Data Science and Analytics

Athens, Georgia

2024

4244

Table of Contents

Introduction and Data Overview

In my early years, I did not have formal Bible studies, which made me feel unprepared when people asked about my religious beliefs. But one day, I saw a video discussing character relationships in stories, and it sparked an idea: Why not apply similar methods to the Bible? Seeing the potential to clarify biblical narratives, I decided to take on this project. Using modern computer tools, I carefully organize Bible text data to highlight character connections, focusing on who appears together in the same book.

For visualizing the data, I use Python with Pandas, Matplotlib, and NetworkX. This creates a clear picture of character relationships, making the Bible more engaging. Overall, this project aims to make the Bible easier to understand for both beginners and experts. It is about turning complex data into meaningful insights and sparking more conversations about the Bible's rich characters and their relationships.

Data Dictionary

Node: In the context of network analysis, a node refers to a point of connection or intersection within a network. It can represent various entities such as individuals, objects, or concepts.

Parsing: The process of analyzing or breaking down a text or data into its parts to extract relevant information.

Database: A structured collection of data organized for efficient retrieval, storage, and manipulation.

Infrastructure: The foundational framework or underlying structure supporting a system or operation, such as a database infrastructure.

Visualizations: Representations of data or information in graphical or visual formats, often used to convey insights or patterns more effectively.

Matplotlib: A popular Python library used for creating static, interactive, and animated visualizations.

Pandas: A Python library for data manipulation and analysis, particularly suited for working with structured data such as tables or data frames.

NetworkX: A Python library for creating, manipulating, and analyzing complex networks or graphs.

Corpus: A large and structured collection of texts or data, often used for linguistic or computational analysis.

Python: A versatile programming language used for data analysis and manipulation.

spaCy: A powerful natural language processing library in Python used to identify and extract characters mentioned within the Bible text.

MySQL: A relational database management system used to store and organize the extracted character information.

Bible Text: We utilize the American Standard Version of the Bible as our primary source of text data. This version is chosen for its widely recognized authority and accessibility.

Characters: If the text includes a proper noun, such as a name beginning with a capital letter, it is classified as a character. It's important to note that there may be duplicates of characters due to the program's rudimentary understanding of identifying them. This issue can be rectified in future studies; however, for now, we can work with the existing data and refine our methods as we progress.

Purpose

This project has a big goal: to explore how characters in the Bible are connected. We are using special computer tools to analyze the stories and figure out who knows who and how they are all related. The idea is to make the Bible easier to understand for everyone, not just experts. We want to show how the characters interact and what their relationships can teach us about the stories.

But there might be some challenges along the way. For example, organizing all the information we find could get messy, and sometimes it might be hard to tell if we got things right. To fix this, we will make sure our computer database stays neat and tidy, and we will ask people who know a lot about the Bible to check our work. Due to the big database, we will also have to group ten characters for each graph due to hard to read data. We will also make our project easy to use, so even people who are not computer experts can learn from it.

Overall, we hope this project helps people see the Bible in a new light and understand it better. By uncovering the connections between characters, we can learn more about the stories and what they mean. By making our findings easy to access and understand, we can share our discoveries with as many people as possible, helping them see the Bible's importance and relevance in today's world.

Methods

Our methodology begins with the collection of Bible text from the American Standard Version. We carefully preprocess this text, ensuring its suitability for analysis by removing any extraneous elements and formatting inconsistencies.

With the preprocessed text in hand, we establish a MySQL database to serve as the repository for the extracted character information. This database structure allows for efficient organization and management of the character data, facilitating seamless retrieval and analysis.

