Natural Language Processing

Table of Contents

1. Problem Statement **1**

2. Data Exploration **1**

3. Text Summarization **2**

4. TextRank Algorithm **3**

5. Sentence Tokenization **3**

6. Text Preprocessing **4**

7. Similarity Matrix **5**

8. Applying PageRank Algorithm **5**

9. Summary Extraction **5**

1. Problem Statement

With growing digital media and long rails of articles publishing every day - Text summarization comes to rescue for generating concise and meaningful summary of text from multiple text resources such as books, news articles, blog posts, research papers, emails and tweets. The goal of this capstone is to employ Automatic Text Summarization NLP mechanism to provide time saving and useful synopsis to the user on tennis news articles dataset.

2. Data Exploration

There are three features in the sports article's dataset

1. article_id
2. article_text
3. source

The most important feature of these is 'article_text' which contains the text of articles.

- **Input**: article_text

```
# View data in dataset
df['article_text'][1]
```

Example: "Maria Sharapova has basically no friends as tennis players on the WTA Tour. The Russian player has no problems in openly speaking about it auch. I think everyone knows this is my job here. When I'm on the courts or when I'm on the court playing, I'm a competitor and I want to beat every single person

whether they're in the locker room or across the net. So, I'm not the one to strike up a conversation about the weather and know that in the next few minutes I have to go and try to win a tennis match. I'm a pretty competitive girl. I say my hellos, but I'm not sending any players flowers as well. Uhm, I'm not really friendly or close to many players. I have not a lot of friends away from the courts.' When she said she is not really close to a lot of players, is that something strategic that she is doing? Is it different on the men's tour than the women's tour? 'No, not at all. I think just because you're in the same sport doesn't mean that you have to be friends with everyone just because you're categorized, you're a tennis player, so you're going to get along with tennis players. I think every person has different interests. I have friends that have completely different jobs and interests, and I've met them in very different parts of my life. I think everyone just thinks because we're tennis players we should be the greatest of friends. But ultimately tennis is just a very small part of what we do. There are so many other things that we're interested in, that we do."

- **Output**: Summary of the article

3. Text Summarization

Automatic Text Summarization can be broadly classified into two categories – Extractive and Abstractive Summarization.

1. **Extractive Summarization**: Extract select parts of sentences or phrases of the original text to form a summary.
2. **Abstractive Summarization**: Generate abstract summary of the text which may or may not find the same parts as in the original text. These methods use advanced NLP techniques to generate an entirely new summary.

Below picture illustrates an easy understanding of the difference between these two categories:

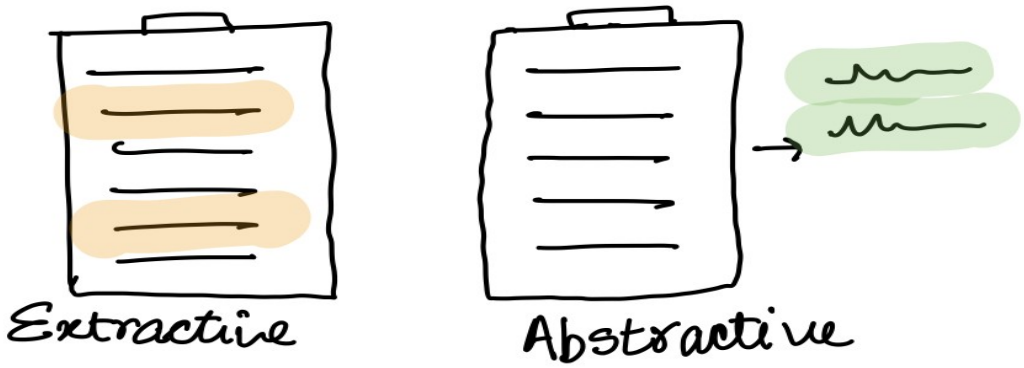

Image source: *medium.com*

4. TextRank Algorithm

TextRank algorithm is inspired based on PageRank algorithm which is primarily used for ranking web pages in online search results. TextRank is an extractive and unsupervised text summarization technique.

Similarities between the two algorithms –

- In place of web pages, we use sentences.
- Similarity between two sentences is used as an equivalent to the page transition probability
- The similarity scores are stored in a square matrix, similar to the matrix M used for PageRank

Below image illustrates the overall process involved in generating the summary for the sports articles

