INFO 624 – FINAL REPORT

Group 5 Friday 4th, 2020
Daniela Raygadas, dr967@drexel.edu Om Prakash Singh, os338@drexel.edu
Likhil, Rachuri, lkr46@drexel.edu Aishwarya Ravi, ar3646@drexel.edu

Search TED

WHY:

With this global pandemic we are finding ourselves spending more time digitally and being active on OTT platforms. Also, we have seen a shift to online learning and as a part of that online learning people are looking for new ways to stay motivated and learn. TED Talks are a great way to learn from speakers' experts on education, business, science, tech, and many more. We decided to build this TED talk search engine for people to find videos based on their interests. Hopefully, this will simplify their search by giving them different options to search based on what they are looking for.

WHAT:

.
We used a Kaggle Dataset that contained information of all audio-video recordings of TED Talks uploaded to the official TED website up to September 21st, 2017. In total we had 2550 TED Talks, and the data was made available to us in a csv file.

The fields available for each record were:

- comments
- description
- duration
- event
- film_date
- languages
- main_speaker
- name
- num_speaker
- published_date
- ratings
- related_talks
- speaker_occupation
- tags
- title
- url
- views

From these fields we decided to just keep the ones that were meaningful to our search engine and decided to remove languages, name, and related talks. The languages field had the number of languages in which the talk was available, the name field was repetitive as it combined the title and main speaker into one field, and the related talks wasn't related to the search engine as this field had a list of recommended talks to watch next.

This is the link to our dataset https://www.kaggle.com/rounakbanik/ted-talks

WHO:

For our use cases we focused on the three main ways a user will search for a TED Talk which can be by searching on a topic of interest, by searching for a specific speaker and lastly by searching based on the expertise of the speaker on a subject of interest.

These are the three use cases we focused on:

1. For our first use case, the user is interested in finding TED Talks related to climate change and global warming. The user wants to learn more and stay up to date on these topics.

2. For the second use case, the user already is looking for a specific speaker, Bill Gates, who is a major influential in our world today. The user is interested in knowing all the talks that this speaker has done.

3. For the third use case, the user wants to find TED Talks where the speaker is knowledgeable in architecture, and the speakers talks about current trends in architecture as building sustainable and environmental structures. As the user is looking to learn, the user might want to see the videos which are most informative and not the ones that are funny.

HOW:

To build our search engine, we decided to use the Kibana interface. Our plan was to first create one index where we would have the default BM25 similarity on the text fields, and then create a second index where we would create our own custom similarity and apply it to our text fields which are Description and Title. For the main speaker field, we will change the similarity to Boolean as we are interested to see if the main speaker matches or not. We will talk more about our custom similarity in the next steps.

1. As mentioned before, we didn't keep all the fields provided on Kaggle's dataset. Below we explain how we handled each data field in both of our indexes.
   a. **Comments:** The data type as integer and we decided to have index as false as we didn't want this field to be searchable. This fields contains the number of first level comments made on the talk.
   b. **Description:** The data type as text and we decided to use the English analyzer as we need would need to apply stemming. This field contains information about what was talk discuss in the talk.
   For the first index we used the default BM25 similarity and for the second one we created our own similarity.
   c. **Duration:** The data type as integer, and we decided to keep it searchable. Even though, we don't use in our use cases. This field contains the duration of the talk in seconds.
   d. **Event:** The data type as keyword as we want this field to be used for filtering and only be searchable by the exact value. The event of the talk doesn't have a lot of variations and is not something that would need to be analyzed as text. This field contains the TED/TEDx event where the talk took place
   e. **film_date:** The data types as date, and this field contains the Unix timestamp of the filming.
   f. **main_speaker:** The data type as text, and the analyzer as standard as we don't want to do any stemming on the names of the speakers. We want to keep the names as it is. For the first index we used the default BM25 similarity, but for the second index we used the Boolean similarity because we are interested in seeing if the speaker matches or not. We don't want partial results as we are looking for a specific speaker.
   g. **num_speaker:** The data type as integer, and index as false because we don't want to have this field as searchable. This fields contains the number of speakers in the talk
   h. **published_date:** The data types as date, and this field contains the Unix timestamp for the publication of the talk on TED.com
   i. **ratings:** The data type as rank features because we wanted to be able to ranked our results based on a rating of interest. People are not always looking for the same type of videos, the videos might be talking about the same topic but the mood is different. We can have videos talking about the same thing but one of them is on a Funny way and the other one on a more Educational way.
   j. **speaker_occupation:** The data type as keyword as we want this field to be used for filtering purposes. There is not a lot of variation when it comes to a speaker occupation.
   k. **tags:** The data type as keyword as we want this field to be used for filtering, and in the official TED website they have these tags as a dropdown list that you can select from. Hence, there is no variations on the values for this field.
   l. **Title:** The data type as text and the analyzer as English as we want to perform stemming. The title of the talk will also contain information of what the talk was about. Also, just like

with the description field for the first index we used the default BM25 similarity and for the second one we created our own similarity.

m. URL: The data type as text, but index as false as we don't want this field to be searchable. We kept this field for informative purposes about the talk.

n. views: The data type as rank feature because we want to be able to rank our results based on the number of views, and this will enable us to have the videos with the highest number of views at the top.