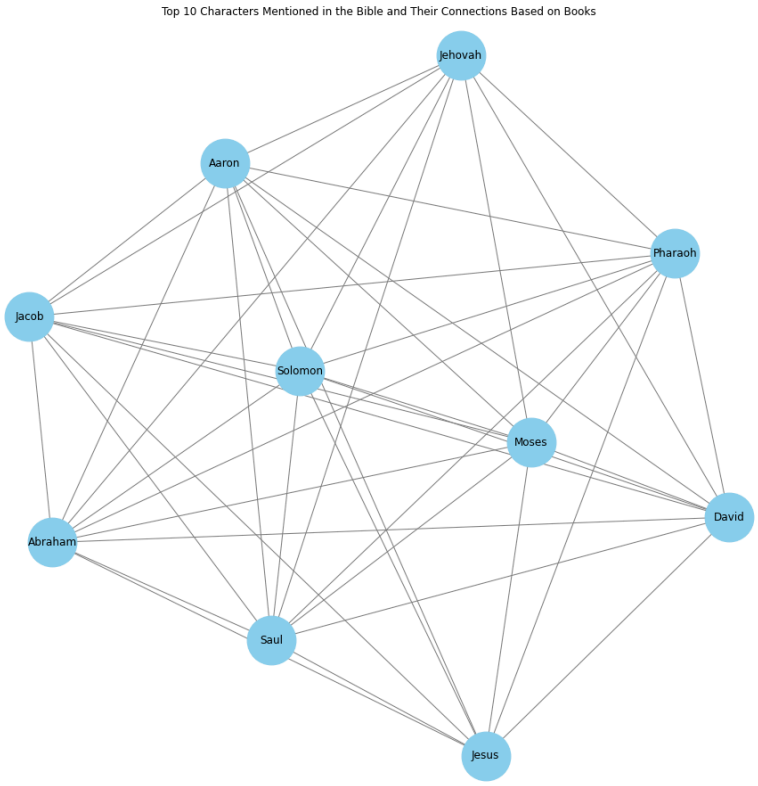
Using spaCy, we initiate character analysis on the preprocessed text by employing its advanced natural language processing capabilities. Specifically, spaCy's functionality allows us to identify proper nouns within the text, which typically correspond to character names. By recognizing these proper nouns, we extract key character mentions from the text. This process involves meticulously cataloging the characters mentioned within the narratives, capturing if characters are mentioned in the same book/chapter/verse. Through spaCy's sophisticated algorithms, we can effectively discern and extract crucial character attributes, laying the groundwork for a comprehensive understanding of the character dynamics within the Bible.

Once the character data is organized and stored in the database, we proceed to visualize the character relationships using Matplotlib and NetworkX. This involves constructing a comprehensive graph representation that illustrates the connections between characters based on their mentions in the same chapter.

Through this methodology, we aim to conduct a thorough analysis and visualization of character relationships in the Bible, providing valuable insights into the narrative structure and connections between characters.

Results

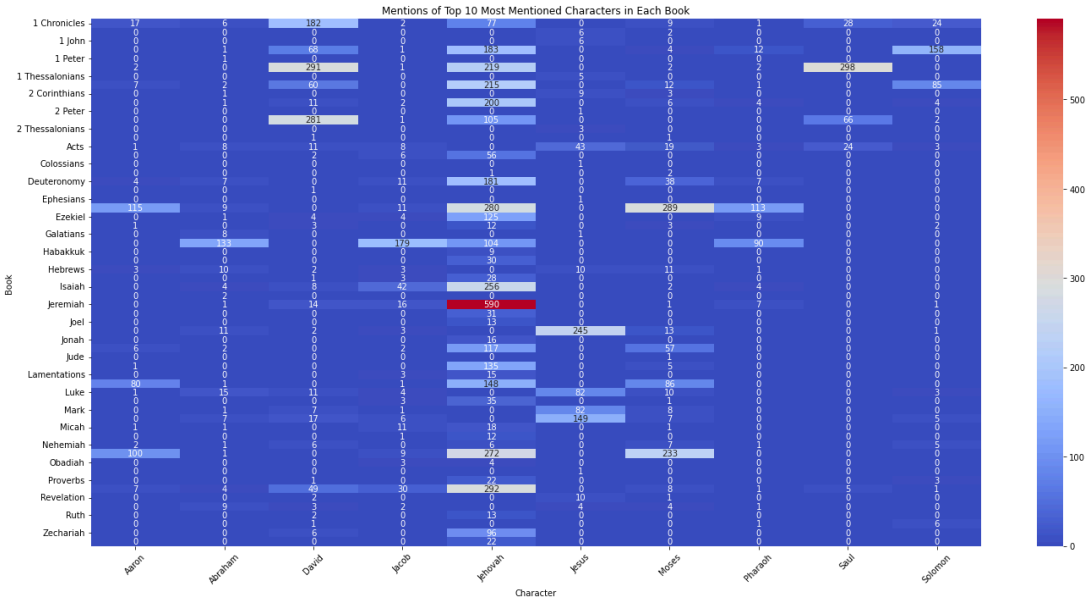
Our first results include a graph based on top ten characters mentioned in the Bible. The reason for only top ten characters mentioned is because there are 2,303 characters based on this dataset. This is a huge amount of characters, and the graph will look odd if we used every single one.



