: **RQ2:** *Can we combine QA pairs into their canonical form?* and **RQ3:** *Can sentence correction be used to improve the development of sentences?*

### 5.1 Pre-processing of Questions

After the classification step mentioned above, we had a CSV file with fields as id, text, and dialog act. Combing through the file, we observed that the text was noisy. The questions possessed extra words which, if removed, would facilitate a better transformation.

Table 5.1 shows some examples of noise found in questions. There was a common set of noisy phrases observed, so we created a dictionary of them. We used regular expressions to

---

<sup>1</sup>Many parts of this section overlap with work done in the course “CS6604: Digital Libraries” and in [21].

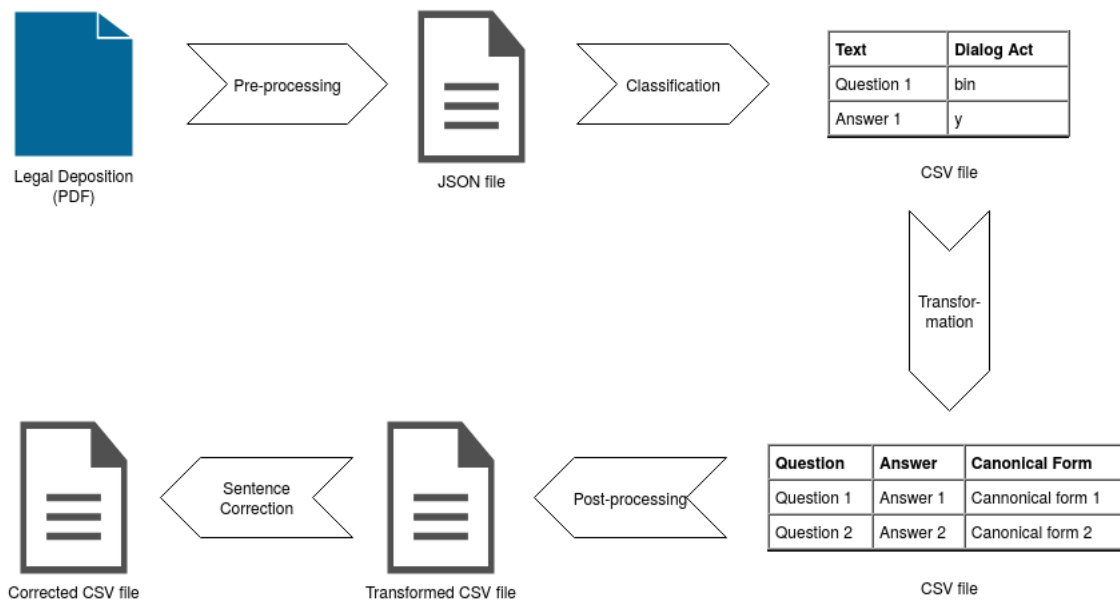


Figure 5.1: Pipeline which we aim to implement

Table 5.1: Question-Answer pairs having noise (highlighted in bold).

Question	Answer
<b>Okay.</b> Does your business maintain a website?	Not that I know of.
<b>Okay.</b> And what about – North Carolina, you have one because you’ve lived here and worked here; <b>is that correct?</b>	Correct.
<b>All right. So – and</b> this next document, is this also part of your prep for the deposition today?	Yes.

identify and remove them from the questions.

### 5.1.1 Processing QAs with Extra Statements

Some dialog acts are much more complicated than others. In QA pairs having dialog act pairs as [bin,y] or [bin,n], the question and answers are mostly one-liners or very short. In QA pairs having dialog act pairs as [bin-d, y] or [bin-d, n], the questions are quite long and typically consist of an extra statement or two, while the answer is still a simple “yes” or “no”.

Similar can be QA pairs having dialog acts [bin, y-d] or [bin, n-d]. The question is a one-liner or quite short (binary) while the answer consists of a “yes”/“no” followed by a set of statements explaining why.<sup>2</sup>

Table 5.2: Example of QA pairs with extra statements.

Question	Qstn-DA	Answer	Ans-DA
All right. And does he have his own staff?	bin	Yes.	y
Okay. Do you have a website?	bin	No.	n
So he used to be your partner?	bin	Yeah. He’s – well, I mean – now, I know that he just works under SVI or his company.	y-d
So while Mr. Arturo is elevated up in the bin lift, the first cable broke. Is that the left cable that you were referring to?	bin-d	I believe so, yes, sir.	y

Table 5.2 gives some example QA pairs having extra statements. To process questions with well-formed sentences, we break the question text into parts, where the first part is the extra statement(s) while the other is the actual question itself. The actual question and answer are processed by the rule-based system, and the canonical sentence is returned. This canonical sentence is appended to the extra statement(s) to obtain the final canonical form of the QA pair.

To process answers with well-formed sentences, we break the answer text into parts, where the first part is the actual answer, while the other part is the extra statement(s). The question and the actual answer are processed by the rule-based system and the canonical sentence is returned. This canonical sentence is appended by the extra statement(s) to obtain the final canonical form of the QA pair.

<sup>2</sup>For more information on dialog act (DA) classes please see Section 2.3 and [20].

## 5.2 Chunking and Chinking method

Chunk extraction, also called shallow parsing, refers to the process of extracting short phrases from a sentence, which has been tagged with Part-of-Speech (POS). The phrase or set of words/tokens combined are called chunks. Chunking was also one of the methods used for entity detection. Chinking refers to the process of defining patterns or words that can't be part of the chunk.

Chunking and chinking allow us to parse the tagged sentences and identify specific phrases for use such as noun phrases, verb phrases, etc. [14] The chunk structure can be represented in the form of trees or tags. These structures can be used to perform transformations and restructuring of the sentence. Chunks can be identified through regular expressions and POS tags.

This can be better understood with an example. Consider we have a QA pair as follows:

**Question:** Were you wearing shoes before the accident took place?

**Answer:** yes, sir

We parse the question and tag its part-of-speech using NLTK's default POS-tagger. This presented us with a tree of tuples <word, POS tag>, seen in Fig. 5.2.



Figure 5.2: Part-Of-Speech tagged question

One of the rules implemented is based on personal pronoun “<.\*>?<PRP><.\*>?”. There are 3 parts to this rule “<.\*>?”, “<PRP>”, and “<.\*>?”. The first part denotes 0 or 1 word coming before the PRP tagged word, the second part defines a PRP tagged word, and the third part defines no or 1 word coming after the PRP tagged word. In simpler terms, this

rule specifies to identify a 1, 2, or 3-word set having at least a personal pronoun in between. In this case, it extracted “were” and “wearing” which came before and after the personal pronoun “you”. Fig. 5.3 shows the chunk formed as part of the chunking process.

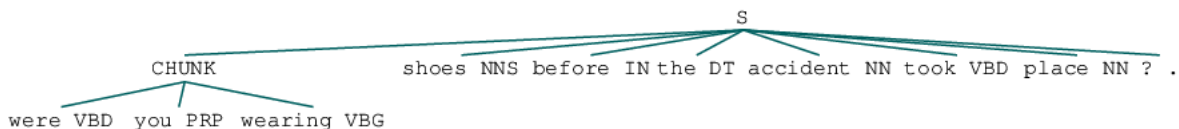


Figure 5.3: Identifying chunks in the question

For the declarative form, we need to transform the question and the identified chunk into first-person from the deponent’s perspective. For this example, we swapped the first word and personal pronoun and transformed “you” to “I” and “were” to “was”. The transformed sentence, in this case, becomes “*I was wearing shoes before the accident took place.*” as seen in Fig. 5.4.



Figure 5.4: Canonical form of QA pair

There are different transformations for different DA pairs and identified POS tag patterns. These transformations are stored in the form of a dictionary, which was expanded iteratively as we observed new highly occurring patterns. We aimed to develop a small number of rules with the ability to be implemented across various datasets.

Table 5.3 shows some of the rules we implemented. Each rule has its code independent of the others, and can be accessed through the meta-transformer as shown in Fig. 5.5.

Table 5.3: Explanation of some rules implemented for transformation.

Question	Answer	Rule Name	Rule Regex	Rule Explanation
And, ma'am, how long <b>have you been</b> licensed as a physical therapist, please?	Four years.	wh-simple chunker	<VB.??*<PR.??><VB.??>+	This is first rule the [wh, sno] QA pairs go through. In this rule, they identify a pattern where there are 0 or more verbs before a personal/possessive pronoun followed by at least one verb after the pronoun. If such a pattern is identified then exchange the initial verb (if present) with the pronoun and replace the pronoun with the correct form. Check tense of the verb that previously succeeded the pronoun.
And <b>can you explain</b> that process just very generally and what is your certification, if you would, please?	I take – we take boards, and then you pass or fail. And once you're passed, you become licensed to practice in LOCATION15. And so I'm licensed by LOCATION15 to provide physical therapy.	wh-partial chunker	<.*?><PRP><.*?>	This is the second rule the [wh, sno] QA pairs go through. In this rule, they identify a pattern where there are no or 1 word before a personal pronoun followed by no or 1 word after the personal pronoun. If such a pattern is identified then exchange the initial word (if present) with the personal pronoun and replace the personal pronoun with the correct form.
Each time you <b>thought about</b> it, you got to the precipice and you stopped?	Yes, sir.	binary-injunction	<VB.*><IN>?	This is the second rule implemented for [bin, y] and [bin,n] QA pairs. It identifies a possible 2-gram window where there is a verb followed by an injunction. We identify the closest personal pronoun preceding this chunk and replace it with its corrected form. We keep the rest of the chunk the same.
Ma'am, <b>are you a member</b> of any professional societies?	No.	vb_dt	<VB.*><.*?><DT>?<NN.*>	This is the third rule for [bin, y] and [bin,n] QA pairs. It aims to identify a pattern which starts with a verb, which may or may not be followed by a word, followed by a determiner, and ending with a noun. We identify the closest personal pronoun preceding this chunk and replace it with its corrected form. We keep the rest of the chunk the same.

### 5.3 Post-processing of Questions

Even after the pre-processing step, we observed instances where there was noise in the canonical sentences. This is possible either because of noise pre-existing in the answer itself or because of some noise introduced by the system. For better transformation quality it is important to handle such conditions. That is handled similarly to the pre-processing step. We identified a few sets of words and phrases which recur frequently and develop a regular expression based rule to handle them.

Another task we handled in post-processing was expanding contracted words. Contracted words are words or phrases which have been shortened by dropping one or more letters (such

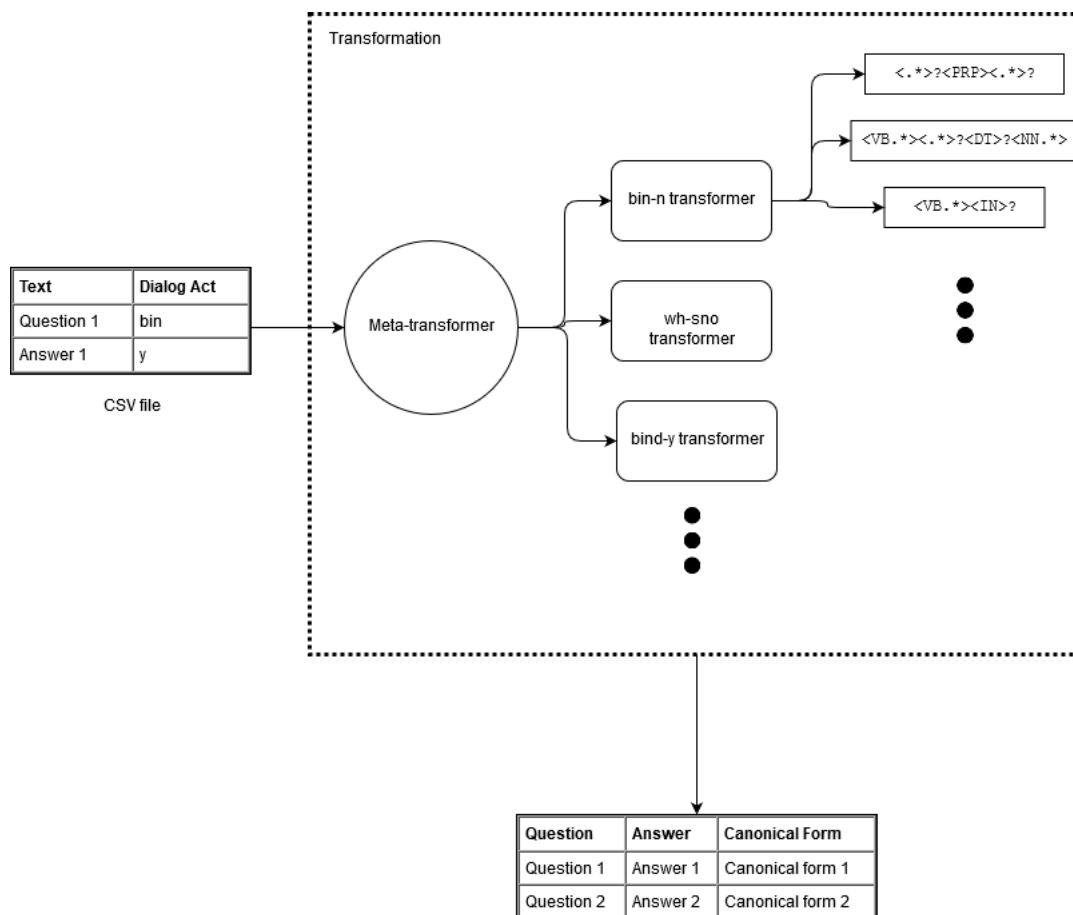


Figure 5.5: Shows how the code is structured internally [21]

as don't, can't, won't). These are generally used in colloquial language. Shortening text is useful for dimensionality reduction, but in this condition, for evaluation, we need expanded forms. We used the pycontraction [10] library to expand the contractions. It performs three passes on the contraction. The first is where simple rules are used to replace the contraction. In the second pass, contractions with possible multiple combinations are replaced with all combinations of rules. All these possibilities are then run through a grammar checker and Word Mover's Distance (WMD) is calculated between each result and the original text. The expansion with the least number of grammatical errors and shortest distance is returned [10].

Table 5.4: Examples of QA pairs having “- -”.

Question	Answer
Okay	we’re required to get i think it’s <b>20 hours of – 20 contact hours</b> for continuing education.
was that <b>your first – that was</b> his first intake here at in motion?	yes.
all that is documented on what <b>has –</b>	yes, sir.

We also observed that there are many instances where “- -” occurs. This typically occurs when there is some interruption in-between the conversation or if the person speaking reforms part of the question/answer. Table 5.4 shows some examples where “- -” is found. We sought to solve the “- -” problem through a 4 step approach.

1. Checking if there are more than 2 “- -” within close vicinity (4-5 spaces). If so, the content between the “- -” occurrences was removed.
2. Checking if the words (1, 2, or 3) before and after “- -” were the same or not. If they were, then one set was removed.
3. Checking similarity of words in close vicinity to “- -” (1, 2, or 3 words before and after). If similar, one set of words was removed.
4. If none of the above conditions was met, “- -” was replaced with a “;”.

Also within this post-processing step after handling the “- -” condition, we processed all the canonical sentences through spell and grammar checkers (GingerIt [47] and LanguageTool [59]). We used both of these libraries to check for issues in the sentences and an combination of the two scores was used to decide if sentence correction was needed or not.

## 5.4 Sentence Correction

The English language is quite complex. Each sentence can be presented in many different ways. Even in a legal deposition, there are different linguistic styles of attorneys as well as deponents, making it quite hard to implement rules for each of them. We therefore aimed at developing a small number of generalized rules, which may cause some lexical errors, but gives some sense of the crux of the QA pair. These erroneous sentences can then be corrected with the help of a sentence correction system. We explore different ways to correct sentences. Sentence correction can also be formulated as a grammatical error correction (GEC) task, in which the aim is to correct all types of errors (grammatical, lexical, and orthographic). This task has been extensively researched in the recent past. Two standard datasets have emerged to test the developed systems, the BEA shared-task dataset [17] and the CoNLL-2014 dataset [61]. KenLM [40] and PIE-model [5] are two GEC systems we explore as baselines to understand the efficacy of pre-trained systems.

