Udacity “AWS ML Engineer nanodegree”
Capstone Project Report

Denis Serbin

1. Definition

Project overview

The project is motivated by the Kaggle competition “Learning Equality - Curriculum Recommendations”, which focuses on building efficient ML models that could match educational content (files and videos in all kinds of formats) to curriculum (K-12) topics. Both content and topics have text descriptions in various languages, so every successful model is going to have a substantial NLP component in it.

I decided to take into account only the data (both topics and content items) given in English, that is, I discarded the data given in other languages.

Data used in the project

The training dataset is given in three files:
1. “topics.csv” - curriculum topics given in the form shown above,
2. “content.csv” - content items with descriptions (a sample item is shown above),
3. “correlations.csv” - an alignment of the topics with the content items.

All the files can be downloaded from the competition page.

Problem statement

Input:

A. A list of topics from K-12 curriculum.
Each topic has an id and several description fields (see below):

id	title	description	channel	category	level	language	parent	has_content
t_002eec45174c	Quadrilateral proofs & angles	Not all things with four sides have to be squares or rectangles! We will now broaden our understanding of quadrilaterals.	2ee29d	aligned	4	en	t_bfd74ce0fd04	TRUE

All topics are organized in a tree so that each topic belongs to a branch of the tree and it “knows” its parent.

B. A list of content items (files in various formats).
   Every content item also has an id and several description fields:

id	title	description	kind	text	language	copyright_holder	license
c_000751f58836	Tangents of circles problem (example 2)	Sal finds a missing angle using the property that tangents are perpendicular to the radius.	video	Angle A is a circumscribed angle on circle O. So this is angle A right over here. Then when they say it's a circumscribed angle, that means that the two sides of the angle are tangent to the circle. So AC is tangent to the circle at point C. AB is tangent to the circle at point B. What is the measure of angle A? Now, I encourage you to pause the video now and to try this out on your own. And I'll give you a hint. It will leverage the fact that this is a circumscribed angle as you could imagine. So I'm assuming you've given a go at it. So the other piece of information they give us is that angle D, which is an inscribed angle, is 48 degrees and it intercepts the same arc-- so this is the arc that it intercepts, arc CB I guess you could call it-- it intercepts this arc right over here. It's the inscribed angle. The central angle that intersects that same arc is going to be twice the inscribed angle. So this is going to be 96 degrees. I could put three markers here just because we've already used the double marker. Notice, they both intercept arc CB so some people would say the measure of arc CB is 96 degrees, the central angle is 96 degrees, the inscribed angle is going to be half of that, 48 degrees. So how does this help us? Well, a key clue is that angle is a circumscribed angle. So that means AC and AB are each tangent to the circle. Well, a line that is tangent to the circle is going to be perpendicular to the radius of the circle that intersects the circle at the same point. So this right over here is going to be a 90-degree angle, and this right over here is going to be a 90-degree angle. OC is perpendicular to CA. OB, which is a radius, is perpendicular to BA, which is a tangent line, and they both intersect right over here at B. Now, this might jump out at you. We have a quadrilateral going on here. ABOC is a quadrilateral, so its sides are going to add up to 360 degrees. So we could know, we could write it this way. We could write the measure of angle A plus 90 degrees plus another 90 degrees plus 96 degrees is going to be equal to 360 degrees. Or another way of thinking about it, if we subtract 180 from both sides, if we subtract that from both sides, we get the measure of angle A plus 96 degrees is going to be equal to 180 degrees. Or another way of thinking about it is the measure of angle A or that angle A and angle O right over here-- you could call it angle COB-- that these are going to be supplementary angles if they add up to 180 degrees. So if we subtract 96 degrees from both sides, we get the measure of angle A is equal to-- I don't want to make that look like a less than symbol, let make it-- measure of angle-- this one actually looks more like a-- measure of angle A is equal to 180 minus 96. Let's see, 180 minus 90 would be 90, and then we subtract another 6 gets us to 84 degrees.	en	Khan Academy	CC BY-NC-SA

The set of content items is not structured.