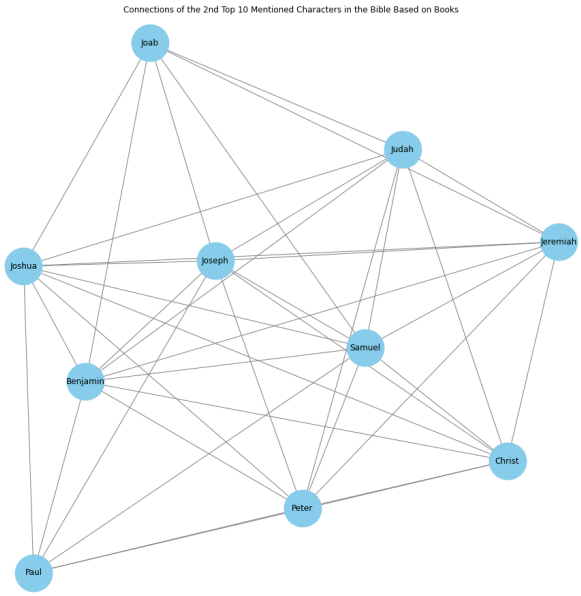
When looking at this graph, it may not surprise many that the top ten characters are all mostly connected in each book. However, if you look closely, the characters named “Jehovah” and “Jesus” are not mentioned in the same book. Further investigating this due to my ignorant knowledge of the Bible, I learned that while both Jehovah and Jesus are integral to Christian theology, they are not mentioned together in the Bible because they represent different aspects of God's relationship with humanity: Jehovah as the divine creator and covenant God of Israel in the Old Testament, and Jesus as the incarnate Son of God and Savior in the New Testament. Thanks to this graph, I was able to easily investigate this a gain further knowledge of what my discovery meant.

Further investigating with these same characters, I wanted to gain insight of the importance of the characters depending on what book they were in. In this heat map, I was able to include different books in the American Standard Version text and conduct how important characters were depending on what book they were mentioned. It is clear that Jehovah was mentioned in the most amount of books, his biggest contribution was

the book ‘Jeremiah.’ This data was very interesting to see how widely spread the characters are throughout the books. Jehovah, Jesus, and David seem to be most mentioned in different books. It is also clear who may have been the main focus of the book depending on how many times they were mentioned in a book.



The last graph I wanted to analyze is the 2nd Top Most mentioned characters based on if they are mentioned in the same book. Similar to the first graph, I believe that this is important to give an insight of character relationships within the Bible. Unlike the first graph however, this gives more characters that are not in the same book. For an example, Jacob and Peter are not mentioned in the same book. This is because Joab is a central figure in the Old Testament while Peter is a central figure in the New Testament. Overall, other characters that are not mentioned together in the same book based on this graph, is most likely because they are mentioned in different parts of history in the Bible.



Conclusions

The analysis of the data regarding the top ten characters mentioned in the Bible reveals an interesting pattern. While most characters are interconnected across various books, it's notable that "Jehovah" and "Jesus" are not mentioned together in any single book. This discrepancy can be attributed to their distinct roles within Christian theology, with Jehovah representing the covenant God of Israel in the Old Testament and Jesus embodying the incarnate Son of God and Savior in the New Testament. Additionally, a heat map analysis of character importance across different books in the American Standard Version text highlights "Jehovah" as the most frequently mentioned character, particularly prominent in the book of Jeremiah. Characters like Jehovah, Jesus, and David appear with significant frequency across various books, indicating their enduring significance in different contexts and narratives. Furthermore, an examination of character relationships through a graph depicting the second most mentioned characters in the Bible unveils insights into their historical contexts and roles. For example, figures like Joab and Peter are not mentioned together in the same book due to their centrality in different historical periods or sections of the Bible—the former in the Old Testament and the latter in the New Testament. Overall, these analyses deepen our understanding of the distribution, significance, and relationships of characters within the biblical narrative, shedding light on theological, historical, and narrative complexities.