### 5.4.1 Using Language Models and Heuristics

The canonical sentences developed from QA pairs in legal depositions may have a slightly different structure from the erroneous dataset used for training the KenLM and PIE-models. This may cause the systems to be unable to identify some of the errors. As we know, BERT and AI-GPT2 are the current state-of-the-art language models that can be fine-tuned for various tasks. They also have an internal functionality of scoring. We utilize a variation of this scoring mechanism to obtain the score of each word in a sentence, summing them to obtain the score for the whole sentence. Higher scores from the internal function indicates that out of the trained vocabulary words, a specific word has a higher chance of completing the sentence.

```

4 [{"I", 3.5976398838623847}, {"know", 0.5747198117835999}, {"obviously", 0.6869436502456665}, {"his", 2.864443382154541}, {"age", 2.6465089321136475}, {"", 6.689393997192383}, {"", -2.361978054046631}]
5 I know obviously his age . 14.697669923385511
6
7 [{"obviously", -6.66588029324707}, {"I", 3.837256956180464}, {"know", 6.492459297180176}, {"his", 4.318914413452148}, {"age", 2.9551889896392822}, {"", 7.845702648162842}, {"", -2.3381288051605225}]
8 Obviously I know his age . 15.64551329612732
9
10 [{"obviously", -6.66588029324707}, {"know", -1.0496476808656616}, {"I", 1.886763572692871}, {"his", 3.122837543487549}, {"age", 1.9495199918746948}, {"", 6.698884963989258}, {"", -2.6981201171875}]
11 Obviously know I his age . 6.13953740234375
12
13 [{"know", -5.686948299407959}, {"I", 1.819367200546265}, {"obviously", -2.9052672386169434}, {"his", 2.6298122406608586}, {"age", 3.8092949390411377}, {"", 6.099327564239502}, {"", -2.508786312530518}]
14 Know I obviously his age . 3.256881594657898
15
16 [{"know", -5.686948299407959}, {"obviously", -1.4097535610198975}, {"I", 3.586249351501465}, {"his", 3.031085729598999}, {"age", 2.72596732711792}, {"", 6.349431901577148}, {"", -2.456474542617798}]
17 Know obviously I his age . 6.13953740234375
18
19 [{"I", 3.5976398838623847}, {"obviously", -1.7805718183517456}, {"know", -0.41284348620040894}, {"his", 4.28870964050293}, {"age", 3.7877724170684814}, {"", 7.313677787780762}, {"", -2.01599168774658}]
20 I obviously know his age . 14.778332016887665
21
22 [{"I", 3.5976398838623847}, {"obviously", -1.7805718183517456}, {"his", 4.732639789581299}, {"know", 0.5547664761543274}, {"age", -0.426776647567749}, {"", 7.111874103546143}, {"", -2.510385404663086}]
23 I obviously know his age . 11.279203502561493
24
25 [{"I", 3.5976398838623847}, {"know", 0.5747198117835999}, {"obviously", 0.6869436502456665}, {"his", 2.864443382154541}, {"age", 2.6465089321136475}, {"", 6.689393997192383}, {"", -2.361978054046631}]
26 I know obviously his age . 14.697669923385511
27
28 [{"I", 3.5976398838623847}, {"know", 0.5747198117835999}, {"his", 5.2217270480529785}, {"obviously", -1.516045411642456}, {"age", 1.7427239418029785}, {"", 7.221044663568115}, {"", -1.4285701513290405}]
29 I know his obviously age . 15.41323034657669
30
31 [{"I", 3.5976398838623847}, {"his", 3.4789958000183105}, {"obviously", -2.6597402095794678}, {"know", 0.7925689816474915}, {"age", -1.394138825996399}, {"", 5.941583156585693}, {"", -2.348158836364746}]
32 I his obviously know age . 7.408757150175187

```

Figure 5.6: Example of scoring of words and sentence using BERT

Fig. 5.6 shows an example of the sentence “*I obviously know his age.*” going through the BERT scoring system with a 3-gram swap. In the figure, each word has a separate score and these scores are then summed up together to obtain the score of the sentence.

As we mentioned before, the structure of the canonical sentence may differ slightly from standard datasets. We observed some common patterns which could be tackled by applying some simple heuristic techniques. Also, with a combination of the heuristics and scoring mechanism we could improve the transformations. The following are the heuristics we explored:

- **N-gram based swap:** In the starting few words of the sentence, we observed that some words are misplaced. If they were swapped with their preceding or succeeding word the sentence would make more sense. Therefore, we introduced an n-gram swap system for the first 10 to 12 words. Within the n-gram window, we performed all possible permutations of word placements and compared their score with the score of the original sentence. Finally, the sentence with the best score was returned.

Table 5.5: Example of an N-gram based swap.

Input sentence	do <b>not</b> <b>do</b> <b>me</b> recall going to his house that night.
Corrected sentence	do me do not recall going to his house that night.

- **One-word deletion:** Some sentences were observed to have extra words, which if

deleted would make the sentence more logical. We calculated a length normalized score for each sentence. If an updated normalized score was better, that identified the corrected sentence.

Table 5.6: Example of one word deletion.

Input sentence	<b>do</b> me do not recall going to his house that night.
Corrected sentence	me do not recall going to his house that night.

- **One word replacement:** Similar to some steps mentioned in the previous sections, we observed that some common words caused lexical errors in sentences. We created a dictionary of such words and what they should be replaced with. The alternative word is substituted and the score is calculated. If the score was better than the original score, we considered this as the new corrected sentence.

Table 5.7: Example of one word replacement.

Input sentence	<b>me</b> do not recall going to his house that night.
Corrected sentence	I do not recall going to his house that night.

## 5.5 Transformation using Deep Learning

Deep learning is a sub-field of machine learning methods, which uses a deeper architecture (multiple layers within a network) to learn, in applications of NLP, speech recognition, computer vision, etc. Deep learning has been used to solve text-based problems such as text classification, language modeling, speech recognition, machine translation, summarization, and question answering. Different architectures have also been introduced for the same.

We hypothesized that the transformation of QA pairs can be considered as a machine translation problem where the QA pair may be the source and the ground truth declarative sentence is the target, even though both the source and target language are the same (English). We

wanted the model to not only learn the transformation but learn the ability to copy as well, somewhat similar to COPYNET [38]. Due to the limited number of annotated sentences we have, this becomes challenging as deep learning models generally are data-hungry. We ran this transformation experiment on a smaller dataset of 4 DA combination classes [bin, y], [bin, n], [bin, y-d], and [bin, n-d], using the OpenNMT [46] toolkit.

```

1  have you ever been deposed? have you ever been in a scenario
   like this where you've been asked questions to give testimony
   about a case, a proceeding?|yes, i have.
2  have you had an opportunity to do that?|yes, i did.
3  as we're getting started, do you have any questions about what's
   getting ready to take place?|no.
4  is there any reason mentally, physically, or otherwise that
   you're not comfortable proceeding today?|no.
5  is there any reason right now mentally, physically, emotionally,
   otherwise, illness or anything else, that causes you concern
   about going forward with today's deposition?|no.
6  are there board certifications or other certifications --|yes.
7  you're not on any kind of medication, good night's sleep, don't
   feel ill, or anything like that?|no.
8  thank you, ma'am. do you understand that you've just taken an
   oath? the court reporter has sworn you in, and that's just the
   same as if you were in court. the same oath applies. do you
   understand that?|yes.
9  will you agree that you'll take that oath just as seriously
   today as if you were sitting in a courtroom right now?|yes.
10 yes, sir. doctor, are you board certified?|that's correct. i am.
11 are you board certified?|yes.

```

(a)

```

1  I have been in a scenario like this where I've been asked
   questions to give testimony about a case a proceeding.
2  I have had an opportunity to do that.
3  I do not have any questions about what's getting ready to take
   place.
4  there is no reason mentally physically that I do not
   comfortable proceeding today.
5  there is not any reason right now mentally physically
   emotionally otherwise illness or anything else that causes
   me concern about going forward with today 's deposition.
6  there are board certifications or other certifications --
7  I am not on any kind of medication. I do not feel ill or
   anything like that.
8  I do understand that I 've just taken an oath. the court
   reporter has me sworn in and that 's just the same as if I were
   in court . the same oath applies . I do understand that.
9  I will agree that I 'll take that oath just as seriously today
   as if I were sitting in a courtroom right now.
10 yes I am board certified.
11 I am board certified.

```

(b)

Figure 5.7: Showing small samples of the source (a) and target (b) files for machine translation

# Chapter 6

## Experiments and Results

In this chapter, we discuss the results obtained for the experiments performed. Initially, we show the improvements observed in parsing, followed by an analysis of the rule-based transformation. We also provide an analysis of the transformation with deep learning. This chapter allows us to analyze **RQ5**: *Can we evaluate the transformed sentences on quantitative metrics such as ROUGE and similarity score?* Other preliminary explorations and studies can be found in Appendix C.

### 6.1 Parsing

Work in [20] was able to handle most of the deposition transcripts available in the proprietary dataset (seen in Fig. 6.1). A small percentage of them were partially parsed and could be improved upon. A common theme we found for the unparsed documents was that they were born digital. Section 3.1 gives more details on how we parsed such documents.

The tobacco deposition transcripts were already OCRed by some system but the quality leaves much to be desired. When we use the default Apache-Tika to parse it, we obtain a JSON similar to Fig. 6.2. It can be observed that the red highlights indicate some errors in parsing. There are different QA pairs identified within it, causing us to lose information and affect the transformation of QA pairs. On the other hand, Fig. 6.3 shows the results from the new parsing technique. It holds all of the information and has done a good job of

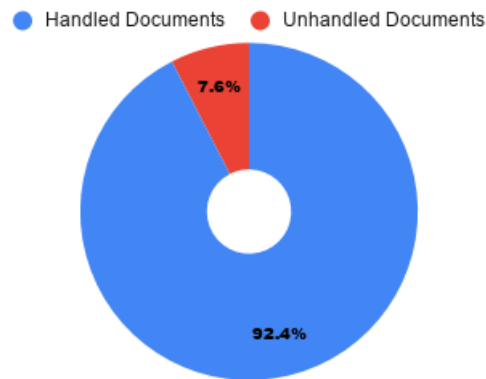


Figure 6.1: Visualization of percentage of handled documents

```
{
  "question": "Explain the total retention column entries if you would, please.",
  "answer": "In traditional records management, when you structure a retention schedule, sometimes you have a breakdown, and people break down how long you keep it in the office before you transfer it to off site storage; and if something has a short retention time, it might only have in office how long you keep it in the office file before you make a retention decision. Or sometimes if it has a lengthy retention time, that's something that would have a breakdown between how long you keep it in the office and when you should transfer it to off site storage for the remainder of the retention time. And total retention time, if you look at a retention schedule from the eighties and nineties, that you are going to find a breakdown 4 that before that domestically was department retention schedule with a retention code, a Esquire Deposition Services category Lynch - Confidential , in office retention, off site retention, and total retention which is the sum of the two."
},
{
  "question": "So looking here we see the letter C. That's current year? years? That's correct. And plus one, plus ten are those That's correct. What is SUP? Superseded. Which means what? If a new version came out that the prior version was now up for possible review for some -- long as codes we column. this retention decision. ACT is active. So you keep those as you -- As long as it is active, yes. Let's see if there are any other need to be concerned with in this P? That's permanent. And if there are any disposal Esquire Deposition Services Lynch - Confidential suspension notices those would supersede whatever the retention period is?",
  "answer": "Disposal suspension for our program is layered on top of the retention requirements, and it frees retention until release by the legal department."
}
```

Figure 6.2: Parsing with old technique

separating out QA pairs for us to transform.

To identify the differences between the old and new parsing techniques, we passed a single tobacco deposition through both processes. Table 6.1a and Table 6.1b show the distribution of the QA pairs across the two different parsing techniques. For the old parsing technique, we observed that some of the QA pairs have been mis-classified as [bin-d, sno]. This is primarily because the system is unable to identify delimiters such as “Q.” or “A.” in the transcript due to poor quality. This causes multiple QAs to collect as a single QA pair. Comparing the two statistics we observed that Amazon Textract can read the document more clearly, allowing us to correctly identify, and classify, QA pairs. There is an increase in the number of QA

```

{
  "question": "Explain the total retention column entries if you would, please.",
  "answer": "In traditional records management, when you structure a retention schedule, sometimes you have a breakdown, and people break down how long you keep it in the office before you transfer it to off site storage; and if something has a short retention time, it might only have in office how long you keep it in the office file before you make a retention decision. Or sometimes if it has a lengthy retention time, that's something that would have a breakdown between how long you keep it in the office and when you should transfer it to off site storage for the remainder of the retention time. And total retention time, if you look at a retention schedule from the eighties and nineties, that you are going to find a breakdown that before that domestically was department retention schedule with a retention code, a Sourcehttpsillwwwindustrydocumentsucsteduldocuslirypole category, in office retention, off site retention, and total retention which is the sum of the two."
},
{
  "question": "So looking here we see the letter C. That's current year?",
  "answer": "That's correct."
},
{
  "question": "And plus one, plus ten are those years?",
  "answer": "That's correct."
},
{
  "question": "What is SUP?",
  "answer": "Superseded."
},
{
  "question": "Which means what?",
  "answer": "If a new version came out that the prior version was now up for possible review for some -- this retention decision."
},
{
  "question": "ACT is active. So you keep those as long as you --",
  "answer": "As long as it is active, yes."
},
{
  "question": "Let's see if there are any other codes we need to be concerned with in this column. P?",
  "answer": "That's permanent."
},
{
  "question": "And if there are any disposal Source httpsillwwwindustrydocuments.ucsteduldocuslirypole suspension notices those would supersede whatever the retention period is?",
  "answer": "Disposal suspension for our program is layered on top of the retention requirements, and it frees retention until release by the legal department."
}
}

```

Figure 6.3: JSON output from parsing with new technique

pairs identified (64 in this document), allowing us to extract more information from the document. We observed that there is an increase in the number of QA pairs for classes such as [wh, sno], [bin, y], [bin, n], which also have an increase in the percentage of total QA pairs. We observed a decrease in the number of [bin-d, sno] QA pairs for the same reasons mentioned in the previous paragraphs.

Additionally, we parsed all the 8 tobacco depositions used to create the old dataset. Amazon Textract did sometimes fail in identifying the difference between “Q” and “9” due to the quality of the deposition scans. We found over 200 instances across the dataset where this occurred. Table 6.2 compares the distributions of the old and new parsing techniques. Similar to the observations on the single deposition, there was an increase in the number of QA pairs identified. Overall, [wh, sno], [bin-d, y], and [bin, y] are the classes which find a boost in the numbers.

Table 6.1: QA distribution with different parsing techniques.

(a) QA distribution while parsing with [20].

(b) QA distribution while parsing with Textract.

Q-DA	A-DA	# of QA pairs	% of Total
wh	sno	62	11.54
bin	sno	56	10.42
bin	n	48	8.93
bin	y	46	8.56
bin-d	sno	44	8.19
bin	y-d	27	5.02
bin-d	y	26	4.84
bin	dno	24	4.46
sno	sno	17	3.16
bin	n-d	15	2.79
wh-d	sno	14	2.60
wh	dno	11	2.04
.	.	.	.
<b>Total</b>		<b>537</b>	<b>100</b>

Q-DA	A-DA	# of QA pairs	% of Total
wh	sno	83	13.81
bin	y	59	9.81
bin	sno	57	9.48
bin	n	56	9.31
bin-d	sno	39	6.48
bin	y-d	32	5.32
bin-d	y	30	4.99
bin	dno	26	4.32
wh-d	sno	20	3.32
sno	sno	17	2.82
bin	n-d	15	2.49
bin-d	y-d	12	1.99
.	.	.	.
<b>Total</b>		<b>601</b>	<b>100</b>

Table 6.2: Top 13 DA distribution on new parsing technique.

Qstn-DA	Ans-DA	New Parsing Technique		Old Parsing Technique	
		# of Samples	% of Total	# of Samples	% of Total
bin-d	sno	419	12.12	454	13.58
bin	sno	407	11.78	441	13.19
wh	sno	336	9.72	297	8.88
bin-d	y	275	7.96	235	7.02
bin	y	215	6.22	183	5.47
bin	n	167	4.83	143	4.27
sno	sno	148	4.28	143	4.27
bin	y-d	116	3.36	118	3.52
bin-d	y-d	113	3.27	92	2.75
wh-d	sno	97	2.80	87	2.6
bin	dno	88	2.54	95	2.84
bin-d	dno	71	2.05	79	2.36
.	.	.	.	.	.
<b>Total</b>		<b>3456</b>	<b>100</b>	<b>3343</b>	<b>100</b>

## 6.2 Proprietary Dataset

There are two important issues regarding handling transformations. One is the quality of transformation, while the other is the coverage of transformation. Since we developed most of the rules across the M3 dataset, we find the coverage across the same.

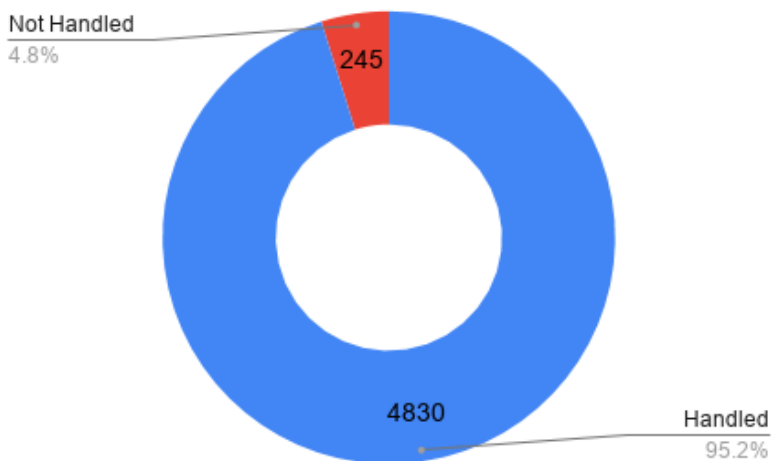


Figure 6.4: Coverage of the rule-based transformer

The M3 dataset consists of approximately 5100 QA pairs. Fig. 6.4 shows a doughnut chart across this dataset and the number of QA pairs, showing the number that we have handled, either fully or partially. A majority of sentences are being handled by the developed rules. We observed that very short questions (2 or 3 word questions) and some specific DA class pairs such as “confront” or “co” are some unhandled cases.