2. Next, we are going to describe the search queries we created for each use case. We then used these same queries to test and evaluate both of our indexes.

a. User searching for TED Talks on climate change and global warming.

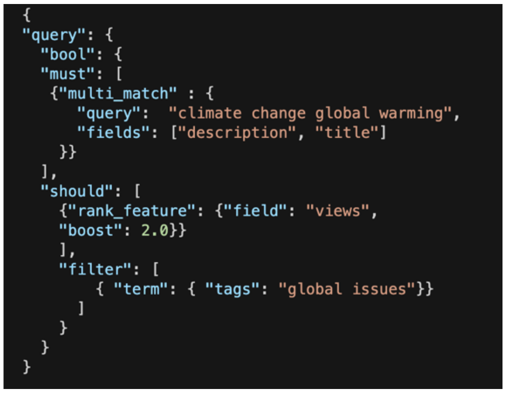
i. The fields that should be search for potential matches are description/title as these fields contain information related to what was discussed on the TED Talk and what the TED Talk was about.

ii. The field that should be included in the scoring is the views field, as the user wants the most popular ones at the top of the results.

iii. The user can also filter the results using the tag field in order to get more specific on what about climate change and global warming as cars, engineering, science and many other.

iv. Below is an image showing the keywords and query structure that should be used for this information need.

We can observe that the keywords we would need to use are climate change global warming and we would perform a multi match query on the description and title fields. Then we would ranked the results retrieved by the numbers of views and boost that score as we want to add more weight or importance to this field to the overall score as the TED Talks that have the highest number of views should come at the top. Finally, we would filter by the tags which are the themes associated with the talk to make our search more specific.



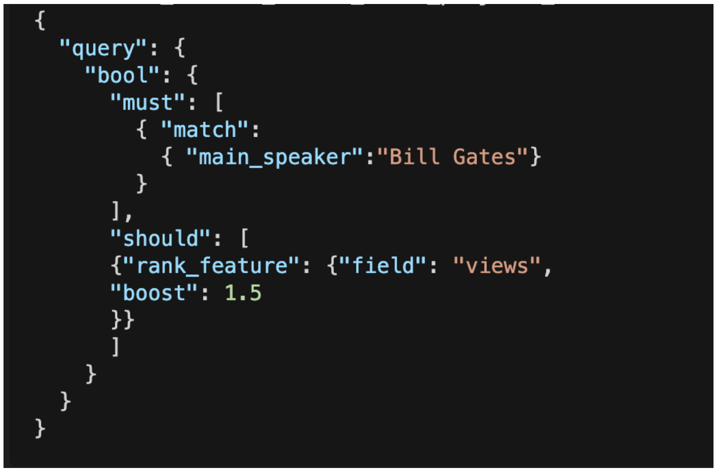
```
{
"query": {
  "bool": {
  "must": [
   {"multi_match" : {
      "query":  "climate change global warming",
      "fields": ["description", "title"]
    }}
  ],
  "should": [
    {"rank_feature": {"field": "views",
    "boost": 2.0}}
    ],
    "filter": [
        { "term": { "tags": "global issues"}}
      ]
    }
  }
}
```

b. User searching for TED Talks done by Bill Gates

i. The fields that should be search for potential matches are main_speaker.

ii. The field that should be included in the scoring is the views field, as the user wants to see the most popular ones at the top of the results.

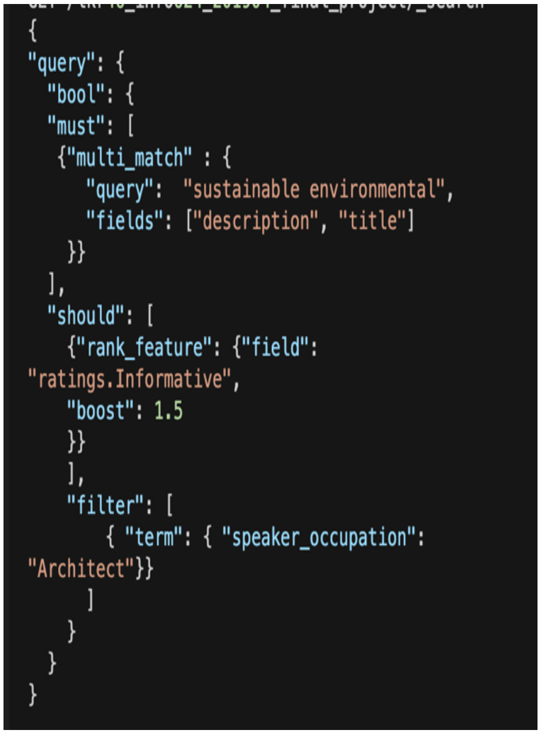
iii. Below is an image showing the keywords and query structure that should be used for this information need.
We can observe that the keywords we would need are Bill Gates as this is the name of the speaker, we are interested in. We would perform a match on the main speaker field and ranked the results retrieved by the number of views as we want the most popular ones at the top. Again, we are boosting the views field to add more weight or importance to this field to the overall score as the TED Talks that have the highest number of views should be at the top.



```
{
  "query": {
    "bool": {
      "must": [
        { "match":
          { "main_speaker":"Bill Gates"}
        }
      ],
      "should": [
      {"rank_feature": {"field": "views",
      "boost": 1.5
      }}
      ]
    }
  }
}
```