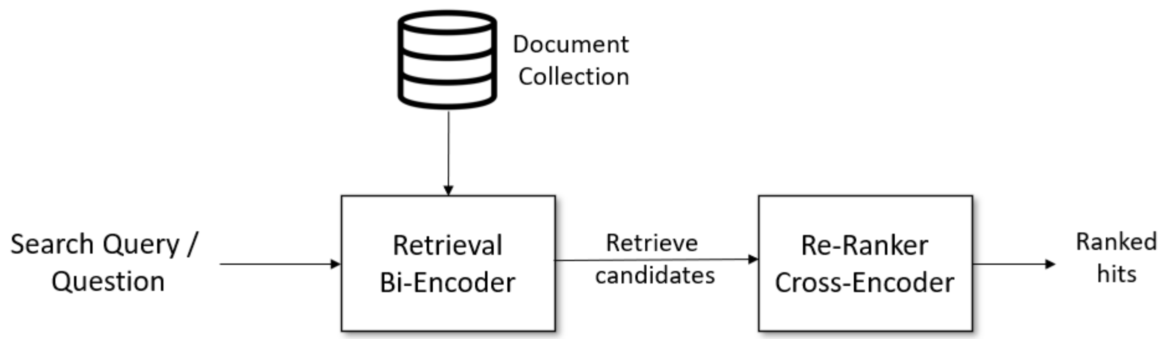
<u>Output:</u> an alignment of the given content items with the given topics (see below).

topic_id	content_ids
t_00004da3a1b2	c_1108dd0c7a5d c_376c5a8eb028 c_5bc0e1e2cba0 c_76231f9d0b5e
t_00068291e9a4	c_639ea2ef9c95 c_89ce9367be10 c_ac1672cdcd2c c_ebb7fdf10a7e
t_00069b63a70a	c_11a1dc0bfb99
t_0006d41a73a8	c_0c6473c3480d c_1c57a1316568 c_5e375cf14c47 c_b972646631cb c_d7a0d7eaf799
t_4054df11a74e	c_3695c5dc1df6 c_f2d184a98231

One topic can be aligned with several content items. At the same time, a content item can be aligned with multiple topics as well

<u>Solution strategy</u>

The stated problem can be viewed as an *information retrieval problem.* Namely, given a query (a topic in our case), one has to find all relevant documents from the list (given content items). Typically, the process can be described as follows (the image is copied from https://www.sbert.net/examples/applications/retrieve_rerank/README.html):



In my solution, I am performing the following steps.

A. Take a pre-trained SentenceTransformer model and fine-tune it on a dataset based on the given data.
B. Using the fine-tuned model, map all topic and content item titles to high-dimensional real-valued vectors.
C. Split content title vectors into clusters of nearest neighbors using the KNN algorithm.
D. Compose a list of the nearest content items for every topic.
E. For each topic, mark its content item neighbor by 1 if the content item is indeed related to the topic and by 0 if it's not related.
F. Based on the previous step, create a dataset for the re-ranker model, whose purpose is, for a given topic and content item, to determine the probability that the content item is related to the topic. To construct a re-ranker, take another (or the same) pre-trained SentenceTransformer model and train (or, fine-tune, since it's already extensively trained on huge datasets) it on the dataset built on the previous step.
G. Finally, for each topic, take only *relevant* content items based on predictions generated by the re-ranker created on the previous step.

Metrics

The competition "Learning Equality - Curriculum Recommendations" uses the F2 metric

$$F2 = \frac{5 \cdot Precision \cdot Recall}{4 \cdot Precision + Recall}$$

where

$$Precision = \frac{true\ positives}{true\ positives + false\ positives}$$

$$Recall = \frac{true\ positives}{true\ positives + false\ negatives}$$

I am using the same metric in my project.

2. Analysis

Data Exploration and visualizations

A thorough exploration of the datasets was performed in
https://www.kaggle.com/code/hasanbasriakcay/learning-equality-eda-fe-modeling

I copied some parts of the above exploration to **EDA.ipynb,** which is a part of the project submission. Below I'm using the images produced there and make some decisions based on them.