Now, the second important aspect is the quality of transformation. Here we compare simplistic transformation techniques with the one we have implemented, to identify differences and improvements in transformations. In our experiments we compare three different methods:

- **Use answer:** The answer is pre-processed for removing noise and then is used as is. This allows us to evaluate how much the answer in isolation, gives context, or how

important the answer is itself.

- **Use question and answer:** The question and answer are pre-processed to remove noise in them. The answer is concatenated to the question and the new string is used as the transformed output.
- **Use output of chunking transformers:** We use the rules and chunkers defined in Chapter 5 for transforming the QA pairs.

The evaluation of these sentences can be done qualitatively and quantitatively. For a qualitative evaluation we go through a set of sentences from each DA class pair and identify any errors in the transformation. For a more quantitative evaluation we use ROUGE scores and Semantic Similarity. ROUGE (1 and 2) scores allow us to track a word-based similarity between the ground truth and transformed canonical form. For a more semantically correct evaluation, we use Infsent Sentence Embeddings [26]. Using a combination of the two we can identify the performance of the transformations.

Table 6.3: Evaluation results for M10 dataset.

Qstn DA	Ans DA	Just Answer			Q+A			Chunking		
		R-1	R-2	Sim	R-1	R-2	Sim	R-1	R-2	Sim
wh	sno	0.73	<b>0.67</b>	0.79	<b>0.77</b>	0.66	<b>0.87</b>	0.74	0.62	0.85
bin	y	0.09	0.03	0.29	0.75	0.56	0.84	<b>0.85</b>	<b>0.70</b>	<b>0.90</b>
bin-d	y	0.016	0.002	0.11	0.81	0.70	0.90	<b>0.89</b>	<b>0.78</b>	<b>0.93</b>
bin	sno	0.67	0.63	0.76	<b>0.83</b>	<b>0.75</b>	<b>0.90</b>	0.81	0.71	0.89
bin	n	0.08	0.04	0.36	0.72	0.54	0.81	<b>0.83</b>	<b>0.71</b>	<b>0.91</b>
sno	sno	0.67	0.62	0.73	<b>0.90</b>	<b>0.85</b>	<b>0.95</b>	0.79	0.71	0.87
ack	sno	<b>0.98</b>	<b>0.98</b>	<b>0.99</b>	0.94	0.93	0.94	0.92	0.85	0.97
wh-d	sno	<b>0.82</b>	<b>0.78</b>	<b>0.87</b>	0.64	0.57	0.78	0.64	0.55	0.81
bin	y-d	0.55	0.47	0.68	<b>0.78</b>	<b>0.65</b>	<b>0.87</b>	0.73	0.60	0.82
bin	n-d	0.45	0.31	0.74	0.59	0.43	0.8	<b>0.65</b>	<b>0.50</b>	<b>0.84</b>

## Analysis

Table 6.3 shows the comparison of the 3 techniques across evaluation metrics such as ROUGE 1 (R-1), ROUGE 2 (R-2), and Inferred Sim (Sim). Let us evaluate the top few DA classes for each of the methods:

- Just Answer
  - *wh/sno*: For these types of QA pairs we observed that many of the answers had enough context so that the question wasn't needed. This is one of the reasons why it has a decent ROUGE as well as Semantic Similarity scores. In cases where the answer was short, not enough context was obtained, causing a dip in performance.
  - *bin/y*: As we expected, these QA pairs did not perform well with the “just answer” transformation. This is because the answer only consisted of a few words such as “yes”, “uh-huh”, or “yeah” which gave zero context on what the deponent was agreeing to.
  - *bin/n*: Similar to “bin|y” QA pairs, where the answer only consisted of words such as “No” or “Nah”, it gave no context on what the deponent was disagreeing with.
  - *bin/sno*: Similar to “wh|sno” we observed that the answers were quite descriptive. There were high chances that the answer had the crux of the information, resulting in a decent similarity to the ground truth.
- Q+A
  - *wh/sno*: Many of the answers in such QA pairs were descriptive enough to give the crux. For the shorter answers, having the question appended gave at least some context which caused a small improvement in performance. For the transformations where, initially, the answer was just enough, appending the question

introduced some repetition which caused a drop of ROUGE scores in some cases.

- *bin-d/y*: This type of transformation performed quite well as the context (present in the question) was also introduced with the answers. For many “yes” answered questions, even if we replace the last punctuation of “?” with a dot, the sentence may make sense. This is why there is a significant improvement.
- *sno/sno*: In this case, both the question and the answer are longer. Combining both of them allowed the whole frame of reference to be captured and therefore giving the best performance of the 3 models (for these QA pairs).
- *bin/y-d*: The question and answer together were able to capture a major part of the subject theme due to which it gave the best performance out of the three methods implemented.

- Chunking

- *wh/sno*: The transformation performance was quite similar to the “Q+A” technique. We observed that in many cases there were perfect overlaps with the ground truth, but there were also cases where the canonical sentence was a good paraphrase of the ground truth. This paraphrasing caused a dip in ROUGE scores but the semantic similarity stayed in the same range.
- *bin/y*: This technique yields the best performance for such QA pairs out of the 3 approaches compared. This is because it portrays the crux of the QA pair while removing redundant words or phrases. This causes an improvement in the ROUGE scores as well as similarity scores.
- *bin-d/y*: Similar to the “bin|y” DA class, this technique has the best performance for such DA pairs. This is mostly because there is a good overlap between the ground truth and the canonical sentences of such form. There is minimal redun-

dancy and the system can identify the semantic sense of the deponent.

- *bin/n-d*: This technique performs the best for such QA pairs. The implemented technique is able to identify the negative tone of the deponent and can factor that into the developed sentence. This results in an increase in ROUGE scores.

## 6.3 Tobacco Dataset

Rule-based transformations are primarily dependent on the rules developed and the dataset used to develop the rules. To ensure that the rules are generalizable enough, we use the tobacco dataset to check the performance of the transformations. Similar to the analysis of the M10 dataset, we will use three techniques and three evaluation metrics for evaluating the performance of transformations. Table 6.4 shows the comparison of the 3 techniques across evaluation metrics such as ROUGE and Semantic Similarity.

Table 6.4: Evaluation results of Tobacco Dataset.

Qstn DA	Ans DA	Just Answer			Q+A			Chunking		
		R-1	R-2	Sim	R-1	R-2	Sim	R-1	R-2	Sim
bin	sno	0.74	0.72	0.83	<b>0.84</b>	<b>0.80</b>	<b>0.91</b>	0.81	0.75	<b>0.91</b>
wh	sno	<b>0.82</b>	<b>0.78</b>	0.87	0.76	0.69	0.87	0.78	0.70	<b>0.89</b>
bin-d	y	0.05	0.01	0.02	0.85	<b>0.78</b>	<b>0.92</b>	<b>0.86</b>	<b>0.78</b>	<b>0.92</b>
bin	y	0.09	0.04	0.21	0.81	0.67	0.87	<b>0.92</b>	<b>0.81</b>	<b>0.94</b>
bin	n	0.11	0.05	0.36	0.79	0.64	0.85	<b>0.90</b>	<b>0.78</b>	<b>0.94</b>
bin	yd	0.60	0.54	0.72	<b>0.76</b>	<b>0.68</b>	<b>0.86</b>	0.74	0.66	0.84
bin	nd	0.63	0.56	0.78	<b>0.79</b>	<b>0.70</b>	<b>0.89</b>	0.75	0.67	0.87
bin	dno	<b>0.78</b>	<b>0.75</b>	0.86	0.74	0.70	0.85	0.74	0.63	<b>0.87</b>

### Analysis

Let us evaluate the top few DA classes for each of the methods:

- Just Answer
  - *wh/sno*: Many QA pairs coming in this class have long answers. Such answers, in many cases, have enough context and information independent of the question. We stumble with some issues with shorter answers as enough information isn't present. This is why we have high ROUGE scores.
  - *bin/y*: Similar to the M10 dataset, this class did not perform well with the “just answer” transformation. This is because the answer only consisted of a few words such as “yes,” “uh-huh,” or “yeah,” which gave no context on what the deponent was agreeing to.
  - *bin/n*: Similar to “bin|y” QA pairs, the answer only consisted of words such as “No,” or “Nah”, giving no context on what the deponent was disagreeing with.
  - *bin/dno*: These QA pairs have answers such as “I don't know,” or “I'm not sure”. These statements are then followed by some explanation, making these answers independent enough to give the crux of the QA pair. Due to this, there is a large overlap between the ground-truth and the answer itself.
- Q+A
  - *wh/sno*: For questions having longer answers, appending the question makes some content redundant, causing a dip in ROUGE scores.
  - *bin/n*: The ground truth of these QA pairs is some form of paraphrasing of the question. Due to this, appending the question with the answer will cause a large overlap of words, even if it may not make grammatical sense.
  - *bin/sno*: This technique gives the best performance for such QA pairs. The answers and questions are of medium length; combining both of them completes the

topic of discussion. There also is a good overlap since a large part of the question and/or answer would be present in the ground truth.

- *bin/y-d*: This technique of transformation gave the best result for this set of QA pairs. The combination of questions and answers offered us enough information on the topic going on. There is a reasonable overlap between the ground truth and the transformed QA pair.

- Chunking

- *wh/sno*: The answers are quite lengthy in such cases. We have a good handle on the shorter sentences, but introduce some noise on the longer QA pairs, causing a small drop in ROUGE scores. We are, however, able to improve the semantic similarity of the canonical sentences. In a few cases, we observe that an improvement in parsing would automatically improve the sentence transformation.
- *bin/n*: This technique performs the best for such QA pairs. It can identify the negative intent of the deponent to the question. We observed some grammatical errors and paraphrasing in these sets of sentences, such as “not ever” for “never”.
- *bin-d/y*: The rules developed for this set of QA pairs gave a similar performance to the “Q+A” technique. We observed that a majority of the sentences have a decent ROUGE score but there are some parts of the QA pair which become redundant.
- *bin/dno*: In this set of QA pairs, we observed that there are many cases where the answer is good enough and some cases where we have to massively modify the question for a grammatically correct sentence. Handling such complicated transformations is slightly difficult, causing a small reduction in ROUGE scores.

## 6.4 Sentence Correction

We also explored the possibility of how sentence correction might help in QA transformation. We explored the use of previously developed sentence correction models and state-of-the-art language models. There are 8 different setups we experiment with, mentioned in Table 6.5. This experimentation is performed on the M10 dataset.

Table 6.5: Experimental setup for sentence correction.

Name	Experiment
Model 1	KenLM
Model 2	PIE
Model 3	AIGPT2 + 3-gram swap
Model 4	BERT + 2-gram swap
Model 5	BERT + 3-gram swap
Model 6	BERT + 3-gram swap + single deletion and replacement
Model 7	PIE + BERT + 3-gram swap
Model 8	pre-processing + PIE + BERT + 3-gram swap

### Analysis

Table 6.6, Table 6.7, and Table 6.8 show the ROUGE 1, ROUGE 2, and Semantic Similarity scores, respectively, across the different models that we have implemented. The following are some key takeaways:

Table 6.6: ROUGE -1 scores for all models

Q-DA	A-DA	No Correction	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8
wh	sno	0.74	0.74	0.74	0.73	<b>0.77</b>	<b>0.77</b>	<b>0.77</b>	<b>0.77</b>	<b>0.77</b>
bind	y	<b>0.89</b>	<b>0.89</b>	<b>0.89</b>	0.85	<b>0.89</b>	<b>0.89</b>	0.88	<b>0.89</b>	<b>0.89</b>
bin	y	<b>0.85</b>	<b>0.85</b>	<b>0.85</b>	0.8	<b>0.85</b>	<b>0.85</b>	0.84	<b>0.85</b>	<b>0.85</b>
bin	n	0.83	0.82	0.82	0.79	0.83	0.83	0.83	0.82	<b>0.84</b>
bin	sno	<b>0.81</b>	<b>0.81</b>	0.8	0.77	<b>0.81</b>	<b>0.81</b>	0.8	0.8	0.8
sno	sno	<b>0.79</b>	0.78	0.77	0.75	<b>0.79</b>	<b>0.79</b>	0.77	0.77	0.77
whd	sno	0.64	0.65	0.64	0.63	0.65	0.65	<b>0.66</b>	0.65	0.65
bin	yd	0.73	0.73	0.73	0.7	0.73	0.73	0.72	0.73	<b>0.74</b>
bin	nd	<b>0.65</b>	0.53	<b>0.65</b>	0.62	<b>0.65</b>	<b>0.65</b>	<b>0.65</b>	<b>0.65</b>	<b>0.65</b>
ack	sno	<b>0.92</b>	-	0.9	0.88	<b>0.92</b>	<b>0.92</b>	0.91	0.9	0.89
Averaged		0.77	0.75	0.78	0.75	<b>0.79</b>	<b>0.79</b>	0.78	0.78	0.78

- ROUGE 1
  - We observed that there are minute gains in ROUGE 1 scores.
  - The KenLM and PIE-models are able to identify some of the errors in the dataset. For the errors it is unable to identify, the systems kept it as it was, hence there is no drop in ROUGE-1 scores.
  - We observed that the sentence scoring system of BERT is consistently better than that of AI-GPT2.
  - There is minimal difference in performing a 2-gram or 3-gram swap for the initial words of the sentence. They gave similar performance and offered the best performance amongst all the models for ROUGE-1 scores.
  - We only observed a 1 point improvement, i.e., the ROUGE scores improved by 0.01, in [bin,n] and [bin,yd] classes when we combined the PIE-models along with our BERT scoring mechanism and heuristics.

Table 6.7: ROUGE -2 scores for all models

Q-DA	A-DA	No Correction	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8
wh	sno	0.62	0.62	0.62	0.54	0.63	0.62	0.61	<b>0.65</b>	<b>0.65</b>
bind	y	0.8	0.78	<b>0.81</b>	0.62	0.76	0.74	0.7	<b>0.81</b>	<b>0.81</b>
bin	y	0.7	0.7	0.71	0.54	0.69	0.67	0.66	0.71	<b>0.72</b>
bin	n	0.71	0.7	0.7	0.58	0.69	0.7	0.68	0.7	<b>0.72</b>
bin	sno	<b>0.71</b>	<b>0.71</b>	0.7	0.59	0.7	0.69	0.67	0.7	0.7
sno	sno	<b>0.71</b>	0.7	0.69	0.57	0.67	0.66	0.63	0.69	0.69
whd	sno	0.55	0.55	0.53	0.52	0.55	0.55	0.55	<b>0.56</b>	<b>0.56</b>
bin	yd	0.6	<b>0.62</b>	0.61	0.5	0.59	0.59	0.56	<b>0.62</b>	<b>0.62</b>
bin	nd	0.5	0.37	<b>0.52</b>	0.42	0.5	0.5	0.49	<b>0.52</b>	0.51
ack	sno	<b>0.85</b>	-	0.8	0.67	0.8	0.77	0.78	0.8	0.8
Averaged		0.65	0.64	0.67	0.55	0.66	0.65	0.63	0.67	<b>0.68</b>

- ROUGE 2
  - We observed that the experimental setups with PIE-models deliver better performance for this metric. The internal spell checking mechanism allows it to perform better than the other models.

- The AI-GPT2 scoring mechanism is consistently outperformed by the BERT scoring mechanisms.
- A hybrid of the PIE-models, BERT scoring mechanism, and heuristics allows for the best performance. The pre-processing allows the removal of any extra noise within the canonical sentence, allowing the sentence correction system to better understand the sentences and erroneous parts within them.

Table 6.8: Semantic Similarity scores for all models

Q-DA	A-DA	No Correction	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8
wh	sno	0.85	0.85	0.85	0.85	<b>0.87</b>	<b>0.87</b>	<b>0.87</b>	<b>0.87</b>	<b>0.87</b>
bind	y	0.93	<b>0.93</b>	0.92	0.91	0.92	<b>0.93</b>	0.91	<b>0.93</b>	<b>0.93</b>
bin	y	0.9	<b>0.9</b>	<b>0.9</b>	0.86	<b>0.9</b>	<b>0.9</b>	0.89	<b>0.9</b>	<b>0.9</b>
bin	n	0.91	<b>0.92</b>	0.9	0.89	0.91	<b>0.92</b>	0.91	0.9	0.91
bin	sno	<b>0.89</b>	<b>0.89</b>	0.88	0.87	<b>0.89</b>	<b>0.89</b>	0.88	0.88	0.88
sno	sno	<b>0.87</b>	<b>0.87</b>	<b>0.87</b>	0.85	<b>0.87</b>	<b>0.87</b>	0.85	<b>0.87</b>	<b>0.87</b>
whd	sno	<b>0.81</b>	0.8	0.8	0.8	0.8	0.8	<b>0.81</b>	0.8	0.8
bin	yd	0.82	0.82	<b>0.83</b>	0.81	<b>0.83</b>	<b>0.83</b>	0.82	<b>0.83</b>	<b>0.83</b>
bin	nd	<b>0.84</b>	0.62	<b>0.84</b>	0.83	<b>0.84</b>	<b>0.84</b>	<b>0.84</b>	<b>0.84</b>	<b>0.84</b>
ack	sno	<b>0.97</b>	0.37	0.96	0.94	<b>0.97</b>	0.96	0.96	0.96	0.96
Averaged		0.87	0.84	0.87	0.86	<b>0.88</b>	0.87	<b>0.88</b>	<b>0.88</b>	<b>0.88</b>

- Semantic Similarity
  - As expected, even after sentence correction, the sentences do not lose their similarity to the ground truth sentences.