c. User searching for a speaker that is knowledgeable in architecture.
   i. The fields that should be search for potential matches are speaker occupation and description/title
   ii. The field that should be included in the scoring is the ratings field, as the user will want to get the video with most rating based on how informative the video was for other people.
   iii. Below is an image showing the keywords and query structure that should be used for this information need.

We can observe that the keywords we would need are sustainable environmental as we are searching for TED Talks that discuss about building sustainable and environmental structures. We are filtering our results by the speaker occupation, Architect, as we are interested in experts on the subject. Finally, we are ranking the results retrieved by the ratings fields. Notice that for this we are giving to the query the rating we are interested in, which is Informative. We are boosting the ratings field to add more weight or importance to this field to the overall score as we want to make sure the TED Talks with the highest informative ratings appearing at the top.



```
GET /lkr46_info624_201904_final_project/_search
{
"query": {
  "bool": {
  "must": [
   {"multi_match" : {
      "query":  "sustainable environmental",
      "fields": ["description", "title"]
    }}
  ],
  "should": [
    {"rank_feature": {"field":
"ratings.Informative",
    "boost": 1.5
    }}
    ],
    "filter": [
        { "term": { "speaker_occupation":
"Architect"}}
      ]
    }
  }
}
```

3. As mentioned previously, on the first index we decided to use the default BM25 similarity on our text fields. But on the second index we created our own default BM25 to see if that would improve our results based on the information needs. On the next step we evaluate both mappings and analyzed which one was better at retrieving the TED Talks based on the information need of our use cases.

   For the scoring, we decided to use ratings and views as both could be use in a rank feature query. We wanted to experiment and give the user different ways in which it could ranked the results retrieved by the query based on the information need. If the user was looking for the most popular videos, or the user is more interested in finding videos which are informative rather than funny.

4. Index creation was done using Kibana interface in elastic search, all the required mapping settings were also done using Kibana.
   Once the mapping is created, we have further used the created index and called it from Jupyter using python notebook where we have loaded the data.
   Before data load few steps of pre-processing is also done to change extract the data into required format and structure.

   Mapping without custom similarity: Index - **lkr46_info624_201904_final_project**

```
{
 "lkr46_info624_201904_final_project" : {
   "mappings" : {
```

```
  "properties" : {
    "comments" : {
      "type" : "integer",
      "index" : false
    },
    "description" : {
      "type" : "text",
      "analyzer" : "english"
    },
    "duration" : {
      "type" : "integer"
    },
    "event" : {
      "type" : "keyword"
    },
    "film_date" : {
      "type" : "date",
      "format" : "epoch_millis"
    },
    "main_speaker" : {
      "type" : "text",
      "analyzer" : "standard"
    },
    "num_speaker" : {
      "type" : "integer",
      "index" : false
    },
    "published_date" : {
      "type" : "date",
      "format" : "epoch_millis"
    },
    "ratings" : {
      "type" : "rank_features"
    },
    "speaker_occupation" : {
      "type" : "keyword"
    },
    "tags" : {
      "type" : "keyword"
    },
    "title" : {
      "type" : "text",
      "analyzer" : "english"
    },
    "url" : {
      "type" : "text",
      "index" : false
    },
    "views" : {
      "type" : "rank_feature"
    }
  }
}
```

```
 }
}
```

Mapping with custom similarity: index- **lkr46_info624_201904_ted_search:**

Data into this index is taken from the above created index without similarity using re-index function from elastic search, which reduced the time for data to load into this index.

```
{
 "settings": {
  "index": {
   "similarity": {
    "my_bm25": {
     "type": "BM25",
     "k1": 2.5,
     "b":0.9
    },
    "my_dfr": {
     "type": "DFR",
     "basic_model": "g",
     "after_effect": "l",
     "normalization": "h2",
     "normalization.h2.c": "3.0"
    }
   }
  }
 },

 "mappings" : {
   "properties" : {
    "comments" : {
     "type" : "integer",
     "index" : false
    },
    "description" : {
     "type" : "text",
     "analyzer" : "english",
     "similarity" : "my_bm25"
    },
    "duration" : {
     "type" : "integer"
    },
    "event" : {
     "type" : "keyword"
    },
    "film_date" : {
     "type" : "date",
     "format" : "yyyy-MM-dd"
    },
    "main_speaker" : {
     "type" : "text",
     "analyzer" : "standard",
          "similarity" : "boolean"
        },
        "num_speaker" : {
          "type" : "integer",
          "index" : false
        },
        "published_date" : {
          "type" : "date",
          "format" : "yyyy-MM-dd"
        },
        "ratings" : {
          "type" : "rank_features"
        },
        "speaker_occupation" : {
          "type" : "keyword"
        },
        "tags" : {
          "type" : "keyword"
        },
        "title" : {
          "type" : "text",
          "analyzer" : "english",
          "similarity" : "my_dfr"
        },
        "url" : {
          "type" : "text",
          "index" : false
        },
        "views" : {
          "type" : "rank_feature"
        }
      }
    }
}
```