As I already mentioned above, the training dataset (available here) is given in three files:

- **topics.csv** - Contains a row for each topic in the dataset. These topics are organized into "channels", with each channel containing a single "topic tree" (which can be traversed through the "parent" reference). Note that the hidden dataset used for scoring contains additional topics not in the public version.
  - id - A unique identifier for this topic.
  - title - Title text for this topic.
  - description - Description text (may be empty)
  - channel - The channel (that is, topic tree) this topic is part of.
  - category - Describes the origin of the topic.
    - source - Structure was given by the original content creator (e.g. the topic tree as imported from Khan Academy). There are no topics in the test set with this category.
    - aligned - Structure is from a national curriculum or other target taxonomy, with content aligned from multiple sources.
    - supplemental - This is a channel that has to some extent been aligned, but without the same level of granularity or fidelity as an aligned channel.
  - language - **Language code for the topic. May not always match the apparent language of its title or description, but will always match the language of any associated content items.**
  - parent - The id of the topic that contains this topic, if any. This field is empty if the topic is the root node for its channel.
  - level - The depth of this topic within its topic tree. Level 0 means it is a root node (and hence its title is the title of the channel).

- has_content - Whether there are content items correlated with this topic. Most content is correlated with leaf topics, but some non-leaf topics also have content correlations.

A typical sample from **topics.csv** is shown below (the dataframe shape is (76972, 9)):

	id	title	description	channel	category	level	language	parent	has_content
0	t_00004da3a1b2	Откриването на резисторите	Изследване на материали, които предизвикват на...	000cf7	source	4	bg	t_16e29365b50d	True
1	t_000095e03056	Unit 3.3 Enlargements and Similarities	NaN	b3f329	aligned	2	en	t_aa32fb6252dc	False
2	t_00068291e9a4	Entradas e saídas de uma função	Entenda um pouco mais sobre funções.	8e286a	source	4	pt	t_d14b6c2a2b70	True
3	t_00069b63a70a	Transcripts	NaN	6e3ba4	source	3	en	t_4054df11a74e	True
4	t_0006d41a73a8	Графики на експоненциални функции (Алгебра 2 н...	Научи повече за графиките на сложните показате...	000cf7	source	4	bg	t_e2452e21d252	True

- **content.csv** - Contains a row for each content item in the dataset. Note that the hidden dataset used for scoring contains additional content items not in the public version. These additional content items are only correlated to topics in the test set. Some content items may not be correlated with any topic.
  - id - A unique identifier for this content item.
  - title - Title text for this content item.
  - description - Description text. May be empty.
  - language - Language code representing the language of this content item.
  - kind - Describes what format of content this item represents, as one of:
    - document (text is extracted from a PDF or EPUB file)
    - video (text is extracted from the subtitle file, if available)
    - exercise (text is extracted from questions/answers)
    - audio (no text)
    - html5 (text is extracted from HTML source)
  - text - Extracted text content, if available and if licensing permitted (around half of content items have text content).
  - copyright_holder - If text was extracted from the content, indicates the owner of the copyright for that content. Blank for all test set items.
  - license - If text was extracted from the content, the license under which that content was made available. Blank for all test set items.

A typical sample from **content.csv** is shown below (the dataframe shape is (154047, 8)):

	id	title	description	kind	text	language	copyright_holder	license
0	c_00002381196d	Sumar números de varios dígitos: 48,029+233,930	Suma 48,029+233,930 mediante el algoritmo está...	video	NaN	es	NaN	NaN
1	c_000087304a9e	Trovare i fattori di un numero	Sal trova i fattori di 120.\n\n	video	NaN	it	NaN	NaN
2	c_0000ad142ddb	Sumar curvas de demanda	Cómo añadir curvas de demanda\n\n	video	NaN	es	NaN	NaN
3	c_0000c03adc8d	Nado de aproximação	Neste vídeo você vai aprender o nado de aproxi...	document	\nNado de aproximação\nSaber nadar nas ondas ...	pt	Sikana Education	CC BY-NC-ND
4	c_00016694ea2a	geometry-m3-topic-a-overview.pdf	geometry-m3-topic-a-overview.pdf	document	Estándares Comunes del Estado de Nueva York\n\...	es	Engage NY	CC BY-NC-SA

Apparently, both topics.csv and content.csv contain rows with NaN fields.