## 6.5 Transformation using Deep learning

We explored the possibility of using deep learning as a method for transforming QA pairs. We consider this as an NMT problem, where the source and the target language is English itself. We utilize the Open-NMT Toolkit for training models for this task. We limit the dataset to 4 specific classes – i.e., [bin,y], [bin,n], [bin, y-d], and [bin, n-d] – which have the most probability to succeed. We take two different approaches to train deep learning models

for transformation.

### 6.5.1 Single model approach

For this experiment, we considered 100 random samples for each pair. For the pairs with less than 100 samples, we up-sampled by duplicating some instances. This is done for all 4 DA pairs, for each of the datasets. Both of the sub-sampled datasets, eventually obtained, are then combined as a single dataset. We chose to do a 80-10-10 split for training, validation, and testing. A single model was trained across this dataset using the annotated data, which was run across 4 DA pairs.

Table 6.9: Deep learning result on mixed dataset

Qstn DA	Ans DA	ROUGE-1	ROUGE-2	Sent Sim
bin	y	0.67	0.44	0.74
bin	n	0.77	0.58	0.82
bin	y-d	0.51	0.32	0.67
bin	n-d	0.55	0.37	0.70

From Table 6.9 we can see promising results. We observed a decent ROUGE-1 score for the auto-generated sentences. The models are able to integrate the sentiment of the sentences giving a good semantic similarity score as well. The input question and answer are separated with a “|” delimiter, as seen in Section 5.7. Possibly because of the limited sentences we have for training, the model has some issues in identifying the separator. We also observed some instances where unrelated words are returned within sentences, and there is repetition of certain words. We believe all these problems can be solved if there is more data for each pair and if the training period is increased.

In addition to this, we wanted to observe whether POS tags have any impact on the transformations or not. We pre-processed the input questions and answers such that each word and

Table 6.10: Deep Learning results on mixed dataset with POS tagging

Qstn DA	Ans DA	ROUGE-1	ROUGE-2	Sent Sim
bin	y	0.74	0.51	0.84
bin	n	0.78	0.59	0.86
bin	y-d	0.56	0.37	0.71
bin	n-d	0.46	0.31	0.69

POS tag is separated by “|”. The question and answer statements were combined together with “\_\_\_\_\_” in between them, acting as a separator. We maintained the same parameters of the model as before and observe results as shown in Table 6.10.

We observe an improvement in the [bin,d] and [bin, y-d] set of QA pairs. We also noted that this model performed well for shorter QA pairs. We couldn’t find any specific reason for a drop in performance for the [bin, n-d] class. We believe that the sentence length and variation within this class was too much for the deep-learning model to handle. Again, as mentioned before, we feel that these results may improve as we increase the number of QA pairs for each DA class.

### 6.5.2 Multi-model approach

For this experiment, we used the M10 dataset. We use all of the samples available in the aforementioned 4 DA classes. We performed a 70-20-10 split for training, validation, and testing. Instead of having just a single model, we developed different models for each type of DA pair. In this case, we developed 4 separate models. Table 6.11 shows the performance of each of the models.

These models perform similarly to the “single-model” approach. The [bin, n] class performs better because the model isn’t confused with [bin|y] statements. The developed models do not do a good job of generating bi-grams, which is largely because of the limited training

Table 6.11: Deep Learning results on M10 dataset

Qstn DA	Ans DA	ROUGE-1	ROUGE-2	Sent Sim
bin	y	0.6	0.38	0.73
bin	n	0.71	0.54	0.83
bin	y-d	0.48	0.26	0.74
bin	n-d	0.44	0.24	0.67

set for each DA class. 100 samples are not enough for a deep learning model. This becomes more pronounced in sequence-to-sequence models where a large number of parameters are in play.

## 6.6 MTurk Study

Another research question we had was **RQ6**: *Can external aid, such as Amazon Mechanical Turk (MTurk), be useful in helping us identify issues in the developed transformation?* This section gives more insight on this.

Amazon Mechanical Turk (MTurk) is a crowd-sourcing website for requesters to hire workers to perform tasks on-demand. MTurks can be used to perform tasks related to images, videos, text classification, annotation, etc. MTurk can be used as a formative and/or summative assessment of our work. There are two separate tasks that we define for the turkers:

- **Annotating QA pairs:** In this task, the turkers were asked to read a question and an answer and provide us with a sentence or a group of sentences that combine both the question and answer. These annotated sentences were considered as ground truth and used to evaluate the auto-generated sentences. This task was done for two reasons: ensuring that there wasn't any bias in the annotation of QA pairs, and assisting us in understanding the transformation rules we might be missing.

- **Rating canonical form of QA pairs:** In this task, turkers were given a question, answer, and the corresponding auto-generated declarative sentence. They had to rate the generated sentence, based on three parameters, namely, grammatical correctness, completeness, and naturalness. The ratings ranged from 1 to 5. Each sentence was rated by 3 different turkers. We used the following steps to decide the rating for each sentence:

1. If at least 2 turkers give the same score, the final score is kept as that one.
2. If all 3 scores are different and at least two of them are less than or equal to 3, we then find the average of the scores and calculate its floor value.
3. If all 3 scores are different and at least two of them are more than 3, we then find the average of the scores and calculate its ceiling value.

For the first task, turkers annotated 1116 records. These QA pairs were selected from a combination of the tobacco and proprietary datasets, with most of the records coming from the latter. Using the annotated canonical form of QA pairs by the turkers, we evaluate the performance of the transformation. Table 6.12 shows the evaluation across 4 different metrics.

Table 6.12: Evaluation against MTurk annotated sentences

<b>Evaluation Metric</b>	<b>Score</b>
ROUGE 1	0.65
ROUGE 2	0.47
Semantic Similarity	0.79
WUP Similarity	0.77

Comparing the annotated sentences and the auto-generated sentences we observed a decent ROUGE 1 score, which was expected as the transformed sentences also have some part of the question and answer. The ROUGE 2 score isn't great because we find many sentences

where bi-grams aren't overlapping. We also observed that there are extra phrases in the transformed sentences which reduce the scores. There were a small number of cases where the ground truth and canonical sentences weren't related semantically. But, a majority of them had a good semantic similarity.

As mentioned previously, turkers were also asked to rate the auto-generated canonical sentences on three parameters, rating them from 1 to 5. The following list explains the mapping of the ratings for a particular parameter [28].

- **Grammatical Correctness:** This rating is used to check whether the sentence makes grammatical/lexical sense or not.

- 1- Extremely poor.
- 2- Poor.
- 3- OK but has some issue(s).
- 4- Good but slightly unnatural.
- 5- Good.

Table 6.13: Good and bad examples of grammatically correct sentences

	<b>Question</b>	<b>Answer</b>	<b>Canonical Statement</b>
<b>Good Example</b>	are you currently taking any medication?	no, sir.	I am not currently taking any medication.
<b>Bad Example</b>	is this currently listed in your answer to interrogatory number 21?	not that i see.	not is this currently listed in my answer to interrogatory number 21.

- **Naturalness/Readability:** This rating is used to identify whether the sentences generated are natural or readable.

- 1- Extremely unnatural.

- 2- Unnatural.
- 3- OK in some contexts.
- 4- Natural, but could be more so.
- 5- Very natural.

Table 6.14: Good and bad examples of natural/readable sentences

	<b>Question</b>	<b>Answer</b>	<b>Canonical Statement</b>
<b>Good Example</b>	Um-hum. Do you know if there were snowbanks on the ground on January 16?	I don't remember.	i do not remember if there were snowbanks on the ground on january 16.
<b>Bad Example</b>	Um-hum. Do you know if there were snowbanks on the ground on January 16?	I don't remember.	i do not if do me know if there were snowbanks on the ground on january 16

- **Completeness:** This rating allows us to understand whether the canonical sentences grab the context of the question and answer.

- 1- Lacks many important words from the question or the answer.
- 2- Lacks a few important words from the question or the answer.
- 3- The sentence is missing one or two words that would add more information, but they aren't necessary.
- 4- The sentence is missing one or two words but it still conveys the same meaning without them.
- 5- The sentence is maximally complete in terms of words (regardless of grammatical correctness).

Fig. 6.5 shows the distribution of the ratings of the canonical sentences. Looking at the “Completeness” rating, we observed that the sentences were able to convey the meaning of

Table 6.15: Good and bad examples of completeness of sentences

	Question	Answer	Canonical Statement
<b>Good Example</b>	you made all of your physical therapy appointments?	if not, i called in and rescheduled.	I made all of my physical therapy appointments. if not, i called in and rescheduled.
<b>Bad Example</b>	do you have any siblings?	i have a little boy.	I do not have any siblings. i have a little boy.

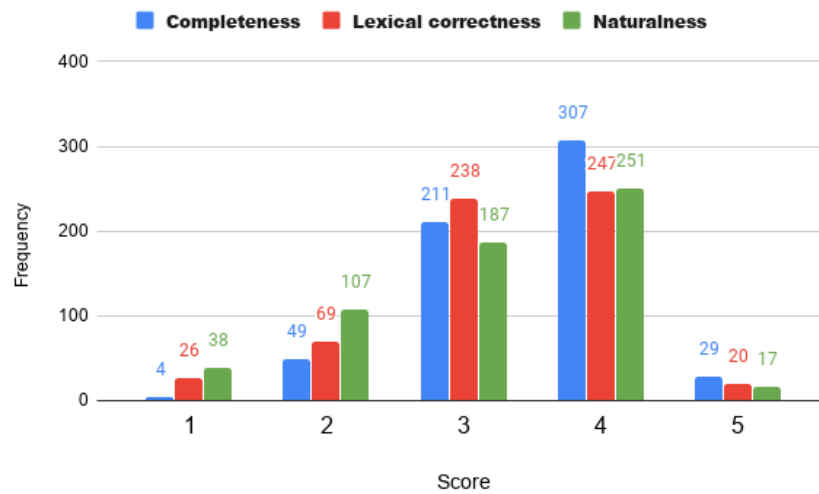


Figure 6.5: Distribution of ratings for auto-generated canonical sentences via MTurk

the QA pair. Many of the sentences did miss some important words which may be useful for better understanding. The “Lexical correctness” rating showed us that there are a small number of sentences having a low score, but many of them did make some grammatical sense. There is scope for improvement in this area. “Naturalness” rating indicated that it also has major scope for improvement. There were quite a few sentences that weren’t readable/natural.

# Chapter 7

## Conclusion and Future work

### 7.1 Conclusion

This work has contributed in many different ways to the field of legal tech. We were able to develop a new parsing technique to handle born-digital PDFs. We also introduce a method for extracting page and line numbers from QA pairs, which may be used for future work such as summarization.

We were able to develop a novel approach for transforming QA pairs in deposition transcripts to their canonical form. We explored both rule-based and deep-learning-based systems. Our analysis on the proprietary dataset showed that the rule-based method performed quite well, beating out simplistic techniques for most of the DA classes. Even for the ones in which it is out-performed by other techniques, it posts reasonably close results. Observing that the performance is also good on the tobacco dataset, we can conclude that the rules developed are generalized enough. Through this work, we also introduced the concept of deep learning, showing a preliminary analysis that such techniques may be fruitful in performing transformations.

Furthermore, we researched sentence correction techniques, analyzing if there are any major improvements. We observed minute improvements quantitatively but qualitatively the sentence was much better. Finally, we used crowd-sourcing to judge the performance of the

transformation and the areas which need improvement. We concluded that the canonical sentences have the correct words but need some improvement lexically and in naturalness.

## 7.2 Research Contributions

The following are the research contributions of this work:

- This work introduces a new technique to parse the previously parsed PDF documents, thus resolving **RQ1**. It also provides an opportunity to process born-digital PDF files. Further, Amazon Textract allows us to improve the quality of the text identified within the PDF file being parsed, allowing for better classification, thereby, better transformations, confirming hypothesis 1.
- Through this work, we implement a method to identify page and line numbers of the QA pair within a document. Results are stored in JSON structures with the respective QA pairs, which may be used in the future for a page and line-number summarizer.
- This work introduces a type of transformational grammar into the field of legal depositions. We utilize simple chunking techniques to split QAs and transform them into their canonical forms, reinforcing our second hypothesis. This work involved looking at the roots of sentences and understanding how sentences are developed and constructed. The promising results, both qualitatively and quantitatively, address **RQ2** and confirm hypothesis 2 and 5.
- This work also allowed the introduction of a modular framework to develop rules for QA pair transformations, and to ease integrating them into the main framework. An end-to-end pipeline has also been introduced within which a PDF file simply needs to

be given as input and a CSV file with corresponding canonical sentences of the QA pairs is generated.

- This work allowed the generation of QA pair datasets for transformation within the legal domain. The dataset size may be small, but through more contributions this may be improved. This may allow the extension to other QA based documents as well.
- Through this work we introduced the concept of deep-learning for transformational grammar. We approached transformation of QA pairs into canonical forms as an NMT problem, allowing us to confirm further our hypothesis, and resolve **RQ4**. We also introduce POS tags into the deep-learning architecture, showing that there is scope for such solutions.
- Another contribution of this work is the research done to introduce some form of sentence correction on the auto-generated sentences to improve performance. We utilize BERT's and AI-GPT2's internal scoring mechanism of words to judge whether sentences could be correct or not, while introducing new heuristics as well. The paltry improvements disproves our hypothesis on sentence correction, but there is still scope for improvement for resolving **RQ3**.
- We introduce ratings provided by MTurkers to judge the quality of transformation. These ratings cover the essence of a sentence, giving a good overall understanding regarding the transformation. We also generate a dataset of canonical sentences annotated by specialized turkers. This allows us to compare the quality of transformations (quantitatively) with an unbiased dataset. The positive feedback obtained from turkers through their ratings, as well as the annotations provided, allowed us to improve the transformation and identify errors within the system. Accordingly we have confirmation of our hypothesis on AMT, and support for resolving **RQ6**.

## 7.3 Future Work

There are various avenues through which our work can be improved.

- Since this work is heavily dependent on the dialog act classifier, any improvement in the classifier would result in an improvement in the transformations.
- For the born-digital files, an improvement in the box-detection algorithm would help in identifying the boundaries of the sub-pages which need to be processed. Better identification would allow for better parsing.
- For the rule-based transformations, instead of limiting the number of rules, we can develop a larger set of rules to handle each and every type of sentence.
- For both the proprietary and tobacco datasets, using the newer parsing technique may allow for better classification of sentences thereby improving the transformation as well.
- For transformation through deep learning, we would need to add a lot more data points (at least thousands) for training models. More data will allow the model to understand the transformation and structures as well.
- Another idea that may help in the transformation could be integrating POS tags within the deep learning architectures (from scratch). This could allow a model to understand the essence and sequence of POS tags.
- This work can be used as a precursor to summarization and be used to identify important parts of the deposition.
- People can apply the work based on the [Appendix A](#), and extend it based on the [Appendix B](#).

- Appendix [D](#) explains the need for future work regarding improving the speed of processing, ensuring privacy, etc.

# Bibliography

- [1] GROBID: Generation of bibliographic data. <https://github.com/kermitt2/grobid>, 2008–2020. Accessed: 2020-06-17.
- [2] Amazon Textract, 2020. URL <https://aws.amazon.com/textract/>. Accessed: 2020-06-17.
- [3] Alan Akbik, Duncan Blythe, and Roland Vollgraf. Contextual string embeddings for sequence labeling. In *COLING 2018, 27th International Conference on Computational Linguistics*, pages 1638–1649, 2018.
- [4] Jeremy Ang, Yang Liu, and Elizabeth Shriberg. Automatic dialog act segmentation and classification in multiparty meetings. In *2005 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '05, Philadelphia, Pennsylvania, USA, March 18-23, 2005*, pages 1061–1064. IEEE, 2005. doi: 10.1109/ICASSP.2005.1415300. URL <https://doi.org/10.1109/ICASSP.2005.1415300>.
- [5] Abhijeet Awasthi, Sunita Sarawagi, Rasna Goyal, Sabyasachi Ghosh, and Vihari Piratla. Parallel iterative edit models for local sequence transduction. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 4259–4269. Association for Computational Linguistics, 2019. doi: 10.18653/v1/D19-1435. URL <https://doi.org/10.18653/v1/D19-1435>.
- [6] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In Yoshua Bengio and Yann LeCun, editors,

- 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL <http://arxiv.org/abs/1409.0473>.
- [7] Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics. URL <https://www.aclweb.org/anthology/W05-0909>.
- [8] Colin Bannard and Chris Callison-Burch. Paraphrasing with bilingual parallel corpora. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, ACL '05*, page 597–604, USA, 2005. Association for Computational Linguistics. doi: 10.3115/1219840.1219914. URL <https://doi.org/10.3115/1219840.1219914>.
- [9] James R. Barlow. OCRmyPDF, 2020. URL <https://ocrmypdf.readthedocs.io/en/latest/introduction.html>. Accessed: 2020-06-17.
- [10] Ian Beaver. PyContractions, 2018. URL <https://github.com/ian-beaver/pycontractions>. Accessed: 2020-06-17.
- [11] Bibek Behera and Pushpak Bhattacharyya. Automated grammar correction using hierarchical phrase-based statistical machine translation. In *Sixth International Joint Conference on Natural Language Processing, IJCNLP 2013, Nagoya, Japan, October 14-18, 2013*, pages 937–941. Asian Federation of Natural Language Processing / ACL, 2013. URL <https://www.aclweb.org/anthology/I13-1122/>.
- [12] Edouard Belval. pdf2image, 2019. URL <https://pdf2image.readthedocs.io/en/latest/index.html>. Accessed: 2020-06-17.

