Лабораторна робота №1

Агрегація ранжирувань отриманих з різних джерел з урахуванням їх важливості

Виконав: Галета М.С.
Група: КМ-91мп

In [1]:
```
1 import numpy as np
2 from search_engines import *
```

In [2]:

```
class SearchQuery:
    def __init__(self, query, n_pages, n_results):
        self.query = query
        self.n_pages = n_pages
        self.n_results = n_results
        self.engines = [Google, Yahoo, Aol, Duckduckgo, Dogpile]
        self.results = None
        self.scores = None

    def custom_search(self):
        look = lambda x: pd.DataFrame()
        self.results = dict(
            zip(
                list(map(lambda engine: engine.__name__, self.engines)),
                list(
                    map(
                        lambda res: {
                            'title': res.titles()[:self.n_results],
                            'link': res.links()[:self.n_results],
                            'description': res.text()[:self.n_results]
                        },
                        list(
                            map(
                                lambda engine: engine().search(self.query, self.n_pages),
                                self.engines
                            )
                        )
                    )
                )
            )
        )

    def show_results(self):
        for engine in self.results.keys():
            print("========================== "+engine+" ==========================\n")
            for i in range(self.n_results):
                print("------------- Result {} -------------".format(i+1))
                print("Title: {}".format(self.results[engine]['title'][i]))
                print("Link: {}".format(self.results[engine]['link'][i]))
                print("Description: {}".format(self.results[engine]['description'][i]))
            print("\n\n")
```

```
42
43      def calculate_scores(self):
44          self.all_links = np.concatenate(np.array(list(map(lambda x: np.array(self.results[x]['link']), self.results.
45          self.unique_links, self.occurences = np.unique(self.all_links, return_counts=True)
46          self.scores = dict(
47              zip(
48                  list(map(lambda engine: engine.__name__, self.engines)),
49                  list(
50                      map(
51                          lambda x: {
52                              'alternatives': self.n_results,
53                              'V': self.n_results/self.all_links.shape[0],
54                              'O': self.n_results/self.unique_links.shape[0],
55                              'E': 1
56                          },
57                          self.results.keys()
58                      )
59                  )
60              )
61          )
```

In [3]:

```
class Ranking:
    def __init__(self, search_results, search_results_scores, all_links, unique_links, occurences):
        self.results = search_results
        self.scores = search_results_scores
        self.all_links = all_links
        self.unique_links = unique_links
        self.occurences = occurences

    def quantity_quality_weights(self):
        ro = np.sum(self.occurences)/(len(self.results.keys())*self.unique_links.shape[0])

        x1 = ro
        x2 = 1 - x1

        weights = np.array(list(map(lambda x: self.scores[x]['E']*(x1*self.scores[x]['O']+x2*self.scores[x]['V']), s
        weights /= np.sum(weights)

        return weights

    def statistical_weights(self):
        D_V = np.var(np.array(list(map(lambda x: self.scores[x]['V'], self.scores.keys()))))

        x1 = D_V
        x2 = 1 - x1

        weights = np.array(list(map(lambda x: self.scores[x]['E']*(x1*self.scores[x]['O']+x2*self.scores[x]['V']), s
        weights /= np.sum(weights)

        return weights

    def borda_ranking(self, weights):
        aggregation_ranks = [{link: i for i, link in enumerate(self.results[engine]['link'], start=1)} for engine in

        for engine_ranks in aggregation_ranks:
            m = len(engine_ranks)
            for link in self.unique_links:
                if link not in engine_ranks:
                    engine_ranks[link] = m + 1

        unique_links_ranks = {}
        for link in self.unique_links:
```

```
            link_rank = 0
            for engine_ranks, weight in zip(aggregation_ranks, weights):
                if link in engine_ranks:
                    link_rank += weight * engine_ranks[link]
            unique_links_ranks[link] = link_rank

        return np.array(sorted(unique_links_ranks.items(), key=lambda x: x[1]))

    def condorcet_ranking(self, weights):
        aggregation_ranks = [{link : i for i, link in enumerate(self.results[engine]['link'], start=1)} for engine :

        matrices = []
        for engine_ranks in aggregation_ranks:
            matrix = np.zeros((self.unique_links.shape[0], self.unique_links.shape[0]))
            for i in range(self.unique_links.shape[0]):
                for j in range(self.unique_links.shape[0]):
                    if i == j:
                        matrix[i, j] = 0
                    elif self.unique_links[i] not in engine_ranks:
                        if self.unique_links[j] not in engine_ranks:
                            matrix[i, j] = 0
                        else:
                            matrix[i, j] = -1
                    elif self.unique_links[j] not in engine_ranks:
                        matrix[i, j] = 1
                    elif engine_ranks[self.unique_links[i]] < engine_ranks[self.unique_links[j]]:
                        matrix[i, j] = 1
                    elif engine_ranks[self.unique_links[i]] > engine_ranks[self.unique_links[j]]:
                        matrix[i, j] = -1
                    elif engine_ranks[self.unique_links[i]] == engine_ranks[self.unique_links[j]]:
                        matrix[i, j] = 0
            matrices.append(matrix)

        ordinal_matrix = np.zeros((self.unique_links.shape[0], self.unique_links.shape[0]))
        for number in range(len(self.results.keys())):
            for i in range(self.unique_links.shape[0]):
                for j in range(self.unique_links.shape[0]):
                    ordinal_matrix[i, j] += (weights[number] * matrices[number][i, j])

        for i in range(self.unique_links.shape[0]):
            for j in range(self.unique_links.shape[0]):
                if ordinal_matrix[i, j] > 0:
```

```
                    ordinal_matrix[i, j] = 1
                elif ordinal_matrix[i, j] < 0:
                    ordinal_matrix[i, j] = -1

        all_ranks = ordinal_matrix.sum(1)

        unique_links_ranks = dict(zip(self.unique_links, all_ranks))

        return np.array(sorted(unique_links_ranks.items(), key=lambda x: x[1], reverse=True))

    def show(self, aggregated_results):
        for i in range(int(len(self.results.keys()) * 10 / 2)):
            for engine in self.results.keys():
                if aggregated_results[i, 0] in self.results[engine]["link"]:
                    source_index = self.results[engine]["link"].index(aggregated_results[i, 0])
                    print("-" * 10 + " Result {} ".format(i + 1) + "-" * 10)
                    print("Title: " + self.results[engine]["title"][source_index])
                    print("Description: " + self.results[engine]["description"][source_index])
                    print("Link: " + self.results[engine]["link"][source_index])
                    print("\n")
                    break
```