Similarity

Elasticsearch allows us to configure a scoring algorithm or similarity per field. The similarity setting provides a simple way of choosing a similarity algorithm other than the default BM25, such as TF/IDF. Similarities are mostly useful for text fields, but can also apply to other field types. Custom similarities can be configured by tuning the parameters of the built-in similarities. We have used 3 types of similarities for our analysis:

- BM25 Similarity
- DFR similarity
- Boolean

Impact after using similarity:

In here we are creating new similarity options with the built-in BM25 and DFR modules that we will use in some of our fields. This way we can customize our index and have each field use a different similarity based on our information needs.

1. For the custom BM25, my_bm25, the impact it will have is that by increasing k1 from the default 1.2 to 3 we will slow down term saturation which means we would pivot at a higher value.
2. By increasing b from the default 0.75 to 1.0 we are making sure we fully normalize the document length. This means that the longer documents will be penalized, and they won't have an advantage to the shorter documents as longer docuemnts tend to have more words and a higher term frequency.
3. For the custom DFR, my_dfr, we are creating a Divergence from Randomness similarity. DFR is a probabilistic model, and it weights terms using probabilitic methods. We can decide which probabilitic model to use to weight this term with the basic_model parameter. For our custom DFR we are deciding to use the geometric approximation of Bose-Einstein.In this basic_model term weights will be computed by measuring the divergence between a term distribution produced by a random process and the actual term distribution.

Impact on our fields:

1. **My_bm25 -** The difference it will make is that the description field will be using the custom BM25 we created for the similarity. By using our custom BM25, my_bm25, instead of the default one in the abstract field the difference it will make is that it will result in a slower term saturation allowing the score for each term to go up the more instances of that term, and it will penalize longer description.
2. **My_dfr -** The difference it will make is that the title field will use the custom DFR similarity we created instead of the default BM25. In this case we will be weighting terms based on the probabilistic model we selected, and by using normalization as h2 we are making sure we also penalize longer titles. The normalization parameter refers to the term frequency normalization, which states that the term frequency is inversely related to the length.
3. **Boolean -** The difference it will have is that the main_speaker field will use a boolean similarity instead of the default BM25 similarity. The standard analyzer doesn't use stemming which is good as we don't want to use stemming on the speaker's name.

HOW GOOD:

To evaluate our indexes, we used our three use case queries and decided to evaluate the performance of them by using precision and DCG as our evaluation metrics.

Without similarity measures
Use case 1.

Request:

```
GET /lkr46_info624_201904_final_project/_search
{
 "from": 0,"size": 20,
 "query": {
   "bool":{
     "must":[
       {
         "multi_match" : {
         "query": "climate change global warming",
         "fields": ["description", "title"]
       }}
       ],
       "should": [
         { "rank_feature": {"field": "views","boost":15.0}
         }],
         "filter": [
           {"term": {"tags": "global issues"}}]
   }
 }
}
```

Result table 1

Doc ID	Main Speaker	title	score	Relevance
1	Al Gore	Averting the climate crisis	27.359032	1
1166	James Hansen	Why I must speak out about climate change	20.79972	1
2181	Hugh Evans	What does it mean to be a citizen of the world?	18.206242	0
2195	Michael Metcalfe	A provocative way to finance the fight against climate change	17.164629	1
491	Gordon Brown	Wiring a web for global good	16.909958	0
214	Al Gore	New thinking on the climate crisis	16.861343	1
1825	Lord Nicholas Stern	The state of the climate — and what we might do about it	16.463585	1
25	David Deutsch	Chemical scum that dream of distant quasars	16.457846	1
2063	Michael Green	How we can make the world a better place by 2030	16.338728	0
2088	Anote Tong	My country will be underwater soon -	16.330212	1

		- unless we work together		
2051	Mary Robinson	Why climate change is a threat to human rights	15.325451	1
618	Bill Gates	Innovating to zero!	15.202998	0
726	Hans Rosling	Global population growth, box by box	15.163556	1
1453	Dan Pallotta	The way we think about charity is dead wrong	14.941391	0
2186	Christiana Figueres	The inside story of the Paris climate agreement	14.911657	1
2058	Alice Bows-Larkin	Climate change is happening. Here's how we adapt	14.803745	1
2333	Joe Lassiter	We need nuclear power to solve climate change	14.509262	1
1779	Simon Anholt	Which country does the most good for the world?	14.460508	1
160	David Keith	A critical look at geoengineering against climate change	14.402176	1
840	Lesley Hazleton	On reading the Koran	14.316608	0