Next Steps

Refine Character Identification: Improve the algorithm for identifying characters by incorporating more sophisticated natural language processing techniques. This could involve training the model on a larger corpus of text or fine-tuning the recognition of proper nouns to reduce duplicate entries.

Contextual Analysis: Expand the analysis to include contextual understanding of character mentions. Instead of solely identifying character names, consider extracting additional information such as relationships, roles, and interactions within the text.

Data Cleaning: Conduct thorough data cleaning to address any inconsistencies or errors in the text. This may involve standardizing character names, resolving ambiguities, and removing irrelevant or erroneous entries to ensure the accuracy of the dataset.

Validation and Verification: Implement mechanisms to validate and verify the accuracy of the extracted character data. This could involve manual review by domain experts or comparison with existing annotated datasets to identify discrepancies and refine the analysis accordingly.

Visualization Enhancement: Explore more advanced visualization techniques to present the character relationships in a more intuitive and insightful manner. This could include interactive visualizations, network analysis tools, or incorporation of additional metadata to enrich the visual representation.

Integration of Additional Data Sources: Consider integrating additional data sources, such as character attributes, thematic analysis, or historical context, to provide deeper insights into the characters and their significance within the narrative.

A Bible Character Analysis
4244
Athens, Georgia
2024

Introduction

Lacking formal Bible studies initially, I felt unprepared discussing my beliefs. Inspired by a video on character relationships, I analyzed the Bible with modern tools, visualizing character connections using Python. This project aims to make the Bible more engaging and understandable, sparking insightful conversations about its characters and relationships.

Purpose

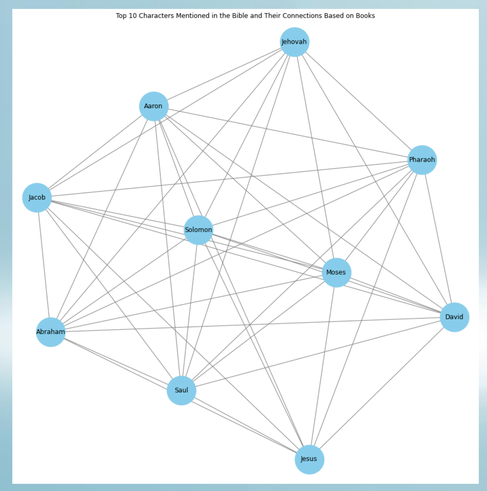
This Bible project uses computer tools to analyze character connections for accuracy and accessibility. Despite data challenges, expert review ensures reliability. Simplified presentation prioritizes user-friendliness, aiming to offer fresh insights into the Bible's relevance today.

Methods

We collect and preprocess Bible text, store character data in a MySQL database, and analyze character mentions using spaCy. Visualizing with Matplotlib and NetworkX, we reveal insights into character relationships and narrative structure in the Bible.

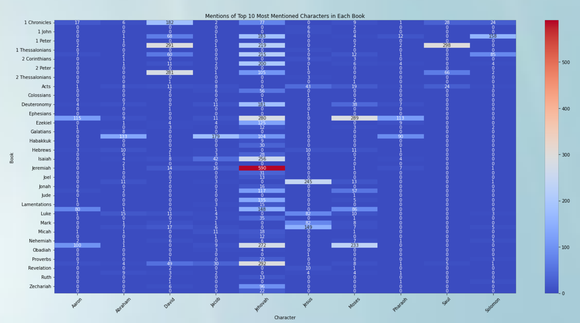
Results

Top Ten Characters Graph:
•Graph based on top ten characters mentioned in the Bible.
•Limitation: Only top ten characters considered due to large dataset (2,303 characters).
•Observation: "Jehovah" and "Jesus" not mentioned together, reflecting distinct aspects of God's relationship with humanity.

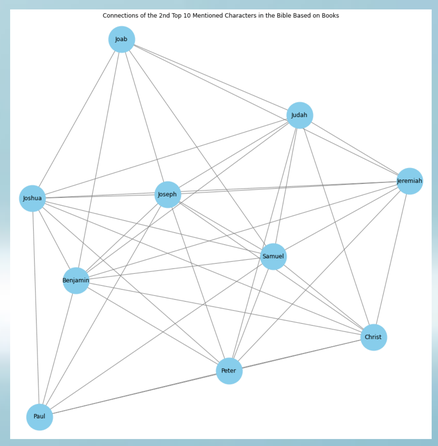


Results

Character Importance Heat Map:
•Heat map showing character importance across different Bible books.
•Insight: "Jehovah" mentioned in most books, with significant presence in "Jeremiah."
•Observation: "Jehovah," "Jesus," and "David" prominently mentioned across various books, indicating potential focal points.