In [4]:
```
query = input("Введіть Ваш запит: ")
```

```
Введіть Ваш запит: Imagine Dragons
```

In [5]:
```
sq = SearchQuery(query, 2, 10)
sq.custom_search()
sq.calculate_scores()
results, scores, all_links, unique_links, occurences = sq.results, sq.scores, sq.all_links, sq.unique_links, sq.occu
```

In [6]:
```
1 sq.show_results()
```

```
========================= Google =========================

------------- Result 1 -------------
Title: Imagine Dragons - Believer - YouTubewww.youtube.com › watch
Link: https://www.youtube.com/watch?v=7wtfhZwyrcc&list=PLxtMGPPd_qJPqhvbyIBHiowvbm_MtsW5A&index=33&t=0s (https://www.youtube.com/watch?v=7wtfhZwyrcc&list=PLxtMGPPd_qJPqhvbyIBHiowvbm_MtsW5A&index=33&t=0s)
Description: Enjoy the videos and music you love, upload original content, and share it all with friends, family, and the world ...
------------- Result 2 -------------
Title: Imagine Dragons - Radioactive - YouTubewww.youtube.com › watch
Link: https://www.youtube.com/watch?v=ktvTqknDobU&list=PLlcmBuY0DD0uveef3_MCpJsW8flq4UbgN&index=9&t=0s (https://www.youtube.com/watch?v=ktvTqknDobU&list=PLlcmBuY0DD0uveef3_MCpJsW8flq4UbgN&index=9&t=0s)
Description: Enjoy the videos and music you love, upload original content, and share it all with friends, family, and the world ...
------------- Result 3 -------------
Title: Imagine Dragons — Вікіпедіяuk.wikipedia.org › wiki › Imagine_Dragons
Link: https://uk.wikipedia.org/wiki/Imagine_Dragons (https://uk.wikipedia.org/wiki/Imagine_Dragons)
Description: Imagine Dragons тричі отримували «American Music Award», п'ять разів — «Billboard Music Awards», одну нагороду Греммі та одну «World Music Award».
```

In [7]:
```
1 r = Ranking(results, scores, all_links, unique_links, occurences)
2 qq_weights = r.quantity_quality_weights()
3 s_weights = r.quantity_quality_weights()
```

Модифікований метод Борда. Визначення відносної вагомості джерел інформації на основі кількості і якості наданої інформації

In [8]:
```
1 aggregated_results = r.borda_ranking(qq_weights)
2 r.show(aggregated_results)
```

```
---------- Result 1 ----------
Title: Imagine Dragons | Official Sitewww.imaginedragonsmusic.com
Description: Login · Home · News · Tour · About; Music. Music · Lyrics; Back. Videos · Photos · Store. Social Links.
facebook · instagram · twitter · youtube_vevo · spotify ...
Link: https://www.imaginedragonsmusic.com/ (https://www.imaginedragonsmusic.com/)


---------- Result 2 ----------
Title: Imagine Dragons - Wikipedia
Description: Imagine Dragons were part of the Wayhome summer 2017 lineup in Oro-Medonte, Ontario. On April 27, 2017,
 Imagine Dragons released "Thunder" as the second single from their third album. On May 8, 2017, Imagine Dragons anno
unced their third studio album Evolve, as well as a new track "Whatever It Takes", which was released on the same da
y.
Link: https://en.wikipedia.org/wiki/Imagine_Dragons (https://en.wikipedia.org/wiki/Imagine_Dragons)


---------- Result 3 ----------
Title: ImagineDragons - YouTube
Description: ImagineDragonsVEVO. ImagineDragonsVEVO. Imagine Dragons on Vevo - Official Music Videos, Live Performanc
```