Evaluation

Document relevant to query	IR system retrieve
1,1166,2195,214,1825,25,2088,2051,726,2186,2058,2333,1779,160	1, 1166,2181,2195,491,2141,1825,25,2063,2088,2051,618,726,1453,2186,2058,2333,1779,160,840

	Relevant	Non-relevant
Retrieved	TP =14	FP = 6

Precision: tp/(tp + fp) = 14/ (14+6) = 0.7

Compute DCG (Discounted Cumulative Gain)

$$DCG_p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2 i}$$

We are focusing on top 3 doc; therefore, we will only be considered doc till position 3(rank 3).

DCG3 = rel1 + rel2/log2 + rel3/log3
= 1 + 1/1 + 0/log3
= **2**

Use Case 2:

Request:

```
GET /lkr46_info624_201904_final_project/_search
{
  "from": 0,"size": 20,
  "query": {
    "bool": {
      "must" : [
        {
          "match": {
            "main_speaker": "Bill Gates"
          }
        }],
        "should": [
          {"rank_feature": {"field": "views","boost": 15.0}}
        ]
    }
  }
}
```

Result table 2:

Doc ID	Main Speaker	title	score	relevance
618	Bill Gates	Innovating to zero!	22.752645	1
380	Bill Gates	Mosquitos, malaria and education	21.331318	1
1950	Bill Gates	The next outbreak? We're not ready	20.79922	1

1496	Bill Gates	Teachers need real feedback	20.749268	1
882	Bill Gates	How state budgets are breaking US schools	20.167202	1
1714	Bill and Melinda Gates	Why giving away our wealth has been the most satisfying thing we've done	18.78536	1
1987	Bill Gross	The single biggest reason why startups succeed	16.673046	0
118	Bill Stone	I'm going to the moon. Who's with me?	14.289991	0
787	Melinda Gates	What nonprofits can learn from Coca-Cola	13.32804	0
1202	Melinda Gates	Let's put birth control back on the agenda	13.169572	0
1944	Theaster Gates	How to revive a neighborhood: with imagination, beauty and art	12.772115	0
1971	Bill T. Jones	The dancer, the singer, the cellist ... and a moment of creative magic	12.111556	0
63	Bill Clinton	My wish: Rebuilding Rwanda	11.505936	0
964	Bill Ford	A future beyond traffic gridlock	11.288578	0
602	Bill Davenhall	Your health depends on where you live	11.108625	0
379	Bill Gross	A solar energy system that tracks the sun	10.355118	0
180	Bill Strickland	Rebuilding a neighborhood with beauty, dignity, hope	10.343769	0
1128	Bill Doyle	Treating cancer with electric fields	9.990719	0
342	Bill Joy	What I'm worried about, what I'm excited about	9.604826	0

Evaluation

Document relevant to query	IR system retrive
618,380,1950,1496,882,1714	618,380,1950,1496,882,1714,1987,118,787,1202,1944,1971,63,964,602,379,180,1128,342

	Relevant	Non-relevant
Retrieved	TP=6	FP = 13

Precision: tp/(tp + fp) = 6/ (6+13) = 0.31578

Compute DCG (Discounted Cumulative Gain)

$$DCG_p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2 i}$$

We are focusing on top 3 doc; therefore, we will only be considered doc till position 3(rank 3).

DCG3 = rel1 + rel2/log2 + rel3/log3
= 1 + 1/1 + 1/1.6
= **2.625**

Use case 3.

```
GET /lkr46_info624_201904_final_project/_search
{
  "from": 0,"size": 20,
  "query": {
    "bool": {
      "must":[
        {
          "multi_match": {
            "query": "sustainable environmental",
            "fields": ["description","title"]
          }
        }
        ],
        "should": [
          {"rank_feature": {"field": "ratings.Informative" ,"boost":15.0}}
        ],
        "filter": [
          {"term": {"speaker_occupation": "Architect"}}]
    }
  }
}
```

Result table 3.

Doc ID	Main Speaker	title	speaker_occupation	score	relevance
867	Michael Pawlyn	Using nature's genius in architecture	Architect	15.573233	1
1125	Bjarke Ingels	Hedonistic sustainability	Architect	12.824576	1
722	Mitchell Joachim	Don't build your home, grow it!	"Architect",”designer"	10.891966	1
1567	Shigeru Ban	Emergency shelters made from paper	Architect	7.7443104	1

Evaluation

Document relevant to query	IR system retrive
867,1125,722,1567	867,1125,722,1567

	Relevant	Non-relevant
Retrieved	TP =4	FP =0

Precision: tp/(tp + fp) = 4/ (4+0) = 1.0
Compute DCG (Discounted Cumulative Gain)

$$DCG_p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2 i}$$

We are focusing on top 3 doc; therefore, we will only be considered doc till position 3(rank 3).