- [13] Steven Bird and Edward Loper. NLTK: The natural language toolkit. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, pages 214–217, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL <https://www.aclweb.org/anthology/P04-3031>.
- [14] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit*. O’Reilly Media, Inc., 2009.
- [15] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017. ISSN 2307-387X.
- [16] Gary R. Bradski and Adrian Kaehler. *Learning OpenCV - computer vision with the OpenCV library: software that sees*. O’Reilly, 2008. ISBN 978-0-596-51613-0. URL <http://www.oreilly.de/catalog/9780596516130/index.html>.
- [17] Christopher Bryant, Mariano Felice, Øistein E. Andersen, and Ted Briscoe. The BEA-2019 shared task on grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 52–75, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4406. URL <https://www.aclweb.org/anthology/W19-4406>.
- [18] Flora Ramírez Bustamante and Fernando Sánchez León. GramCheck: A grammar and style checker. In *16th International Conference on Computational Linguistics, Proceedings of the Conference, COLING 1996, Center for Sprogteknologi, Copenhagen, Denmark, August 5-9, 1996*, pages 175–181, 1996. URL <https://www.aclweb.org/anthology/C96-1031/>.
- [19] G Krishna Chaitanya and P Bhattacharyya. Grammatical error correction. *Indian Institute of Technology Bombay*, 2017. Accessed: 2020-06-17.

- [20] Saurabh Chakravarty, Raja Venkata Satya Phanindra Chava, and Edward A. Fox. Dialog acts classification for question-answer corpora. In Kevin D. Ashley, Katie Atkinson, Luther Karl Branting, Enrico Francesconi, Matthias Grabmair, Bernhard Waltl, Vern R. Walker, and Adam Zachary Wyner, editors, *Proceedings of the Third Workshop on Automated Semantic Analysis of Information in Legal Texts co-located with the 17th International Conference on Artificial Intelligence and Law (ICAIL 2019), Montreal, QC, Canada, June 21, 2019*, volume 2385 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2019. URL <http://ceur-ws.org/Vol-2385/paper6.pdf>.
- [21] Saurabh Chakravarty, Maanav Mehrotra, Raja Venkata Satya Phanindra Chava, Han Liu, Matthew Krivansky, and Edward A. Fox. Improving the processing of question answer based legal documents. In Michal Araszkievicz and Víctor Rodríguez-Doncel, editors, *Legal Knowledge and Information Systems - JURIX 2019: The Thirty-second Annual Conference, Madrid, Spain, December 11-13, 2019*, volume 322 of *Frontiers in Artificial Intelligence and Applications*, pages 13–22. IOS Press, 2019. doi: 10.3233/FAIA190302. URL <https://doi.org/10.3233/FAIA190302>.
- [22] R. Chandrasekar, Christine Doran, and B. Srinivas. Motivations and methods for text simplification. In *COLING 1996 Volume 2: The 16th International Conference on Computational Linguistics*, 1996. URL <https://www.aclweb.org/anthology/C96-2183>.
- [23] Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. One billion word benchmark for measuring progress in statistical language modeling. In *INTERSPEECH 2014, 15th Annual Conference of the International Speech Communication Association, Singapore, September 14-18, 2014*, pages 2635–2639, 2014. URL [http://www.isca-speech.org/archive/interspeech\\_2014/i14\\_2635.html](http://www.isca-speech.org/archive/interspeech_2014/i14_2635.html).

- [24] Zheqian Chen, Rongqin Yang, Zhou Zhao, Deng Cai, and Xiaofei He. Dialogue act recognition via CRF-attentive structured network. In Kevyn Collins-Thompson, Qiaozhu Mei, Brian D. Davison, Yiqun Liu, and Emine Yilmaz, editors, *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, pages 225–234. ACM, 2018. doi: 10.1145/3209978.3209997. URL <https://doi.org/10.1145/3209978.3209997>.
- [25] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. In Dekai Wu, Marine Carpuat, Xavier Carreras, and Eva Maria Vecchi, editors, *Proceedings of SSST@EMNLP 2014, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation, Doha, Qatar, 25 October 2014*, pages 103–111. Association for Computational Linguistics, 2014. doi: 10.3115/v1/W14-4012. URL <https://www.aclweb.org/anthology/W14-4012/>.
- [26] Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. URL <https://www.aclweb.org/anthology/D17-1070>.
- [27] Daniel Dahlmeier and Hwee Tou Ng. A beam-search decoder for grammatical error correction. In Jun’ichi Tsujii, James Henderson, and Marius Pasca, editors, *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL 2012, July 12-14, 2012, Jeju Island, Korea*, pages 568–578. ACL, 2012. URL <https://www.aclweb.org/anthology/D12-1052/>.

- [28] Dorottya Demszky, Kelvin Guu, and Percy Liang. Transforming question answering datasets into natural language inference datasets. *CoRR*, abs/1809.02922, 2018. URL <http://arxiv.org/abs/1809.02922>.
- [29] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL <https://www.aclweb.org/anthology/N19-1423>.
- [30] Li Dong, Jonathan Mallinson, Siva Reddy, and Mirella Lapata. Learning to paraphrase for question answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 875–886, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1091. URL <https://www.aclweb.org/anthology/D17-1091>.
- [31] C.C. Elwell and R.B. Smith. *Practical Legal Writing for Legal Assistants*. Delmar Cengage Learning, 1996. ISBN 9780314061157. URL <https://books.google.com/books?id=UCTQB8wtt1UC>.
- [32] Raul Fernandez and Rosalind W Picard. Dialog act classification from prosodic features using support vector machines. In *Speech Prosody*, pages 291–294, 2002.
- [33] Jenny Rose Finkel, Trond Grenager, and Christopher D. Manning. Incorporating non-local information into information extraction systems by Gibbs sampling. In Kevin Knight, Hwee Tou Ng, and Kemal Oflazer, editors, *ACL 2005, 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the*

- Conference, 25-30 June 2005, University of Michigan, USA*, pages 363–370. The Association for Computer Linguistics, 2005. doi: 10.3115/1219840.1219885. URL <https://www.aclweb.org/anthology/P05-1045/>.
- [34] The Apache Software Foundation. Apache Tika, 2007. URL <http://tika.apache.org/>. Accessed: 2020-06-17.
- [35] Michelle A. Fox. Syllable-final /s/ lenition in the LDC’s callhome Spanish corpus. In *Sixth International Conference on Spoken Language Processing, ICSLP 2000 / INTER-SPEECH 2000, Beijing, China, October 16-20, 2000*, pages 556–559. ISCA, 2000. URL [http://www.isca-speech.org/archive/icslp\\_2000/i00\\_1556.html](http://www.isca-speech.org/archive/icslp_2000/i00_1556.html).
- [36] Roman Grundkiewicz and Marcin Junczys-Dowmunt. Near human-level performance in grammatical error correction with hybrid machine translation. In Marilyn A. Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 284–290. Association for Computational Linguistics, 2018. doi: 10.18653/v1/n18-2046. URL <https://doi.org/10.18653/v1/n18-2046>.
- [37] Roman Grundkiewicz, Marcin Junczys-Dowmunt, and Kenneth Heafield. Neural grammatical error correction systems with unsupervised pre-training on synthetic data. In Helen Yannakoudakis, Ekaterina Kochmar, Claudia Leacock, Nitin Madnani, Ildikó Pilán, and Torsten Zesch, editors, *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications, BEA@ACL 2019, Florence, Italy, August 2, 2019*, pages 252–263. Association for Computational Linguistics, 2019. doi: 10.18653/v1/w19-4427. URL <https://doi.org/10.18653/v1/w19-4427>.
- [38] Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O. K. Li. Incorporating copying mech-

- anism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics, 2016. doi: 10.18653/v1/p16-1154. URL <https://doi.org/10.18653/v1/p16-1154>.
- [39] Na-Rae Han, Martin Chodorow, and Claudia Leacock. Detecting errors in English article usage with a maximum entropy classifier trained on a large, diverse corpus. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation, LREC 2004, May 26-28, 2004, Lisbon, Portugal*. European Language Resources Association, 2004. URL <http://www.lrec-conf.org/proceedings/lrec2004/summaries/695.htm>.
- [40] Kenneth Heafield. KenLM: Faster and smaller language model queries. In Chris Callison-Burch, Philipp Koehn, Christof Monz, and Omar Zaidan, editors, *Proceedings of the Sixth Workshop on Statistical Machine Translation, WMT@EMNLP 2011, Edinburgh, Scotland, UK, July 30-31, 2011*, pages 187–197. Association for Computational Linguistics, 2011. URL <https://www.aclweb.org/anthology/W11-2123/>.
- [41] Matthew Honnibal and Ines Montani. spaCy Industrial-strength Natural Language Processing in Python, 2016. URL <https://spacy.io/>. Accessed: 2020-06-17.
- [42] Gang Ji and Jeff A. Bilmes. Dialog act tagging using graphical models. In *2005 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '05, Philadelphia, Pennsylvania, USA, March 18-23, 2005*, pages 33–36. IEEE, 2005. doi: 10.1109/ICASSP.2005.1415043. URL <https://doi.org/10.1109/ICASSP.2005.1415043>.
- [43] Swaathi Kakarla. Natural language processing: NLTK vs. spaCy, May

2020. URL <https://www.activestate.com/blog/natural-language-processing-nltk-vs-spacy/>. Accessed: 2020-06-17.
- [44] Nal Kalchbrenner and Phil Blunsom. Recurrent convolutional neural networks for discourse compositionality. In Alexandre Allauzen, Hugo Larochelle, Christopher D. Manning, and Richard Socher, editors, *Proceedings of the Workshop on Continuous Vector Space Models and their Compositionality, CVSM@ACL 2013, Sofia, Bulgaria, August 9, 2013*, pages 119–126. Association for Computational Linguistics, 2013. URL <https://www.aclweb.org/anthology/W13-3214/>.
- [45] Su Nam Kim, Lawrence Cavedon, and Timothy Baldwin. Classifying dialogue acts in one-on-one live chats. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, EMNLP 2010, 9-11 October 2010, MIT Stata Center, Massachusetts, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 862–871. ACL, 2010. URL <https://www.aclweb.org/anthology/D10-1084/>.
- [46] Guillaume Klein, Yoon Kim, Yuntian Deng, Vincent Nguyen, Jean Senellart, and Alexander M. Rush. OpenNMT: Neural machine translation toolkit. In Colin Cherry and Graham Neubig, editors, *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas, AMTA 2018, Boston, MA, USA, March 17-21, 2018 - Volume 1: Research Papers*, pages 177–184. Association for Machine Translation in the Americas, 2018. URL <https://www.aclweb.org/anthology/W18-1817/>.
- [47] Tim Kleinschmidt. Gingerit, 2015. URL <https://gingerit.readthedocs.io/en/latest/readme.html>. Accessed: 2020-06-17.
- [48] Marios Koniaris, George Papastefanatos, and Yannis Vassiliou. Towards automatic structuring and semantic indexing of legal documents. In *Proceedings of the 20th Pan-Hellenic Conference on Informatics, PCI '16, New York, NY, USA, 2016*. Association

- for Computing Machinery. ISBN 9781450347891. doi: 10.1145/3003733.3003801. URL <https://doi.org/10.1145/3003733.3003801>.
- [49] Pavel Král and Christophe Cerisara. Automatic dialogue act recognition with syntactic features. *Lang. Resour. Evaluation*, 48(3):419–441, 2014. doi: 10.1007/s10579-014-9263-6. URL <https://doi.org/10.1007/s10579-014-9263-6>.
- [50] Harshit Kumar, Arvind Agarwal, Riddhiman Dasgupta, and Sachindra Joshi. Dialogue act sequence labeling using hierarchical encoder with CRF. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 3440–3447. AAAI Press, 2018. URL <https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16706>.
- [51] Matt J. Kusner, Yu Sun, Nicholas I. Kolkin, and Kilian Q. Weinberger. From word embeddings to document distances. In Francis R. Bach and David M. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 957–966. JMLR.org, 2015. URL <http://proceedings.mlr.press/v37/kusnerb15.html>.
- [52] UCSF Library and Center for Knowledge Management. Truth Tobacco Industry Documents, 2002. URL <https://www.industrydocuments.ucsf.edu/tobacco/>. Accessed: 2020-06-17.
- [53] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text*

- Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL <https://www.aclweb.org/anthology/W04-1013>.
- [54] Yang Liu. Using SVM and error-correcting codes for multiclass dialog act classification in meeting corpus. In *INTERSPEECH 2006 - ICSLP, Ninth International Conference on Spoken Language Processing, Pittsburgh, PA, USA, September 17-21, 2006*. ISCA, 2006. URL [http://www.isca-speech.org/archive/interspeech\\_2006/i06\\_1306.html](http://www.isca-speech.org/archive/interspeech_2006/i06_1306.html).
- [55] Yang Liu, Kun Han, Zhao Tan, and Yun Lei. Using context information for dialog act classification in DNN framework. In Martha Palmer, Rebecca Hwa, and Sebastian Riedel, editors, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 2170–2178. Association for Computational Linguistics, 2017. doi: 10.18653/v1/d17-1231. URL <https://doi.org/10.18653/v1/d17-1231>.
- [56] Marion Mast, Ralf Kompe, Stefan Harbeck, Andreas Kießling, Heinrich Niemann, Elmar Nöth, Ernst Günter Schukat-Talamazzini, and Volker Warnke. Dialog act classification with the help of prosody. In *The 4th International Conference on Spoken Language Processing, Philadelphia, PA, USA, October 3-6, 1996*. ISCA, 1996. URL [http://www.isca-speech.org/archive/icslp\\_1996/i96\\_1732.html](http://www.isca-speech.org/archive/icslp_1996/i96_1732.html).
- [57] Chris Mattmann. Tika-Python, 2015. URL <https://github.com/chris mattmann/tika-python>. Accessed: 2020-06-17.
- [58] Chris Mattmann and Jukka Zitting. *Tika in Action*. Manning Publications Co., USA, 2011. ISBN 1935182854.
- [59] Steven Myint. Python wrapper for LanguageTool, 2017. URL <https://github.com/myint/language-check>. Accessed: 2020-06-17.

- [60] Daniel Naber. A rule-based style and grammar checker. Master's thesis, Universität Bielefeld, 2003. URL <http://www.danielnaber.de/publications>.
- [61] Hwee Tou Ng, Siew Mei Wu, Ted Briscoe, Christian Hadiwinoto, Raymond Hendy Susanto, and Christopher Bryant. The CoNLL-2014 shared task on grammatical error correction. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–14, Baltimore, Maryland, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/W14-1701. URL <https://www.aclweb.org/anthology/W14-1701>.
- [62] Christos H. Papadimitriou, Hisao Tamaki, Prabhakar Raghavan, and Santosh Vempala. Latent semantic indexing: A probabilistic analysis. In *Proceedings of the Seventeenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, PODS '98, page 159–168, New York, NY, USA, 1998. Association for Computing Machinery. ISBN 0897919963. doi: 10.1145/275487.275505. URL <https://doi.org/10.1145/275487.275505>.
- [63] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL <https://www.aclweb.org/anthology/P02-1040>.
- [64] Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1162. URL <https://www.aclweb.org/anthology/D14-1162>.

- [65] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019. URL [https://d4mucfpksywv.cloudfront.net/better-language-models/language\\_models\\_are\\_unsupervised\\_multitask\\_learners.pdf](https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf). Accessed: 2020-06-17.
- [66] Raja Venkata Satya Phanindra Chava. Natural Language Processing Techniques for Comprehending Legal Depositions. Technical report, Virginia Tech, 2013. ECE 5904 MS Project and Report.
- [67] Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for SQuAD. *CoRR*, abs/1806.03822, 2018. URL <http://arxiv.org/abs/1806.03822>.
- [68] Nithin Ramacandran. Dialogue act detection from human-human spoken conversations. *International Journal of Computer Applications*, 67(5), 2013. URL <https://www.ijcaonline.org/archives/volume67/number5/11392-6688>.
- [69] Radim Řehůřek and Petr Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May 2010. ELRA. URL <http://is.muni.cz/publication/884893/en>.
- [70] K. Ries. HMM and neural network based speech act detection. In *Proceedings of the Acoustics, Speech, and Signal Processing, 1999. on 1999 IEEE International Conference - Volume 01*, ICASSP '99, page 497–500, USA, 1999. IEEE Computer Society. ISBN 0780350413. doi: 10.1109/ICASSP.1999.758171. URL <https://doi.org/10.1109/ICASSP.1999.758171>.