2nd Top Ten Graph:
•Graph analyzing second most mentioned characters based on co-occurrence in the same book.
•Insight: Characters like "Jacob" and "Peter" not mentioned together due to their roles in different historical periods.
•Observation: Graph highlights character relationships across distinct historical contexts within the Bible.



Conclusion

Top Bible characters analyzed: "Jehovah" and "Jesus" never mentioned together, reflecting distinct roles. "Jehovah" prominent, especially in Jeremiah. Consistent appearances of characters like Jehovah, Jesus, and David across books signify enduring significance. Examination reveals historical insights; figures like Joab and Peter not mentioned together due to differing periods.

Next Steps (TPACK)

Refine Character Identification, Contextual Analysis, Data Cleaning, Validation and Verification, Visualization Enhancement, and Integration of Additional Data

References

Works Cited

American Standard Version Text File. https://openbible.com/texts.htm

MatPlotLib. https//matplotlib.org/.

NetworkX. https://networkx.org/.

SpaCy. https://spacy.io/.

Works Cited

American Standard Version Text File. https://openbible.com/texts.htm.

MatPlotLib. https://matplotlib.org/.

NetworkX. https://networkx.org/.

SpaCy. https://spacy.io/.

Appendix

Create Database:

Python

```
create database `bible`;
use `bible`;
```

Create Id, Book, Chapter, Verse, and text

Python

```
use `bible`;

DROP TABLE IF EXISTS `verses`;

CREATE TABLE `verses` (
   `id` bigint not null primary key,
   `book` varchar(255) not null,
   `chapter` int not null,
   `verse` int not null,
   `text` text
);
```

Create Table

Python

```
use `bible`;

DROP TABLE IF EXISTS `characters`;

CREATE TABLE `characters` (
   `id` bigint not null auto_increment primary key,
```

```
   `name` varchar(255) not null,
   `verse_id` bigint not null,
   key `index_verse_id` (`verse_id`)
);
```

Imports and collecting data

Python

```
handler = logging.StreamHandler()
handler.setFormatter(formatter)
logger.addHandler(handler)

file_handler = logging.FileHandler(filename='output.log', mode='a',
encoding='utf')
file_handler.setFormatter(formatter)
logger.addHandler(file_handler)

def connect_to_database(config, attempts=3, delay=2):
   attempt = 1
   while attempt < attempts:
       try:
           return mysql.connector.connect(**config)
       except (mysql.connector.Error, IOError) as error:
           if attempt >= attempts:
               logger.info("Failed to connect, exiting without a connection:
%s" , error)
               return None
           logger.info("Connection failed. %s. Retrying (%d/%d)...", error,
attempt, attempts)
           time.sleep(delay * attempts)
           attempt += 1
```

```
        return None

def parse_line(line):
    line = line.strip()
    line = line.split("\t")
    if len(line) > 1:
        text = line[1]
    else:
        text = ''

    elements = line[0].split(":")
    verse = elements[1]
    chapter = elements[0].split(" ")[-1]
    book = elements[0].split(" ")[0:-1]
    if len(book) > 1:
        book = " ".join(book)
    else:
        book = book[0]
    return {
        "book": book,
        "chapter": chapter,
        "verse": verse,
        "text": text
    }


def add_line_to_db(line, cursor, count):
    data = parse_line(line)
    query = "INSERT INTO `verses`(`id`, `book`, `chapter`, `verse`, `text`)
VALUES (%s, %s, %s, %s, %s)"
    cursor.execute(query, (count, data["book"], data["chapter"], data["verse"],
data["text"]))
    add_character_to_db(data, cursor, count)
```

```
def add_character_to_db(data, cursor, count):
   doc = nlp(data['text'])
   for entity in doc.ents:
       if(entity.label_ == "PERSON"):
           query = "INSERT INTO `characters` (`name`, `verse_id`) VALUES (%s,
%s)"
           cursor.execute(query, (entity.text, count))
           print("Added character", entity.text)


def read_file():
   f = open(file_name, 'r')
   line = f.readline()
   my_cnx = connect_to_database(config, attempts=3, delay=2)
   cursor = my_cnx.cursor()
   count = 1
   while line:
       if len(line) > 0:
           print("Running insert data from verse count: {}" .format(count))
           add_line_to_db(line, cursor, count)
           count = count + 1
       line = f.readline()

   f.close()
   my_cnx.commit()
   cursor.close()
   my_cnx.close()

def main():
   read_file()

main()
```