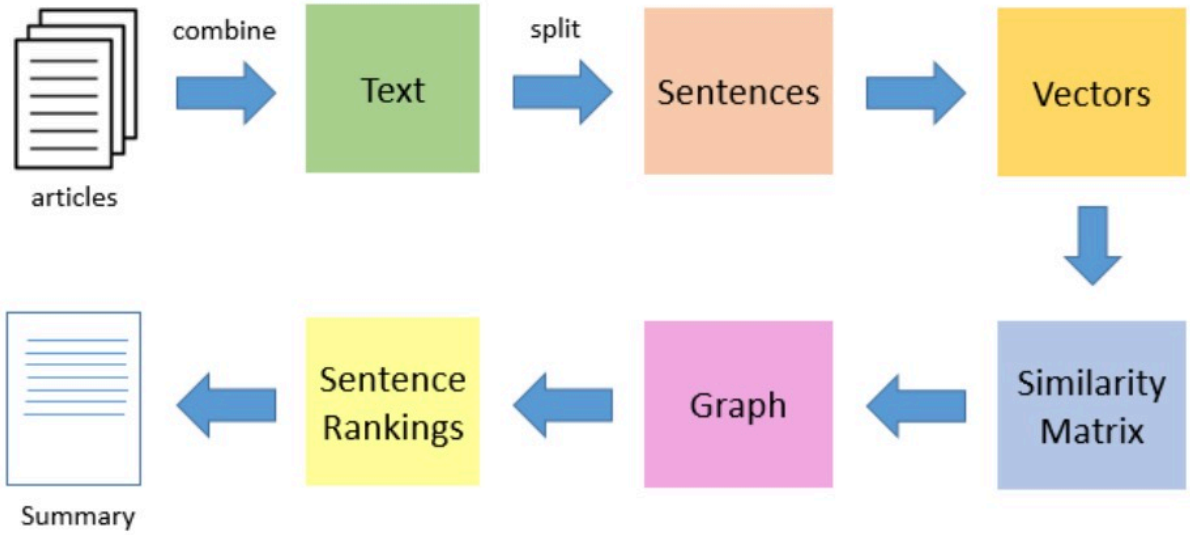

Image Source: *analyticsvidhya.com*

5. Sentence Tokenization

Tokenizing the articles into sentences by leveraging `sent_tokenize` from the NLTK library.

```
# Necessary imports
from nltk.tokenize import sent_tokenize

# Initialize n empty array
sentences = []

# Iterate over each article
for s in df['article_text']:
  sentences.append(sent_tokenize(s))
```

```
# Flatten the list
sentences = [y for x in sentences for y in x]
# Examine the sentences after the split
sentences[:5]
```

Output:
```
['Maria Sharapova has basically no friends as tennis players on the WTA Tour.',
"The Russian player has no problems in openly speaking about it and in a recent interview she said: 'I don't really hide any feelings too much.",
 'I think everyone knows this is my job here.',
 "When I'm on the courts or when I'm on the court playing, I'm a competitor and I want to beat every single person whether they're in the locker room or across the net.
So, I'm not the one to strike up a conversation about the weather and know that in the next few minutes I have to go and try to win a tennis match.",
 "I'm a pretty competitive girl."]
```

6. Text Preprocessing

The following preprocessing steps have been done in order to generate clean sentences on which the algorithm can be applied on.

1. Remove punctuations, numbers and special characters
2. Maintain all lowercase sentences
3. Identify English stop words (as all the articles in the dataset are in English)
4. Remove stop words
5. TF-IDF vector representation of sentences

Below illustrates the abstract code used for performing the preprocessing steps.

```
# Remove punctuations, numbers and special characters
clean_sentences = pd.Series(sentences).str.replace("[^a-zA-Z]", " ")

# Translate alphabets to lowercase
clean_sentences = [s.lower() for s in clean_sentences]

# Set stopwords to English
stop_words = stopwords.words('english')

# Remove stopwords from the sentences
clean_sentences = [remove_stopwords(r.split()) for r in clean_sentences]

# TF-IDF Vector Representation of Sentences
vectorizer = TfidfVectorizer(norm = False, smooth_idf = False)
sentence_vectors = vectorizer.fit_transform(sentences)
```

7. Similarity Matrix

In order to rank the sentences that need to be extracted for summary and also compute a similarity between a pair of sentences we will need to build a similarity matrix. Here, we will be using a cosine similarity matrix available from the sklearn library.

Below illustrates the abstract code used for building the similarity matrix.

```
# Necessary imports
from sklearn.metrics.pairwise import cosine_similarity

# Initialize similarity matrix
sim_mat = np.zeros([len(sentences), len(sentences)])
print(sim_mat)

# Initialize the matrix with cosine similarity scores.
for i in range(len(sentences)):
    for j in range(len(sentences)):
        if i != j:
            sim_mat[i][j] = cosine_similarity(sentence_vectors[i], sente
nce_vectors[j])[0,0]
```

8. Applying PageRank Algorithm

As the PageRank algorithm accepts a graph, we will have to convert the similarity matrix into a graph. The nodes of this graph will represent the sentences and the edges will represent the similarity scores between the sentences.

```
# Convert the similarity matrix sim_mat into a graph
nx_graph = nx.from_numpy_array(sim_mat)

# apply pagerank on the graph above
scores = nx.pagerank(nx_graph)
```

9. Summary Extraction

Extracting the top sentences for summary extraction using a for loop.

Output:
```
When I'm on the courts or when I'm on the court playing, I'm a competitor and I want to
beat every single person whether they're in the locker room or across the net.So I'm not the one
to strike up a conversation about the weather and know that in the next few minutes I have to go
and try to win a tennis match.
Federer is in action at the Swiss Indoors in Basel and if he reaches the final, he could pull out
of Paris in a bid to stay fresh for London.
"Not always, but I really feel like in the mid-2000 years there was a huge shift of the attitudes
of the top players and being more friendly and being more giving, and a lot of that had to do wit
h players like Roger coming up.
Kei Nishikori will try to end his long losing streak in ATP finals and Kevin Anderson will go for
his second title of the year at the Erste Bank Open on Sunday.
```

But as it stands, Federer is in the draw and is scheduled to face either former world No 3 Milos Raonic or Jo-Wilfried Tsonga in the second round.
He used his first break point to close out the first set before going up 3-0 in the second and wr apping up the win on his first match point.
The 20-time Grand Slam winner is chasing his 99th ATP title at the Swiss Indoors this week and he faces Jan-Lennard Struff in the second round on Thursday (6pm BST).
Speaking at the Swiss Indoors tournament where he will play in Sundays final against Romanian qua lifier Marius Copil, the world number three said that given the impossibly short time frame to ma ke a decision, he opted out of any commitment.
And he was delighted to be watched on by all of his family and friends as he purchased 60 tickets for the final for those dearest to him.
The Spaniard broke Anderson twice in the second but didn't get another chance on the South Africa n's serve in the final set.