- [71] Ellen Riloff and Michael Thelen. A rule-based question answering system for reading comprehension tests. In *Proceedings of the 2000 ANLP/NAACL Workshop on Reading Comprehension Tests as Evaluation for Computer-Based Language Understanding Systems - Volume 6*, ANLP/NAACL-ReadingComp '00, page 13–19, USA, 2000. Association for Computational Linguistics. doi: 10.3115/1117595.1117598. URL <https://doi.org/10.3115/1117595.1117598>.
- [72] Lina Maria Rojas-Barahona, Milica Gasic, Nikola Mrksic, Pei-Hao Su, Stefan Ultes, Tsung-Hsien Wen, and Steve J. Young. Exploiting sentence and context representations in deep neural models for spoken language understanding. In Nicoletta Calzolari, Yuji Matsumoto, and Rashmi Prasad, editors, *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 258–267. ACL, 2016. URL <https://www.aclweb.org/anthology/C16-1025/>.
- [73] Alla Rozovskaya and Dan Roth. Algorithm selection and model adaptation for ESL correction tasks. In Dekang Lin, Yuji Matsumoto, and Rada Mihalcea, editors, *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA*, pages 924–933. The Association for Computer Linguistics, 2011. URL <https://www.aclweb.org/anthology/P11-1093/>.
- [74] Abigail See, Peter J. Liu, and Christopher D. Manning. Get to the point: Summarization with pointer-generator networks. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1073–1083. Association for Computational Linguistics, 2017. doi: 10.18653/v1/P17-1099. URL <https://doi.org/10.18653/v1/P17-1099>.

- [75] Minglai Shao and Liangxi Qin. Text similarity computing based on LDA topic model and word co-occurrence. In *2014 2nd International Conference on Software Engineering, Knowledge Engineering and Information Engineering (SEKEIE 2014)*. Atlantis Press, 2014.
- [76] R. Smith. An overview of the tesseract OCR engine. In *9th International Conference on Document Analysis and Recognition (ICDAR 2007), 23-26 September, Curitiba, Paraná, Brazil*, pages 629–633. IEEE Computer Society, 2007. doi: 10.1109/ICDAR.2007.4376991. URL <https://doi.org/10.1109/ICDAR.2007.4376991>.
- [77] Ray Smith. Tesseract-OCR documentation, 2005. URL <https://tesseract-ocr.github.io/>. Accessed: 2020-06-17.
- [78] Matthew Stamy. PyPDF2 Project, 2016. URL <http://mstamy2.github.io/PyPDF2/>. Accessed: 2020-06-17.
- [79] Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca A. Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie Meteer. Dialogue act modeling for automatic tagging and recognition of conversational speech. *CoRR*, cs.CL/0006023, 2000. URL <https://arxiv.org/abs/cs/0006023>.
- [80] Anand Venkataraman, Luciana Ferrer, Andreas Stolcke, and Elizabeth Shriberg. Training a prosody-based dialog act tagger from unlabeled data. In *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '03, Hong Kong, April 6-10, 2003*, pages 272–275. IEEE, 2003. doi: 10.1109/ICASSP.2003.1198770. URL <https://doi.org/10.1109/ICASSP.2003.1198770>.
- [81] Kanan Vyas. A Box detection algorithm for any image containing boxes, Sep 2019. URL <https://medium.com/coinmonks/a-box-detection-algorithm-for-any-image-containing-boxes-756c15d7ed26>. Accessed: 2020-06-17.

- [82] Evan Pete Walsh. Incorporating a copy mechanism into sequence-to-sequence models, Mar 2020. URL <https://medium.com/@epwalsh10/incorporating-a-copy-mechanism-into-sequence-to-sequence-models-40917280b89d>. Accessed: 2020-06-17.
- [83] Wikipedia contributors. Adobe Acrobat — Wikipedia, the free encyclopedia, 2020. URL [https://en.wikipedia.org/w/index.php?title=Adobe\\_Acrobat&oldid=965249838](https://en.wikipedia.org/w/index.php?title=Adobe_Acrobat&oldid=965249838). Accessed: 2020-06-17.
- [84] Zhibiao Wu and Martha Palmer. Verbs semantics and lexical selection. In *Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics*, ACL '94, page 133–138, USA, 1994. Association for Computational Linguistics. doi: 10.3115/981732.981751. URL <https://doi.org/10.3115/981732.981751>.
- [85] Zheng Yuan, Ted Briscoe, and Mariano Felice. Candidate re-ranking for SMT-based grammatical error correction. In Joel R. Tetreault, Jill Burstein, Claudia Leacock, and Helen Yannakoudakis, editors, *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications, BEA@NAACL-HLT 2016, June 16, 2016, San Diego, California, USA*, pages 256–266. The Association for Computer Linguistics, 2016. doi: 10.18653/v1/w16-0530. URL <https://doi.org/10.18653/v1/w16-0530>.

# Appendices

# Appendix A

## User Manual

It is quite easy for any new user to use the code written by us. Along with separate steps, there is an end to end pipeline also developed for users to simply run and obtain the canonical form of QA pairs. The steps to follow are:

1. Request access to GitLab repositories (deposition-summarization/scratch; saurabc/sentence-correction; maanav/eval-scratch) from Dr. Fox.
2. Download the scratch repository anywhere (preferably in the home folder).

Create a folder `new_sentence_correction` in the home folder and download the contents of the sentence-correction repository into it.

Download the contents of the eval-scratch repo into `Downloads/rouge_2_scores`.

3. Use the `requirements.txt` present in the scratch folder to download the needed Python libraries.
4. Sequentially read the 3 `README.md` for setting up individual steps of the pipeline:
  - `scratch/preprocessing/parsing`
  - `scratch/DA_classification`
  - `scratch/SentenceTransformation`

5. Once your environment is set up, copy the deposition transcript which needs to be processed into the preprocessing/parsing folder.
6. Come back to the main repository and run the shell\_python. For the file which needs to be processed, enter the file name with its path and extension.

The transformed file is stored in scratch/data/Answer\_extraction.

For evaluation, please create a new folder within Downloads/rouge\_2\_scores and copy the scratch/data/Answer\_extraction/declarative\_sentences.csv to it. From the rouge\_2\_scores folder, you can simply run evaluation.py while giving input arguments ground\_truth and auto\_generated file names. A CSV file will be generated with the evaluation outputs. Please check the README within eval-scratch for more details.

For running sentence correction, first, run shell\_python from scratch. Create a new folder on Desktop called FINAL\_ANNOTATION. Within that, create a folder, e.g., Tobacco, and a sub-folder within that as Done. Paste the ground truth CSV file with the following format “ID|Question|Answer|Declarative Sentence”. Rename this file as defined\_QA1-piped\_annot.csv and execute the Python script v2\_shell\_python\_BERT.py, from the scratch folder, giving as input name defined\_QA1-piped.csv. This will run the code and perform sentence correction using the BERT scoring mechanism and also run the evaluation. Depending on the heuristic you want, please make changes in the sub-process call after defining BERT\_path. For more information on the sentence correction file, please read the README of saurabc/sentence-correction.

Note: The CSV data needs to be separated with “|”.

# Appendix B

## Developer Manual

The system has been developed in a modular fashion. The following shows the different modules that are present.

### B.1 deposition-summarization/scratch

#### B.1.1 Parsing and Anonymization

The pre-processing folder holds all of the code related to parsing and anonymization of depositions. In the parsing sub-folder are three files:

- `depositionparser.py`: This code file is responsible for processing input deposition transcripts in PDF format into JSON files that store question-answer pairs.
- `depositionanonymizer.py`: This code file is responsible for identifying for each person their: name, age, address, date, and other private information. These are anonymized to preserve personal privacy.
- `depositionparser_with_line_number.py`: This is a variation of the original `depositionparser.py` file. Along with QA pairs, it identifies starting line numbers and starting page numbers, and stores them in the JSON file.

The folder `pdf_image_processing` employs the newer technique to parse born-digital depositions. It utilizes Amazon Textract for parsing the images.

- `pdf.py`: This script divides each PDF page into a separate PNG image.
- `box_detection.py`: This script is used to identify boxes within each PNG image. This is done to identify the 4 separate pages in a condensed format of a deposition transcript.
- `connecting_aws.py`: This code connects the local system to AWS and runs the Textract service. We obtain the text content of the image in the file `sample.txt`.
- `aws_ocr.py`: This is a post-processing script which reformats `sample.txt`, which can be the input to the pipeline previously implemented.

### B.1.2 Classifier

The `DA_Classification` folder is responsible for the classification of the questions and answers to their corresponding dialog acts. We specifically use BERT in this case. The following are the relevant Python files.

- `input/pre-process.py`: This code file converts the input data to the required format for running BERT.
- `input/pre-process_test.py`: This code file converts the JSON deposition into the input file for predicting the DA of all QAs of the deposition.
- `input/pre-process_test_splitting_large_sentence.py`: This Python script splits large questions which have multiple QAs embedded, based on specific rules mentioned in Section 3.2.

- `input/pre-process_test_with_line_no.py`: This is a simple variation of the `pre-process_test` file which handles 3 or 4 keys instead of two.
- `input/with_page_number.ipynb`: This code takes as input a JSON file in which elements have 4 keys (`question`, `answer`, `start_line_number`, `start_page_number`). It processes this file and returns a JSON with elements having 6 keys (`question`, `answer`, `start_line_number`, `end_line_number`, `start_page_number`, `end_page_number`).
- `define.py`: This file converts the obtained probability file from the BERT classifier into a single dialog act defined by the DA ontology. The final output is stored in `defined_QA1.csv`.

### B.1.3 Transformation

This is the code relevant to the rule-based transformer. It is stored in the `SentenceTransformation/source` folder. This folder has 5 separate sub-folders. Four consist of different chunkers or I/O processing routines, while the framework consists of Python scripts for individual rules.

- `framework/meta_transformer_factory`: This code file processes all the QAs and their dialog acts into their canonical form. This calls upon the mini-transformers depending upon the input.
- `mini-transformers` and corresponding Python files: This refers to all of the smaller transformers that are present for each DA combination. These files can individually be changed without the need to change the `meta_transformer`. Each mini-transformer has 2 files; one is the handler file while the other is the factory file. The handler file is responsible for the transformations.

- Pre-processing files: There is a major pre-processing file (`preprocessor.py`) which handles the most common pre-processing needed, e.g., noise removal, splitting “y-d” or “n-d” sentences, etc. The other pre-processing files are DA specific and are needed to better process that QA pair.
- `framework/post_processor.py`: This code file is responsible for all the post-processing that occurs on the transformed sentences for grammatical correctness and to produce more readable sentences. This includes some noise removal, the substitution of words, and handling of contracted words.
- `framework/dash_post_processor.py`: This script is responsible for handling the post-processing of instances where “- -” is observed. It implements the rules mentioned in Section 5.3.
- `framework/grammar_checker.py`: This utilizes the `Gingerit` and `language-check` libraries to check whether the generated canonical sentences are grammatically correct. We use them to return a score of 0 (if grammatically correct) or 1 (if grammatically incorrect).
- `framework/decoding_BERT_file.py`: This code updates the canonical sentences which are grammatically incorrect with the updated sentences from the BERT scoring mechanism and heuristics.

## B.2 saurabc/sentence-correction

This GitLab repository consists of the code snippets which are connected to sentence correction. The following are some scripts present:

- `ai_gpt_lm_ngram_swap_v2.py`: This script utilizes the AI-GPT2 language model to predict the score of a given sentence. In an n-gram window, swapping takes place and the best sentence is returned.
- `bert_lm_ngram_swap_v2.py`: This script utilizes the BERT language model to predict the score of a given sentence. In an n-gram window, swapping takes place and the best sentence is returned.
- `bert_lm_ngram_swap_v3.py`: This script utilizes the BERT language model to predict the word-normalized score of a given sentence. In an n-gram window, swapping takes place and from the best sentence, a single word is deleted to check whether there is any improvement in the normalized score.
- `bert_lm_ngram_swap_v4.py`: This script utilizes the BERT language model to predict the word-normalized score of a given sentence. In this file, the deletion module is integrated within the n-gram swapping method.

### B.3 `maanav/eval-scratch`

The main code file in this repository is `evaluation.py`. This has embedded all the evaluation metrics we use. It takes as input the ground truth file and the auto-generated declarative file, both of which are pipe (`|`) separated. This also incorporates both the InferSent sentences embeddings as well as the `pythonrouge` library.

# Appendix C

## Other Experiments

### C.1 Parsing born-digital PDF files

#### C.1.1 GROBID on Documents

Legal deposition documents also come in PDF forms. Even though their structure may be slightly different from scientific documents, we wanted to examine whether this library would work. We processed the born-digital PDF files through Adobe's OCR system, and the final file was processed by GROBID.

From the processed file we observed that GROBID performed quite badly. The manner in which GROBID identified text in the document was quite haphazard. GROBID failed to identify the condensed format of the PDF file. We noticed that many pages were not even parsed at all by GROBID. The parsing output varied from page to page as seen in Fig. C.1 and Fig. C.2. Fig. C.1 shows that a part of the page is parsed quite well in a structured manner from which details may be extracted, but then Fig. C.2 shows that there are instances where there are line numbers that come sequentially without any connection to questions or answers.

```

</figure>
<figure
  xmlns="http://www.tei-c.org/ns/1.0" type="table" xml:id="tab_8" validated="false">
  <head></head>
  <label></label>
  <figDesc>Interposing) Yeah. I believe so, yeah. And you pulled up to [REDACTED]; correct? Ye
s. [REDACTED] didn't tie up? I don't remember any lines being tied up. He just pulled up to --I thought to l
et me out. At least a --at least a very quick stop? Yeah, a quick stop. [REDACTED] Page35 1
A. Yes. 2 Q. Leaving the two men on the boat? 3 A. Yes. Although, I don't remember the second man. I don't know where
he was? 5 Q. {Interposing) He may have gotten off or he 7 A. {Interposing) He may have; I don't know. 8 Q. You got off
the boat? 9 A. I got off the boat. 10 Q. And that's where I lost you. I didn't hear squad. And they called --[REDACTED]
[REDACTED] came up in just a few minutes. They asked 15 me a lot of questions about: did you breathe 16 in any water? T
hey looked at my eyes; talked --asked me questions, name, you 18 know, just to see I was making sense. They 19 took me
[REDACTED] and I waited around there for a while and [REDACTED]
and in just a few minutes, [REDACTED] up. And I don't remember if [REDACTED] were with them or no
t. 25 Q. [REDACTED] Yes. How do you spell that? [REDACTED] At that point, did they know [REDACTED] wa
s? No. Did you tell them he left on [REDACTED] or do you recall? I --I'm pretty sure that --I can't quote yo
u, but I'm sure I told them that he had gone with [REDACTED] Do you recall what time this would hav
e been --somewhere in the [REDACTED] </figDesc>
  <table>4
6
stayed?
11
you say you got off the boat.
12 A. Okay. I got off the boat. There were two
13
[REDACTED] there because they'd been

```

Figure C.1: Output of page parsed in GROBID

```
7 Q.  
8 A.  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
  
Page37  
  
</table>  
  </figure>  
  <figure  
    xmlns="http://www.tei-c.org/ns/1.0" type="table" xml:id="tab_9" validated="false">  
    <head></head>  
    <label></label>  
    <figDesc>[REDACTED] 1 Q. So, you went to the optometrist at Wal-Mart? 2 A. Yes. And that's partly why I s  
ay it's 3 sometime around five, because I was concerned they were going to close. I knew it was Saturday. 6 Q. And the  
conversation that you heard [REDACTED] on the phone? 8 A. Yes. 9 Q. And he said, why don't they know
```

Figure C.2: Output of another page parsed in GROBID

### C.1.2 Adding Text layer

We used two different methods to add a text layer over the born digital PDF files. In one of them we used the built-in OCR functionality of Adobe, which can convert a PDF file into text. In the other method, we made use of OCRmyPDF for converting the same file.