SQL Query from Dataset:

```
Python
SELECT v.book, v.chapter, c.name, COUNT(c.name) AS name_count FROM
verses v JOIN characters c ON c.verse_id = v.id GROUP BY v.book,
v.chapter, c.name order by v.id, v.book,v.chapter,c.name
```

Graphing Top Ten Characters Relationships from CSV File:

```
Python
import pandas as pd
import matplotlib.pyplot as plt
import networkx as nx

# Load the data
df = pd.read_csv('characters.csv')

# Aggregate to find the top 10 most mentioned characters
character_counts = df.groupby('name')['name_count'].sum().nlargest(10)
top_characters = character_counts.index.tolist()

# Filter dataframe for top 10 characters
filtered_df = df[df['name'].isin(top_characters)]

# Create graph
G = nx.Graph()

# Add edges if characters are mentioned in the same book
for book in filtered_df['book'].unique():
  book_df = filtered_df[filtered_df['book'] == book]
  characters_in_book = book_df['name'].unique()
  for i in range(len(characters_in_book)):
    for j in range(i+1, len(characters_in_book)):
      G.add_edge(characters_in_book[i], characters_in_book[j])

# Draw graph
plt.figure(figsize=(12, 12))
pos = nx.spring_layout(G, seed=42)
nx.draw(G, pos, with_labels=True, node_size=3000, node_color='skyblue',
font_size=12, edge_color='gray')
plt.title('Top 10 Characters Mentioned in the Bible and Their Connections Based
on Books')
plt.axis('off')
```

```
plt.show()
```

Heatmap Graph from Top Ten Characters Relationships from CSV File:

```python
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Load the data
df = pd.read_csv('characters.csv')

# Aggregate to find the top 10 most mentioned characters
character_counts = df.groupby('name')['name_count'].sum().nlargest(10)
top_characters = character_counts.index.tolist()

# Filter dataframe for top 10 characters
filtered_df = df[df['name'].isin(top_characters)]

# Pivot table to show mentions over books
pivot_df = filtered_df.pivot_table(index='book', columns='name',
values='name_count', aggfunc='sum', fill_value=0)

# Plot
plt.figure(figsize=(20, 10))
sns.heatmap(pivot_df, annot=True, cmap='coolwarm', fmt='g')
plt.title('Mentions of Top 10 Most Mentioned Characters in Each Book')
plt.xlabel('Character')
plt.ylabel('Book')
plt.xticks(rotation=45)
plt.yticks(rotation=0)
plt.tight_layout()
plt.show()
```

Graphing 2nd Top Most Mentioned Characters' Relationships using CSV File:

```python
import pandas as pd
import matplotlib.pyplot as plt
```

```
import networkx as nx

# Load the data
df = pd.read_csv('characters.csv')

# Aggregate to find the top 20 most mentioned characters, then select the 2nd
set of 10
character_counts = df.groupby('name')['name_count'].sum().nlargest(20)
second_top_characters = character_counts.index[10:20].tolist()

# Filter dataframe for the 2nd top 10 characters
filtered_df = df[df['name'].isin(second_top_characters)]

# Create graph
G = nx.Graph()

# Add edges if characters are mentioned in the same book
for book in filtered_df['book'].unique():
  book_df = filtered_df[filtered_df['book'] == book]
  characters_in_book = book_df['name'].unique()
  for i in range(len(characters_in_book)):
    for j in range(i+1, len(characters_in_book)):
      G.add_edge(characters_in_book[i], characters_in_book[j])

# Draw graph
plt.figure(figsize=(12, 12))
pos = nx.spring_layout(G, seed=42)
nx.draw(G, pos, with_labels=True, node_size=3000, node_color='skyblue',
font_size=12, edge_color='gray')
plt.title('Connections of the 2nd Top 10 Mentioned Characters in the Bible
Based on Books')
plt.axis('off')
plt.show()
```