DCG3 = rel1 + rel2/log2 + rel3/log3
= 1 + 1/1 + 1/1.6
= **2.625**

With Similarity

Use case 1:
Request:

```
            GET / lkr46_info624_201904_ted_search /_search
{
 "from": 0,"size": 20,
  "query": {
    "bool": {
      "must": [
        {
          "multi_match" : {
          "query": "climate change global warming",
          "fields": ["description", "title"]
        }}
        ],
        "should": [
          { "rank_feature": {"field": "views","boost":15.0}
          }],
          "filter": [
            {"term": {"tags": "global issues"}}]
    }
  }
}
```

Result Table 4.

Doc ID	Speaker	Title	Score	Relevance
1	Al Gore	Averting the climate crisis	28.35261	1
1166	James Hansen	Why I must speak out about climate change	22.76149	1
2181	Hugh Evans	What does it mean to be a citizen of the world?	18.44517	0
214	Al Gore	New thinking on the climate crisis	17.79548	1
491	Gordon Brown	Wiring a web for global good	17.5626	1
2088	Anote Tong	My country will be underwater soon -- unless we work together	17.24678	1
25	David Deutsch	Chemical scum that dream of distant quasars	17.09587	1
2195	Michael Metcalfe	A provocative way to finance the fight against climate change	16.9399	1
1825	Lord Nicholas Stern	The state of the climate — and what we might do about it	16.70658	1

2063	Michael Green	How we can make the world a better place by 2030	15.91601	1
2051	Mary Robinson	Why climate change is a threat to human rights	15.90358	1
160	David Keith	A critical look at geoengineering against climate change	15.43849	1
2186	Christiana Figueres	The inside story of the Paris climate agreement	15.38617	1
618	Bill Gates	Innovating to zero!	15.35287	0
2058	Alice Bows-Larkin	Climate change is happening. Here's how we adapt	15.30638	1
1453	Dan Pallotta	The way we think about charity is dead wrong	15.07067	0
726	Hans Rosling	Global population growth, box by box	15.04441	1
2333	Joe Lassiter	We need nuclear power to solve climate change	15.01636	1
509	James Balog	Time-lapse proof of extreme ice loss	14.64819	1
1345	Vicki Arroyo	Let's prepare for our new climate	14.33191	1

Evaluation

Document relevant to query	IR system retrieve
1,1166,214,491,2088,25,2195,1825,2063,2051,160,2186,2058,726,2333,509,1345	1,1166,2181,214,491,2088,25,2195,1825,2063,2051,160,2186,618,2058,1453,726,2333,509,1345

	Relevant	Non-relevant
Retrieved	TP = 17	FP = 3

Precision: tp/(tp + fp) = 17/ (17+3) = 17/20 = 0.85
Compute DCG (Discounted Cumulative Gain)

$$DCG_p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2 i}$$

We are focusing on top 3 doc; therefore, we will only be considered doc till position 3(rank 3).

DCG3 = rel1 + rel2/log2 + rel3/log3
= 1 + 1/1 + 0/log3
= **2**

Use Case 2:

Request:

```
GET / lkr46_info624_201904_ted_search /_search
{
 "from": 0,"size": 20,
 "query": {
   "bool": {
     "must" : [
       {
         "match": {
           "main_speaker": "Bill Gates"
         }
       }],
       "should": [
         {"rank_feature": {"field": "views","boost": 15.0}}
       ]
   }
 }
}
```

Result Table 5.

Doc ID	Speaker	Title	Score	Relevance
618	Bill Gates	Innovating to zero!	13.8209	1
1714	Bill and Melinda Gates	Why giving away our wealth has been the most satisfying thing we've done	12.93511	1
1987	Bill Gross	The single biggest reason why startups succeed	12.49094	0
380	Bill Gates	Mosquitos, malaria and education	12.39957	1
1950	Bill Gates	The next outbreak? We're not ready	11.86747	1
1496	Bill Gates	Teachers need real feedback	11.81752	1
882	Bill Gates	How state budgets are breaking US schools	11.23545	1
118	Bill Stone	I'm going to the moon. Who's with me?	10.10788	0
1971	Bill T. Jones	The dancer, the singer, the cellist ... and a moment of creative magic	8.779661	0
787	Melinda Gates	What nonprofits can learn from Coca-Cola	8.578398	0
1202	Melinda Gates	Let's put birth control back on the agenda	8.41993	0
1944	Theaster Gates	How to revive a neighborhood: with imagination, beauty and art	8.022472	0
63	Bill Clinton	My wish: Rebuilding Rwanda	7.323829	0

964	Bill Ford	A future beyond traffic gridlock	7.106472	0
602	Bill Davenhall	Your health depends on where you live	6.926518	0
379	Bill Gross	A solar energy system that tracks the sun	6.17301	0
180	Bill Strickland	Rebuilding a neighborhood with beauty, dignity, hope	6.161663	0
1128	Bill Doyle	Treating cancer with electric fields	5.808612	0
342	Bill Joy	What I'm worried about, what I'm excited about	5.422719	0