<pre> 86 -- some document when ██████████ 87 or something of the sort. And so I called 88 89 her and she told me how to get in touch 90 91 with him. 92 93 What ██████████ at any 94 point, about how the accident occurred on 95 ██████████ 96 97 ██████████ I 98 believe he's already answered that one a 99 couple of times. 100 101 ██████████ 102 103 104 105 106 107 108 109 110 his leg hit the propeller; he got cut by the 111 propeller. And then, he went up on the 112 beach and I remember the skin was just -- 113 there was a flap there and he said he had 114 to hold the flap closed to keep from -- the 115 116 Sa A TRS STEUER SDSS 117 118 Page 74 1, 119 120 bleeding under control. I remember him 121 ██████████ 122 help. And that in general is--- 123 124 ██████████ 125 ██████████ </pre>	<pre> 86 -- some document when ██████████ 87 or something of the sort. And so I called 88 89 her and she told me how to get in touch 90 91 with him. 92 93 What ██████████ at any 94 point, about how the accident occurred on 95 ██████████ 96 97 ██████████ I 98 believe he's already answered that one a 99 couple of times. 100 101 me Sc 102 103 ██████████ 104 105 Q. 106 107 Do you understand the question? 108 109 ██████████ 110 You can answer it again. 111 112 ██████████ I 113 don't remember the specifics. But in 114 general, at some point he got out of the 115 ██████████ ran over him, and 116 his leg hit the propeller; he got cut by the 117 propeller. And then, he went up on the 118 beach and I remember the skin was just -- 119 there was a flap there and he said he had 120 to hold the flap closed to keep from -- the 121 122 Page 74 123 124 bleeding under control. I remember him 125 ██████████ 126 help. And that in general is--- 127 128 ██████████ 129 130 Q. 131 A. 132 133 Q. 134 A. 135 136 </pre>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Figure C.3: OCR output of Abode-OCR (left) and OCRmyPDF (right)

## Analysis

Fig. C.3 shows the output of the two OCR engines on the same deposition page. The sections in grey indicate content that hasn't been identified by the other. The left part of the image exhibits the text parsed by Adobe-OCR. The content highlighted in light red indicates the line in which there is a difference and dark red indicates the actual difference in the words identified. Similarly, on the opposite side, the right-hand part of the image exhibits the text parsed by OCRmyPDF. Light green indicates the lines different from the Adobe-OCR, while the darker shade indicates specific differences within the line. The following are some observations we made when using the Adobe-OCR engine and OCRmyPDF library:

- Adobe-OCR engine and the OCRmyPDF library worked similarly and were able to identify and convert text identified within the scan of the deposition transcript. There were some instances where Adobe-OCR was able to identify more content while for other pages, OCRmyPDF worked well. There were no such patterns, but performance-wise both were similar.
- OCRmyPDF was able to identify the delimiters “Q.” and “A.” within the document, more frequently than Adobe, but both of them didn't perform well on this, possibly because of the poor quality of the document.
- Both systems were able to identify the condensed layout of the document, but also identified some garbage text and random characters and returned them in the OCR output.
- Line numbers weren't identified consistently by both systems. And when they were, there were many cases where line numbers came sequentially at the start of the document, making it relatively useless as we couldn't correlate the text with the line numbers.

## C.2 Transformations

### C.2.1 Next sentence prediction using BERT

Since the rule-based system was developed considering each QA pair independent of other QA pairs, we wanted to also experiment with the possibility that the inter-dependency may help us extract more information. As we previously know, BERT [29] has been developed on the next sentence prediction as well. We hypothesized that using preceding QA pairs can allow us to understand the flow of the document and can identify breaks in it. These breaks could mean a change in topics. We consider a single long deposition transcript for this case.

For cracking this problem, we initially follow the steps to parse the PDF document and identify the classes of the QA pairs. We create two separate arrays for questions and answers separately. Then, we create two new arrays, where one array stores the content of the preceding question and answer followed by the actual question (concatenated together). The other array simply consists of the answer to the final question. We then calculate the intent of the content within the two new arrays. If the contents are related to each other, a high continuity value is returned else a high discontinuity value is returned.

Table C.1: Sample example where discontinuity is high.

Question	Answer	Decl Sent	Continuation	Discontinuation
And you were using it to install gutters, you said?	Yes, sir.	I was using it to install gutters.	0.997	0.003
Who were you working – who were you working for when you did that work?	Brainza International.	I was working for Brainza International at that time.	0.083	0.917

This was done across a single deposition transcript. It was observed that out of 1800 QA

pairs, there were only 43 instances where the discontinuity was more than 0.3 and only 31 instances where the discontinuity was higher than 0.5. A major drawback we observed was that if the previous question/answer was empty then the chances that there is a discontinuity increases. We noticed that in some cases there wasn't specifically a subject change; maybe in the previous statement the company name wasn't mentioned and in the new question, it was. This caused BERT to think of them as separate conditions. The discontinuity cases identified by this method weren't useful due to which this wasn't developed further. Table C.1 shows a sample example of the continuity and discontinuity values calculated.

## C.3 Evaluation

### C.3.1 Flair Embeddings

Flair is an NLP library, developed by Zalando Research, and has released pre-trained models on tasks such as Named-Entity Recognition, Part-of-Speech Tagging, Text classification, and other custom models [3]. This framework also introduces a new set of embeddings called "flair embeddings" which implement the concept of contextual string embeddings. This framework allows us to stack different word-embeddings together, enabling us to optimize which embedding to use for a given task. One of its main advantages is its ease of use.

Our goal of using the flair library was to stack different word-embeddings together and utilize that to embed the auto-generated sentences and ground-truth sentences. Using these two embeddings we wanted to analyze the semantic similarity between them. This library performed quite nicely but had some issues in differentiating between negative intent sentences and positive intent sentences (sentence consisting of "not") giving a relatively high semantic score. In-comparison, InferSent allowed better differentiation due to which we proceeded

with the latter for evaluation purposes.

### C.3.2 Run Time Analysis

Table C.2 shows the time taken to run the end-to-end pipeline from parsing to transforming the QA pairs. The time reported refers to the total elapsed time from the start of the pipeline to that particular step. We report this for each of the 3 documents in the M3 dataset.

Table C.2: Run time for each step in transforming QA pairs (in seconds).

	<b>Doc 1</b>	<b>Doc 2</b>	<b>Doc 3</b>	<b>Average</b>
<b>PDF -&gt; QA Pair</b>	10	25	20	18
<b>QA Pair -&gt; Classification</b>	1300	1410	1350	1353
<b>Classification -&gt; Canonical Form</b>	2250	2060	2335	2215

From the Table C.2 we observed that for long depositions (approx. 450 pages) it takes an average time of around 35 minutes to process. Extracting QA pairs from the PDF file takes a very small amount of time but classifying the QA pairs and transforming them takes more time, as it takes time to load the pre-trained models.

All runtimes are reported from a system having 16GB RAM, i7 7700HQ with 6GB GeForce GTX 1060 graphics card.

### C.3.3 Coverage Analysis

Table C.3 shows the coverage of the rules developed across some of the DA pairs of the M10 dataset.

Barring all of the other DA pairs, from the 8 DA pairs shown in Table C.3, it was observed that the transformation system was able to handle more than 80% of the cases. The [bin-

Table C.3: Coverage of rules across DA pairs.

Qstn DA	Ans DA	Unhandled cases	Total cases	Coverage %
wh	sno	35	517	93.23
bin	y	16	326	95.09
bin-d	y	65	322	79.81
bin	sno	4	277	98.56
bin	n	7	270	97.41
sno	sno	32	159	79.87
wh-d	sno	0	121	100

d, sno] and [wh-d, sno] classes are two sets of QA pairs which have less coverage. This is because QA pairs under these categories are less consistent. There isn't a set of fixed steps or structures followed while appending questions with extra statements, phrases, or sentences. For the remaining DA pairs, we were able to observe some recurring patterns which we converted into rules.

# Appendix D

## Discussion

**Q** What is the intuition on why the various models perform the way they do?

**A** Talking about the models developed through deep learning (see Section 2.5, we simply had an intuition that this problem could be solved using machine translation. We were effectively transforming the QA pairs from one form to another; similar to how sentences can be transformed from one language to another. We had a hunch that by introducing POS tags into the architecture or as input, the models would have more information and might be able to identify underlying patterns within them.

Regarding the sentence correction models, the PIE and KenLM models were baseline models. The KenLM model was trained on the One Billion Word Benchmark dataset[23]. BERT and AIGPT2 are the latest state-of-the-art-language models introduced, so they were expected to perform better than KenLM. The PIE model was trained on the Cambridge English Write & Improve (W&I) corpus and the LOCNESS corpus [17]. This corpus is newer and more varied, due to which models developed on this are more generalizable. We somewhat expected the PIE model to perform well, as it was predicting edits to be made rather than words themselves, allowing it to handle deletion, spell checking, etc.

We found some common issues within the transformed sentences and developed the 3 heuristics. The 2 and 3-gram swaps were simply aimed to structure sentences better; thereby improving the ROUGE 2 scores. The one-word deletion and one-word substi-

tution heuristics show better qualitative results, but the impact is not enough across the whole dataset due to which we do not find a major impact in the quantitative evaluation.

**Q Please explain the intuition on possible underlying or observed interdependence between word-based evaluation (e.g., R1 and R2) and Semantic Similarity.**

**A** The ROUGE evaluation metric, i.e., R1 and R2 scores, are not enough to demonstrate if a transformation is good or not. Generally, when we calculate ROUGE scores we can remove stopwords such: as me, I, or not. But for our case, these words are quite important since they may change the meaning of the sentences. There also could be many cases where the difference between the two sentences is simply the word “not”. It would indicate a high R1 score and a relatively high R2 score as well, but the two sentences may be opposites of each other.

For such cases, we found that some kind of similarity score would help us better understand the canonical sentence and whether we were able to extract the sentiment as well as the crux of the information. From our work, we observed that having a similarity score of 0.9 along with moderately high R1 and R2 scores, indicates a good transformation.

**Q Please elaborate on the trade-off between accuracy and run time of your proposed approach.**

**A** The rule-based system on average takes around 860 seconds to run for long depositions (approx. 450 pages or 1500 QA pairs). The time needed for each QA pair to process through the actual rule would be less than a second, but the post-processing which requires the loading of different models, which takes up more time. The coverage may

change depending on the type of questions asked or the way the answer is given, but the average time to transform each QA pair itself remains the same.

Regarding the deep-learning models, we ran the single-model approach for 35000 steps which took around 3 hours. For this case, accuracy refers to how often the model predicts the correct target token, given the previous correct target sequence. We observed a training accuracy of 65.84% and validation accuracy of 19.68% for the model without POS tag, while a training accuracy of 75.49% and validation accuracy 23.55% was observed for the model with POS tags.

Fig. D.1 shows how accuracy changes across the number of steps. We can observe an increase in training accuracy as the number of steps increases for both models. The validation accuracy for model with POS tags is lower than the other model when reported in the first 5000. Subsequently, the latter model outperforms the former.



Figure D.1: Change of accuracy across steps.

**Q** Beyond PII and anonymization, what other privacy and confidentiality con-

**cerns or implications could arise? How could they be handled? For example, if there is interest in preventing inference of additional information, what future extension of this work can make such guarantees?**

**A** There is much data available through the transcripts of the legal depositions, e.g., date of the proceeding, location of the proceeding, the company transcribing, the person being deposed, deponent information, etc. These are some examples of important details related to the deposition which may be harmful in the wrong hands. The first line of defense is simply to only allow authorized users access to the deposition transcripts. If the user doesn't have access to the deposition no harm can be done. Even if they have access and use the tool, most of the information available through the depositions such as phone numbers, names, locations, company names, etc. are already handled through anonymization.

In the future, if we wish to run this tool through a cloud service, we need to ensure that the data transfers are encrypted and there are multiple layers of firewalls present. It could also be possible to break the data into smaller chunks and then transfer it to ensure all the data isn't captured at once. For an HTTPS connection, we need to ensure it is secure. For the specific data files we store at each step, we can ensure they are stored in some hidden location along with password protection known to authorized users only.

Trying to infer extra details from a set of given QA pairs would be quite useful for us. We would be able to extract what subject matter the QA pairs are referring to. This would help us in two ways: allowing us to integrate that subject into the canonical sentences, and aiding us in the future goal of summarization of legal deposition. This could assist in trial preparation as well. If such can be identified, then methods could be explored that would make sure such inferences could not be correctly made. Further

study is needed, however, regarding privacy, and protection against inference attacks.

# Appendix E

## IRB Approval and Supporting Files

### E.1 VT IRB Authorization Letter



Division of Scholarly Integrity and  
Research Compliance  
Institutional Review Board  
North End Center, Suite 4120 (MC 0497)  
300 Turner Street NW  
Blacksburg, Virginia 24061  
540/231-3732  
irb@vt.edu  
<http://www.research.vt.edu/sirc/hrpp>

#### MEMORANDUM

**DATE:** March 16, 2020  
**TO:** Edward Fox, Maanav Mehrotra, Saurabh Chakravarty  
**FROM:** Virginia Tech Institutional Review Board (FWA00000572, expires October 29, 2024)  
**PROTOCOL TITLE:** Human Evaluation for Transformation of Sentences  
**IRB NUMBER:** 20-143

Effective March 16, 2020, the Virginia Tech Human Research Protection Program (HRPP) and Institutional Review Board (IRB) determined that this protocol meets the criteria for exemption from IRB review under 45 CFR 46.104(d) category(ies) 2(ii).

Ongoing IRB review and approval by this organization is not required. This determination applies only to the activities described in the IRB submission and does not apply should any changes be made. If changes are made and there are questions about whether these activities impact the exempt determination, please submit a new request to the IRB for a determination.

This exempt determination does not apply to any collaborating institution(s). The Virginia Tech HRPP and IRB cannot provide an exemption that overrides the jurisdiction of a local IRB or other institutional mechanism for determining exemptions.

All investigators (listed above) are required to comply with the researcher requirements outlined at:

<https://secure.research.vt.edu/external/irb/responsibilities.htm>

(Please review responsibilities before beginning your research.)

#### PROTOCOL INFORMATION:

Determined As: **Exempt, under 45 CFR 46.104(d) category(ies) 2(ii)**  
Protocol Determination Date: **March 16, 2020**

#### ASSOCIATED FUNDING:

The table on the following page indicates whether grant proposals are related to this protocol, and which of the listed proposals, if any, have been compared to this protocol, if required.

*Invent the Future*

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY  
*An equal opportunity, affirmative action institution*

**SPECIAL INSTRUCTIONS:**

\*\*\* The Virginia Tech IRB/HRPP has requested that research involving person-to-person contact or gatherings of human research participants be paused as soon as possible. The duration of the pause is unknown, but to reduce disruption to the extent possible, we will be reassessing daily. Although we continue to issue approval notices, Virginia Tech guidance should be followed. Please visit <https://www.research.vt.edu/covid-19-updates-impacts.html> for updates.

Date*	OSP Number	Sponsor	Grant Comparison Conducted?

\* Date this proposal number was compared, assessed as not requiring comparison, or comparison information was revised.

If this protocol is to cover any other grant proposals, please contact the HRPP office ([irb@vt.edu](mailto:irb@vt.edu)) immediately.

## E.2 Online Recruitment

**VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY**  
**Online Recruitment**  
**in Research Projects Involving Human Subjects**

**Title of Project:** Human Evaluation for transformation of sentences

**Protocol No.:** IRB #20-143

**Investigator(s):** Principal Investigator:  
Dr. Edward A. Fox (Professor of Computer Science)  
Email: fox@vt.edu Phone: 540-231-5113

Saurabh Chakravarty  
KnowledgeWorks II, Virginia Tech  
Blacksburg, VA 24061, USA  
(5<sup>th</sup> year PhD Student)

Maanav Mehrotra  
2030 Torgersen Hall, Virginia Tech  
Blacksburg, VA 24061, USA  
(2<sup>nd</sup> Year Master's Student)

### I. Introduction

The task is to either manually score a set of machine generated sentences based on grammar and readability or to write the declarative form of a question answer pair. These manually labeled results and annotated sentences will help us to evaluate our proposed transformation model and identify areas of improvement. Participants will need to complete one or more assignments. In each assignment, participants are requested to score 20 transformed sentences or else transform 12 question answer pairs themselves. Participants will not be asked to answer more than 20 questions.

This research is being conducted by Virginia Tech. Before continuing we request the participants to note down the IRB number and the investigator's contact information for any further questions or issues.

### II. Requirement

The following eligibility requirement must be satisfied:

1. Participants should be at least 18-years-old;
2. Participants should have been granted the Mechanical Turk Masters Qualification.

### III. Compensation

Each assignment can be completed within 30 minutes. Each participant will be compensated no more than \$0.30 per question answered.

If you have questions, concerns, or complaints, or think this task has hurt you, talk to the research team using the contact information listed above. This task is being overseen by an Institutional Review Board ("IRB"). An IRB is a group of people who perform independent review of research studies. You may also contact IRB at Virginia Tech at irb@vt.edu if you have questions, concerns, or complaints that are not being answered by the research team or you have questions about your rights as a research subject.

## E.3 Consent Form

### RESEARCH SUBJECT CONSENT FORM

**Title:** Human Evaluation for transformation of sentences

**Protocol No.:** IRB #20-143

**Sponsor:** Mayfair Group LLC

**Investigator:** Dr. Edward A. Fox  
114 McBryde Hall, Dept. of CS, M/C 0106, Virginia Tech  
Blacksburg, VA 24061, USA

**Daytime Phone Number:** 540-231-5113

**Co-Investigator:** Saurabh Chakravarty  
KnowledgeWorks II, Virginia Tech  
Blacksburg, VA 24061, USA  
Maanav Mehrotra  
2030 Torgersen Hall, Virginia Tech  
Blacksburg, VA 24061, USA

You are being invited to take part in a research study. Participation is voluntary. You can choose not to take part, or agree to take part and later change your mind. There will be no penalty or loss of benefits to which you are otherwise entitled.

