```
import PyPDF
from googleapiclient.discovery import build
import nltk
from nltk.sentiment.vader import SentimentIntensityAnalyzer

Google PALM API settings
api_key = "YOUR_API_KEY"
cse_id = "YOUR_CSE_ID"

Initialize Google PALM API
def init_palm_api():
    return build("customsearch", "v1", developerKey=api_key)

Extract text from PDF
def extract_text_from_pdf(file_path):
    pdf_file_obj = open(file_path, 'rb')
    pdf_reader = PyPDF2.PdfFileReader(pdf_file_obj)
    num_pages = pdf_reader.numPages
    text = ''
    for page in range(num_pages):
        page_obj = pdf_reader.getPage(page)
        text += page_obj.extractText()
    pdf_file_obj.close()
    return text

Advanced search using PALM
def advanced_search(query, text):
    service = init_palm_api()
    res = service.cse().list(q=query, cx=cse_id).execute()
    results = []
    for result in res['items']:
        results.append(result['title'] + ' ' + result['snippet'])
    return results

Summarize document
def summarize_document(text):
    sentences = nltk.sent_tokenize(text)
    summary = ''
    for sentence in sentences:
        summary += sentence + ' '
    return summary[:500]

Recognize entities
def recognize_entities(text):
    # Use NLTK or spaCy for entity recognition
```

```
    return []

Answer questions
def answer_questions(query, text):
    # Use PALM or other NLP techniques for question answering
    return 'Answer'

Analyze text sentiment
def analyze_sentiment(text):
    sia = SentimentIntensityAnalyzer()
    sentiment = sia.polarity_scores(text)
    return sentiment

Main function
def main():
    pdf_file_path = 'example.pdf'
    text = extract_text_from_pdf(pdf_file_path)

    query = 'Example query'
    results = advanced_search(query, text)
    print('Advanced Search Results:')
    for result in results:
        print(result)

    summary = summarize_document(text)
    print('Document Summary:')
    print(summary)

    entities = recognize_entities(text)
    print('Recognized Entities:')
    for entity in entities:
        print(entity)

    question = 'Example question'
    answer = answer_questions(question, text)
    print('Answer:')
    print(answer)

    sentiment = analyze_sentiment(text)
    print('Sentiment Analysis:')
    print(sentiment)

if __name__ == '__main__':
    main()
```