Evaluation

Document relevant to query	IR system retrieve
618,1714,380,1950,1496,882	618,1714,1987,380,19501496,882,118,1971,787, 1202,1944,63,964,602,379,180,1128,342

	Relevant	Non-relevant
Retrieved	TP = 6	FP =13

Precision: tp/(tp + fp) = 6/ (6+13) = 0.31578
Compute DCG (Discounted Cumulative Gain)

$$DCG_p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2 i}$$

We are focusing on top 3 doc; therefore, we will only be considered doc till position 3(rank 3).

DCG3 = rel1 + rel2/log2 + rel3/log3
= 1 + 1/1 + 0/log3
= **2**

Use Case 3.
Request:

```
  GET lkr46_info624_201904_ted_search /_search
{
 "from": 0,"size": 20,
 "query": {
  "bool": {
   "must": [
    {
     "multi_match": {
      "query": "sustainable environmental",
      "fields": ["description","title"]
     }
    }
    ],
    "should": [
     {"rank_feature": {"field": "ratings.Informative" ,"boost":15.0}}
    ],
    "filter": [
     {"term": {"speaker_occupation": "Architect"}}]
  }
 }
}
```

Result Table 6.

Doc ID	Main Speaker	title	speaker_occupation	score	relevance
867	Michael Pawlyn	Using nature's genius in architecture	Architect	15.573233	1
722	Mitchell Joachim	Do not build your home, grow it!	"Architect",”designer"	11.6867485	1
1125	Bjarke Ingels	Hedonistic sustainability	Architect	11.506586	1
1567	Shigeru Ban	Emergency shelters made from paper	Architect	7.678932	1

Evaluation

Group 5
Friday 4th, 2020
Daniela Raygadas, dr967@drexel.edu
Om Prakash Singh, os338@drexel.edu
Likhil, Rachuri, lkr46@drexel.edu
Aishwarya Ravi, ar3646@drexel.edu

Document relevant to query	IR system retrieve
867,1125,722,1567	867,1125,722,1567

	Relevant	Non-relevant
Retrieved	TP =4	FP =0

Precision: tp/(tp + fp) = 4/ (4+0) = 1.0
Compute DCG (Discounted Cumulative Gain)

$$DCG_p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2 i}$$

We are focusing on top 3 doc; therefore, we will only be considered doc till position 3(rank 3).

DCG3 = rel1 + rel2/log2 + rel3/log3
= 1 + 1/1 + 1/1.6
= **2.625**

Comparison and Discussion

Without similarity scoring

Use case keyword	Precision	DCG3
climate change global warming	**0.7**	**2**
Bill Gates	**0.31578**	**2.625**
sustainable environmental	**1.0**	**2.625**

With similarity scoring

Use case keyword	Precision	DCG3
climate change global warming	**0.85**	**2**
Bill Gates	**0.31578**	**2**
sustainable environmental	**1.0**	**2.625**

From the tables above, we can observe that our custom similarities improved the precision of our first use case, in which the user was searching for TED Talks discussing about climate change and global warming. The precision score with the default BM25 was 0.7 and with our two custom similarities for BM25 and DFR we got a precision score of 0.85. The DCG remained the same for our first use case.

For our second use case, there precision score did not change. Our search engine retrieved the same 13 documents, but with a different ranking. We thought that by changing the similarity to Boolean for the main_speaker we would get better results as we are only interested in getting all the TED Talks by Bill Gates. Our search engine retrieved other main speakers that had first name as Bill and even retrieved TED Talks of Melinda Gates. The DCG score changed as the ranking of our documents retrieved changed. In this case, it decreases the score from a 2.625 to a 2.

For our third use case, the precision score and the DCG score remained the same. This could have happened as we have a small number of documents of an Architect talking about building sustainable environmental structures. Hence, our search engine retrieved these 4 documents in both indexes and with the same ranking. Therefore, we see a precision score of 1, as all the documents that were relevant were retrieved, and the DCG score remained the same as all the documents were relevant and the ranking was the same.

Although, we had some improvements in the score in the first use case, the index with the custom similarity did not make a huge impact to our search engine.