The purpose of this research is to manually evaluate the machine generated sentences and to aid in future development. We will ask you questions and determine your feedback. Your participation in this research will last until you have completed the questionnaire. The only risk is effort involved in the questionnaire. There are no direct benefits to you from your taking part in this research, but the general public may benefit from the information gained during this research. Your alternative is to not take part in the research. We may publish the results of this research. As we are not collecting any identifiable information, your information will be confidential.

This research is being sponsored by Mayfair Group LLC of which Edward Fox has an ownership/equity interest in.

If you have questions, concerns, or complaints, or think this task has hurt you, talk to the research team using the contact information listed above. This task is being overseen by an Institutional Review Board ("IRB"). An IRB is a group of people who perform independent review of research studies. You may also contact IRB at Virginia Tech at [irb@vt.edu](mailto:irb@vt.edu) if you have questions, concerns, or complaints that are not being answered by the research team or you have questions about your rights as a research subject.

For taking part in this research, you may be paid up to a total of \$0.30 per question answered. You will be answering either a set of 12 or 20 questions. You will not be asked to answer more than 20 questions.

By continuing in the survey, you are consenting to continue.

## E.4 Sample Tasks

**VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY**  
Sample Task  
in Research Projects Involving Human Subjects

**Title:** Human Evaluation for Transformation of Sentences

**Protocol No.:** IRB #20-143

**Sponsor:** Mayfair Group LLC

**Investigator:** Dr. Edward A. Fox  
114 McBryde Hall, Dept. of CS, M/C 0106, Virginia Tech  
Blacksburg, VA 24061, USA

**Daytime Phone Number:** 540-231-5113

**Co-Investigator:** Saurabh Chakravarty  
KnowledgeWorks II, Virginia Tech  
Blacksburg, VA 24061, USA

Maanav Mehrotra  
2030 Torgersen Hall, Virginia Tech  
Blacksburg, VA 24061, USA

The participants will be asked to participate in only one assignment at a time.

If you have questions, concerns, or complaints, or think this task has hurt you, talk to the research team using the contact information listed above. This task is being overseen by an Institutional Review Board (“IRB”). An IRB is a group of people who perform independent review of research studies. You may also contact IRB at Virginia Tech at [irb@vt.edu](mailto:irb@vt.edu) if you have questions, concerns, or complaints that are not being answered by the research team or you have questions about your rights as a research subject.

Please find appended below sample human evaluation assignment.

## Sample Annotating Assignment

**Instructions**

In this task, you need to read a question-answer pair from either the Tobacco deposition documents or from another proprietary source. Then, you are requested to convert the question-answer pair into its declarative form, i.e. combine the question and the answer together in a seamless manner. The main idea is to keep the essence of the subject of the question and answer in the declarative statement. Below are some examples.

**Example:**

Question: did you ever work with org11?

Answer: yes.

Declarative Statement: I did work with org11.

Question: got you. good enough. yes; that's fine. so i guess my question is, why aren't they listed?

Answer: oh, that one in particular, it -- technically, in particular it fell under the denise memory scale.

Declarative Statement: they aren't listed because it fell under the denise memory scale.

Question: have you taught courses in the field of vascular surgery?

Answer: No, not, not courses, but it comes up in teaching about tobacco and vascular disease, that is something we do teach and talk about.

Declarative Statement: i have not taught courses in the field of vascular surgery. but it comes up in teaching about tobacco and vascular disease, that is something we do teach and talk about.

**Annotation Question 1**

Question: you went to use the bathroom upstairs?

Answer: yes.

Declarative Statement:

**Annotation Question 2**

Question: how frequently do you see your son?

Answer: him frequently, like every other week.

Declarative Statement:

**Annotation Question 3**

Question: is it fair to say you don't have an opinion one way or the other as to whether the left-sided features were related to the carbon monoxide exposure?

Answer: i -- you know, i'm not sure if you can -- if you end up seeing an exacerbation of those weaknesses, or i'm not really sure how it --

Declarative Statement:

<b>Annotation Question 4</b>
Question: the one measure which prompted his leaving was the wcst. what is the wcst? Answer: that's the wisconsin card sorting test. Declarative Statement:
<b>Annotation Question 5</b>
Question: is it fair to characterize that that was a routine inspection? Answer: yes. Declarative Statement:
<b>Annotation Question 6</b>
Question: yes, sir, okay. sir, did you or anybody else, to your knowledge, take any photographs of what was done that day? Answer: i took no photos, no. Declarative Statement:
<b>Annotation Question 7</b>
Question: your first sentence. and my question is about the phrase "significantly lower than expected." can you read that and tell me what you meant by that? Answer: statistically predicted. Declarative Statement:
<b>Annotation Question 8</b>
Question: what are the names of some of the parks that you played at? Answer: location6. another park on portsmouth boulevard; i don't recall the name. there's two off portsmouth boulevard. Declarative Statement:
<b>Annotation Question 9</b>
Question: dr. flora, do you know how much market share juul has for e-cigarettes in the united states? Answer: i have i've seen articles in the newspaper with that or with that information. Declarative Statement:
<b>Annotation Question 10</b>
Question: on occasions, are documents sent outside the united states that relate to the manufacture and sale of cigarettes in the united states? Answer: Not that I know of. Anything is possible. Declarative Statement:

<b>Annotation Question 11</b>
Question: did mr. williams ever jump down the stairs at your unit? Answer: no. Declarative Statement:

<b>Annotation Question 12</b>
Question: you were asked some questions about the u.s. food and drug administration's ban on characterizing flavors. do you generally recall those questions? Answer: yes. Declarative Statement:

## Sample Rating Assignment

Instructions
<p>In this task, you need to read a question-answer pair and its declarative form generated by us. Then, you are requested to evaluate the generated declarative sentences based on the following criteria<sup>(1)</sup>:</p> <ul style="list-style-type: none"> <li>• <b>Grammatically:</b> Checking if the grammar of the sentence is correct or not. 1 – Extremely poor, 2 – Poor, 3 – OK but has some issue(s), 4 – Good but slightly unnatural, 5 – Good</li> <li>• <b>Naturalness/Readability:</b> The sentences generated are natural or readable. 1 – Extremely unnatural, 2 – Unnatural, 3 – OK in some contexts, 4 – Natural, but could be more so, 5 – Very natural</li> <li>• <b>Completeness:</b> Whether the question and answer context is grabbed by the declarative statements. 1 – Lacks many important words from the question or the answer, 2 – Lacks a few important words from the question or the answer, 3 – The sentence is missing one or two words that would add more information, but they aren't necessary, 4 – The sentence is missing one or two words but it still conveys the same meaning without them, 5 – The sentence is maximally complete in terms of words (regardless of grammaticality)</li> </ul>

- (1) Demszky, Dorottya, Kelvin Guu, and Percy Liang. "Transforming question answering datasets into natural language inference datasets." *arXiv preprint arXiv:1809.02922* (2018).

Rating 1
<p>Question: Was it any new construction?            Answer: We did some new construction.            Declarative Sentence: it was any new construction. we did some new construction.</p>
<p>Grammatical correctness</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p>
<p>Naturalness of language</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p>
<p>Completeness</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p>

<b>Rating 2</b>
Question: Okay. Do you recall what kind of building you were working on? Answer: It was a assisted living building. Declarative Sentence: i do recall what kind of building i was working on it was a assisted living building.
Grammatical correctness <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5  Naturalness of language <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5  Completeness <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5

<b>Rating 3</b>
Question: And laborers who worked under the foreman? Answer: Yes. Declarative Sentence: Laborers worked under the foreman.
Grammatical correctness <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5  Naturalness of language <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5  Completeness <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5

<b>Rating 4</b>
Question: And was boom lift operation discussed in this safety meetings? Answer: Yes, sir. Declarative Sentence: boom lift was operation discussed in this safety meetings.
Grammatical correctness <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5  Naturalness of language <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5  Completeness <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5

<b>Rating 5</b>
<p>Question: you are familiar with the engle 0 findings?          Answer: Generally, I am. I think I've heard them, yes.          Declarative Sentence: i are familiar with the engle 0 findings.,i am. i think i've heard them yes.</p>
<p>Grammatical correctness</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p> <p>Naturalness of language</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p> <p>Completeness</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p>
<b>Rating 6</b>
<p>Question:you also say that it's not a product standard of that level is not adequately supported by science and evidence?          Answer: Yeah, that's clear. I think that's something we certainly agree with the FDA on.          Declarative Sentence: i also say that it 's not a product standard of that level is not adequately supported by science and evidence.,that's clear. i think that's something we certainly agree with the fda on.</p>
<p>Grammatical correctness</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p> <p>Naturalness of language</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p> <p>Completeness</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p>

<b>Rating 7</b>				
Question: was that filter that you've described one that was ever intended to be used on a consumer cigarette product? Answer: No, it was an experimental cigarette. Declarative Sentence: do not was that filter that i have described one that was ever intended to be used on a consumer cigarette product.,it was an experimental cigarette.				
Grammatical correctness				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5
Naturalness of language				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5
Completeness				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5

<b>Rating 8</b>				
Question: Did you meet him Saturday evening? Answer: No Declarative Sentence: I did not meet him Saturday evening.				
Grammatical correctness				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5
Naturalness of language				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5
Completeness				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5

<b>Rating 9</b>
<p>Question: have you taught courses in epidemiology?                  Answer: Not courses, per se, but we teach about the epidemiology of a variety of things.                  Declarative Sentence: i have not taught courses in epidemiology.,per se but we teach about the epidemiology of a variety of things.</p>
<p>Grammatical correctness</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p> <p>Naturalness of language</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p> <p>Completeness</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p>

<b>Rating 10</b>
<p>Question:So was -- at the time that the chassis was moved, was that during a precheck inspection and to set it up the initial time?                  Answer: Yes.                  Declarative Sentence: was -- at the time that the chassis was moved , was that during a precheck inspection and to it set up the initial time.</p>
<p>Grammatical correctness</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p> <p>Naturalness of language</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p> <p>Completeness</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p>

<b>Rating 11</b>				
Question: When was the last time that you met with a vocational professional?				
Answer: As to date I can't tell you, but I think it was one day last year. I can't tell you exactly the date.				
Declarative Sentence: that i it with a vocational professional as to date i can't tell I, but i think it was one day last year. i can't tell I exactly the date.				
Grammatical correctness				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5
Naturalness of language				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5
Completeness				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5

<b>Rating 12</b>				
Question: And what -- what did you do for them there?				
Answer: I did carpentry work for one and we did, you know, different home improvements --				
Declarative Sentence: i did do for them there i did carpentry work for one and we did, I know, different home improvements --				
Grammatical correctness				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5
Naturalness of language				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5
Completeness				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5

<b>Rating 13</b>
Question: You inspected the extend wires? Answer: Yes. Declarative Sentence: i inspected the extend wires.
Grammatical correctness <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5  Naturalness of language <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5  Completeness <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5

<b>Rating 14</b>
Question: Are you a certified accident reconstruction expert? Answer: No. Declarative Sentence: i are not a certified accident reconstruction expert.
Grammatical correctness <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5  Naturalness of language <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5  Completeness <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5

<b>Rating 15</b>
Question: And how many other times? Answer: I can think of one that was fairly recent. Declarative Sentence: i can think of one that was fairly recent.
Grammatical correctness <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5  Naturalness of language <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5  Completeness <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5

<b>Rating 16</b>				
Question: is that research and development related to the manufacture of cigarettes?				
Answer: i'm not sure if they do other research. i just don't know very much about those operations.				
Declarative Sentence: i'm not sure if they do other research. i just don't know very much about those operations.				
Grammatical correctness				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5
Naturalness of language				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5
Completeness				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5

<b>Rating 17</b>				
Question: are there records created overseas within philip morris international that relate to the manufacture and sale of cigarettes in the united states?				
Answer: i'm not sure. that's an operating company. i don't on a daily basis manage records internationally.				
Declarative Sentence: i'm not sure. that's an operating company. i don't on a daily basis manage records internationally.				
Grammatical correctness				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5
Naturalness of language				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5
Completeness				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5

<b>Rating 18</b>				
Question: And how would you characterize Mr. Vega?				
Answer: The operator.				
Declarative Sentence: i would characterize mr. vega the operator.				
Grammatical correctness				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5
Naturalness of language				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5
Completeness				
<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5

<b>Rating 19</b>
<p>Question: What were the other materials, the other additional materials, that you've reviewed since completing this report?                  Answer: There were three depositions we identified earlier. His was one. The other -- we named the other two. And I've reviewed the three expert reports in the case.                  Declarative Sentence: i 've reviewed since completing this report there were three depositions we identified earlier. his was one. the other -- we named the other two. and i've reviewed the three expert reports in the case.</p>
<p>Grammatical correctness</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p> <p>Naturalness of language</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p> <p>Completeness</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p>

<b>Rating 20</b>
<p>Question: Do you know if the ANSI 9.25 standard requires that you have a new -- each new user perform a pre-start inspection?                  Answer: I don't recall.                  Declarative Sentence: i don't if I know if the ansi 9.25 standard requires that I have a new -- each new user perform a pre-start inspection.</p>
<p>Grammatical correctness</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p> <p>Naturalness of language</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p> <p>Completeness</p> <p><input type="checkbox"/> 1      <input type="checkbox"/> 2      <input type="checkbox"/> 3      <input type="checkbox"/> 4      <input type="checkbox"/> 5</p>

# Appendix F

## Mturk Final Report

**PROTOCOL TITLE** : Human Evaluation for Transformation of Sentences  
**IRB NUMBER** : 20-143  
**GRANT #** : 460683  
**DATE** : 09/01/2020

### Ground Truth of Declarative Sentences

In this study, we provided each turker **12 QA pairs** to generate their canonical/declarative forms. The turkers were asked to consider each QA pair to be unrelated to another QA pair. This was done to ensure no cross-referencing is done while annotating the QA pairs. In such a manner, we collected a total of **1,836 records** from the turkers. Approximately 70 of the QA pairs were from the Tobacco dataset while the remaining were from anonymized versions of depositions provided by Mayfair. These canonical sentences were used in two different ways: 1) Develop rules and identify issues frequently occurring within the system, 2) As ground truth to evaluate the performance of the rule-based system.

The table below shows the evaluation results against the annotated sentences.

<b>Evaluation Metric</b>	<b>Score</b>
ROUGE 1	0.654
ROUGE 2	0.467
Semantic similarity	0.795
Noun-Verb similarity	0.771

Qualitative evaluation showed the system is able to identify and transform the QA pairs till a readable level, but work still needs to be done on the grammatical side.

# of turkers : 153  
Money Spent : \$349.41  
Avg time per assignment : 11 minutes

## Rating of generated sentences

In this study, we provided each turker with **20 QA pairs** and their machine-generated canonical sentence. We requested them to evaluate and rate the sentences based on the following 3 parameters:

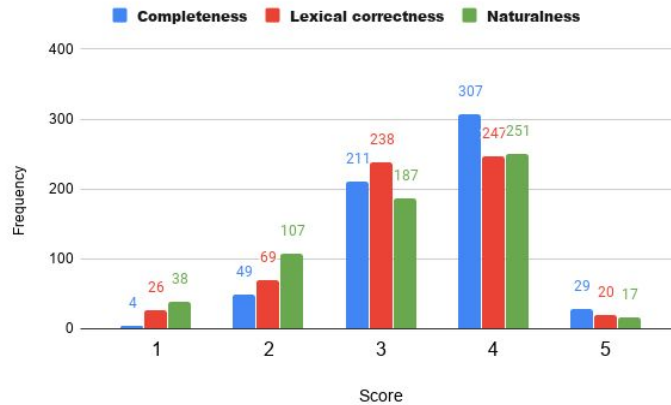
- **Grammatical/Lexical correctness:** Checking if the grammar of the sentence is correct or not.
- **Naturalness:** The sentences generated are natural or readable.
- **Completeness:** Whether the question and answer context is grabbed by the declarative statements.

The turkers were asked to give a rating between 1 and 5 for all the 3 parameters. To ensure that there isn't any bias, we asked **3 different turkers** to rate a sentence.

We handle the scoring in the following manner:

- If more than 2 of the scores are the **same** we take the final score to be that one.
- If all 3 scores are different and at least two of them are **less** than or equal to 3 then we find the average of the scores and then round it down to its integral value.
- If all 3 scores are different and at least two of them are **greater** than 3 then we find the average of the scores and then round it up to its integral value.

We collected a total of **1200 responses** which corresponds to **400 QA pairs**. The following graph shows the compilation of the responses and indicates that the transformation system



performs well. There are some transformed sentences which need improvement grammatically and in the readability aspects. We also observe that many of the transformed sentences are able to identify the important contexts of the question and/or answer.

# of turkers : 60  
Money Spent : \$150.00  
Avg time per assignment : 16 minutes