Модифікований метод Борда. Визначення відносної вагомості джерел інформації статистичним підходом

In [9]:
```
1 aggregated_results = r.borda_ranking(s_weights)
2 r.show(aggregated_results)
```

```
---------- Result 1 ----------
Title: Imagine Dragons | Official Sitewww.imaginedragonsmusic.com
Description: Login · Home · News · Tour · About; Music. Music · Lyrics; Back. Videos · Photos · Store. Social Links.
facebook · instagram · twitter · youtube_vevo · spotify ...
Link: https://www.imaginedragonsmusic.com/ (https://www.imaginedragonsmusic.com/)


---------- Result 2 ----------
Title: Imagine Dragons - Wikipedia
Description: Imagine Dragons were part of the Wayhome summer 2017 lineup in Oro-Medonte, Ontario. On April 27, 2017,
 Imagine Dragons released "Thunder" as the second single from their third album. On May 8, 2017, Imagine Dragons anno
unced their third studio album Evolve, as well as a new track "Whatever It Takes", which was released on the same da
y.
Link: https://en.wikipedia.org/wiki/Imagine_Dragons (https://en.wikipedia.org/wiki/Imagine_Dragons)


---------- Result 3 ----------
Title: ImagineDragons - YouTube
Description: ImagineDragonsVEVO. ImagineDragonsVEVO. Imagine Dragons on Vevo - Official Music Videos, Live Performanc
```

Метод Кондорсе. Визначення відносної вагомості джерел інформації на основі кількості і якості наданої інформації

In [10]:
```
1 aggregated_results = r.condorcet_ranking(qq_weights)
2 r.show(aggregated_results)
```

```
---------- Result 1 ----------
Title: Imagine Dragons | Official Sitewww.imaginedragonsmusic.com
Description: Login · Home · News · Tour · About; Music. Music · Lyrics; Back. Videos · Photos · Store. Social Links.
facebook · instagram · twitter · youtube_vevo · spotify ...
Link: https://www.imaginedragonsmusic.com/ (https://www.imaginedragonsmusic.com/)


---------- Result 2 ----------
Title: Imagine Dragons - Wikipedia
Description: Imagine Dragons were part of the Wayhome summer 2017 lineup in Oro-Medonte, Ontario. On April 27, 2017,
 Imagine Dragons released "Thunder" as the second single from their third album. On May 8, 2017, Imagine Dragons anno
unced their third studio album Evolve, as well as a new track "Whatever It Takes", which was released on the same da
y.
Link: https://en.wikipedia.org/wiki/Imagine_Dragons (https://en.wikipedia.org/wiki/Imagine_Dragons)


---------- Result 3 ----------
Title: ImagineDragons - YouTube
Description: ImagineDragonsVEVO. ImagineDragonsVEVO. Imagine Dragons on Vevo - Official Music Videos, Live Performanc
```

Метод Кондорсе. Визначення відносної вагомості джерел інформації статистичним підходом

In [11]:
```
1 aggregated_results = r.condorcet_ranking(s_weights)
2 r.show(aggregated_results)
```

```
---------- Result 1 ----------
Title: Imagine Dragons | Official Sitewww.imaginedragonsmusic.com
Description: Login · Home · News · Tour · About; Music. Music · Lyrics; Back. Videos · Photos · Store. Social Links.
facebook · instagram · twitter · youtube_vevo · spotify ...
Link: https://www.imaginedragonsmusic.com/ (https://www.imaginedragonsmusic.com/)


---------- Result 2 ----------
Title: Imagine Dragons - Wikipedia
Description: Imagine Dragons were part of the Wayhome summer 2017 lineup in Oro-Medonte, Ontario. On April 27, 2017,
 Imagine Dragons released "Thunder" as the second single from their third album. On May 8, 2017, Imagine Dragons anno
unced their third studio album Evolve, as well as a new track "Whatever It Takes", which was released on the same da
y.
Link: https://en.wikipedia.org/wiki/Imagine_Dragons (https://en.wikipedia.org/wiki/Imagine_Dragons)


---------- Result 3 ----------
Title: ImagineDragons - YouTube
Description: ImagineDragonsVEVO. ImagineDragonsVEVO. Imagine Dragons on Vevo - Official Music Videos, Live Performanc
```