WHERE:

Our search index is located in Kibana, and the two index we used for our analysis and compared the results were:

1. lkr46_info624_201904_final_project (without similarities)
2. lkr46_info624_201904_ted_search (with custom similarities)

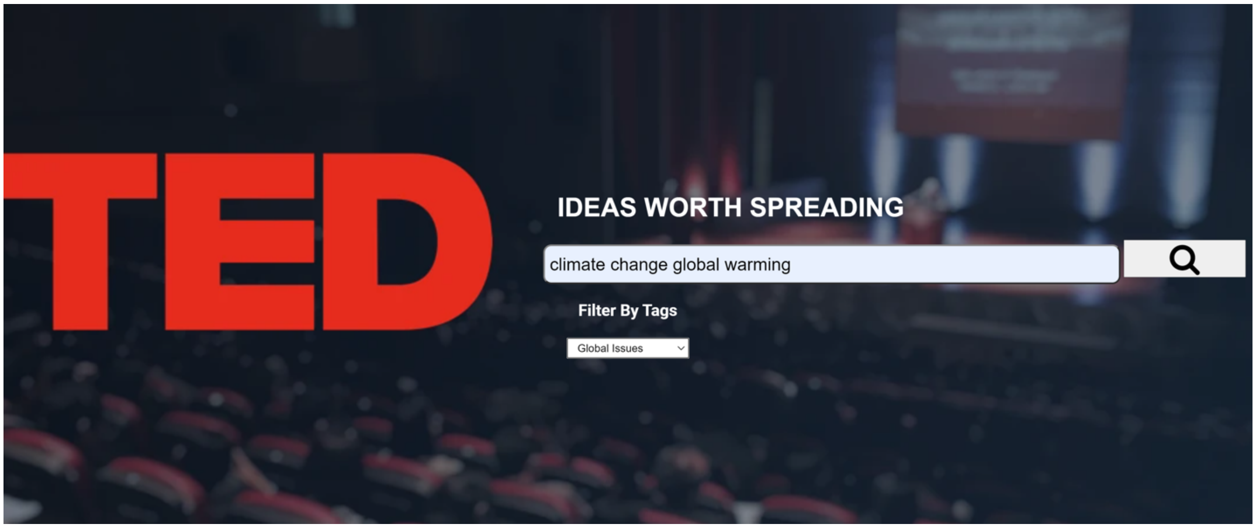
We are using the mapping with custom similarity which is present in **lkr46_info624_201904_ted_search** for the search engine

EXPERIENCES:

- Data was complicated as it was in different format, So, preprocessing was the main task before loading the bulk data. Preprocessing and bulk data load is done using pandas.
- Date format was in epoch, converted it into date format to get better understanding
- Ratings column was different from json, converted the format to json for loading into elastic search.
- Mappings were depending on many columns as we have speaker, speaker_occupation. We were deciding on the type to select for those fields whether this should be a keyword or text. This made a huge effect on the results as we have changed it from text to keyword. As they are more important as keywords than text
- Analyzer was also applied on few columns (description, title, main_speaker). Results from both with and without using custom similarity are varying lot.
- Filtering by tags was also varying results a lot. As they are keyword matching it was more biased. But we like to implement this functionality as sometimes we need to concentrate more on specific category. For that we have added a filter element from UI.

- Initially we thought of sorting our results using date columns, but that is diluting the results a lot. So, in the final query we didn't include date to reflect in results, instead just posted on UI so it would be easier while selecting the link.
- Parameter tuning had also made huge difference in our results for bm25 and dfr. Finally, best tuned values are kept in our mapping.

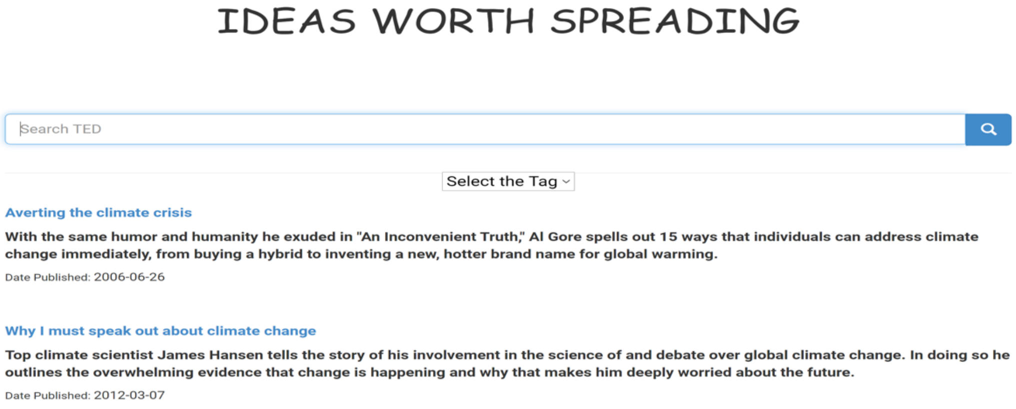
UI Demo Screenshots with sample search:



climate change global warming

Filter By Tags

Global Issues



Search TED

Select the Tag

Averting the climate crisis

With the same humor and humanity he exuded in "An Inconvenient Truth," Al Gore spells out 15 ways that individuals can address climate change immediately, from buying a hybrid to inventing a new, hotter brand name for global warming.

Date Published: 2006-06-26

Why I must speak out about climate change

Top climate scientist James Hansen tells the story of his involvement in the science of and debate over global climate change. In doing so he outlines the overwhelming evidence that change is happening and why that makes him deeply worried about the future.

Date Published: 2012-03-07