- **correlations.csv** - Contains the content items associated to topics in the training set. A single content item may be associated with more than one topic. In each row, we give a topic_id and a list of all associated content_ids. These comprise the targets of the training set.

  A typical sample from **correlations.csv** is shown below (the dataframe shape is (61517, 2)):

	topic_id	content_ids
0	t_00004da3a1b2	c_1108dd0c7a5d c_376c5a8eb028 c_5bc0e1e2cba0 c...
1	t_00068291e9a4	c_639ea2ef9c95 c_89ce9367be10 c_ac1672cdcd2c c...
2	t_00069b63a70a	c_11a1dc0bfb99
3	t_0006d41a73a8	c_0c6473c3480d c_1c57a1316568 c_5e375cf14c47 c...
4	t_0008768bdee6	c_34e1424229b4 c_7d1a964d66d5 c_aab93ee667f4

Both topics and content items are given in various languages (below content items on the left and topics on the right).

	kind
video	61487
document	33873
html5	32563
exercise	25925
audio	199

	language
en	65939
es	30844
fr	10682
pt	10435
ar	7418
bg	6050
hi	4042
zh	3849
gu	3677
bn	2513
sw	1447
it	1300
mr	999
as	641
fil	516
km	505
kn	501
swa	495
or	326
pl	319
te	285
ur	245
tr	225
ta	216
my	206
ru	188
pnb	184

	license
CC BY-NC-SA	52088
CC BY-NC-ND	8714
CC BY	5927
CC BY-SA	4554
Public Domain	2044
CC BY-NC	691
CC BY-ND	17

	category
source	43487
supplemental	19368
aligned	14117

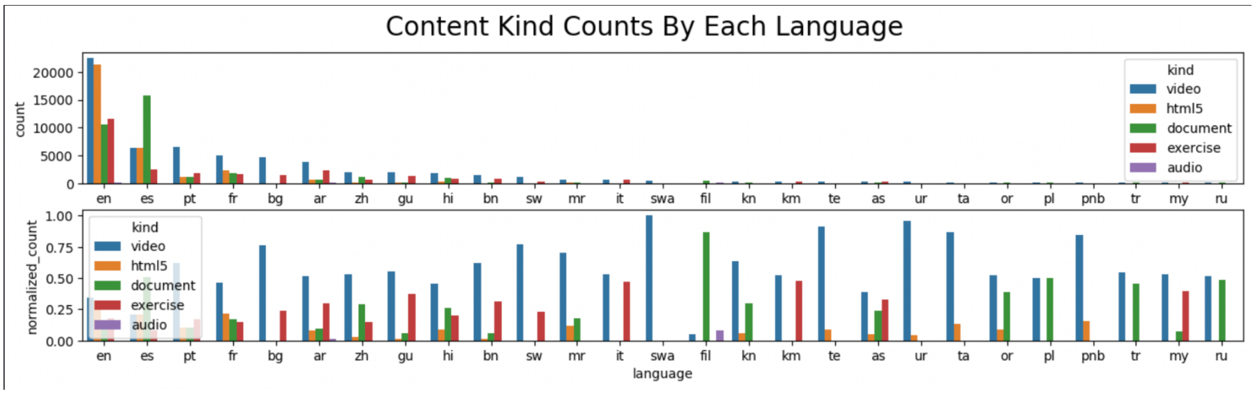
	level
4	38810
3	14898
5	9744
6	6210
2	4874
1	1104
7	1028
0	171
8	119
9	12
10	2

	language
en	36161
es	13910
pt	4177
ar	3701
fr	3701
bg	2867
sw	2860
gu	2320
bn	2176
hi	1786
it	866
zh	862
mr	300
fil	247
as	167
my	135
km	121
kn	119
te	93
or	70
ur	66
ta	60
pnb	51
pl	43
tr	40
swa	35
ru	34
mul	4

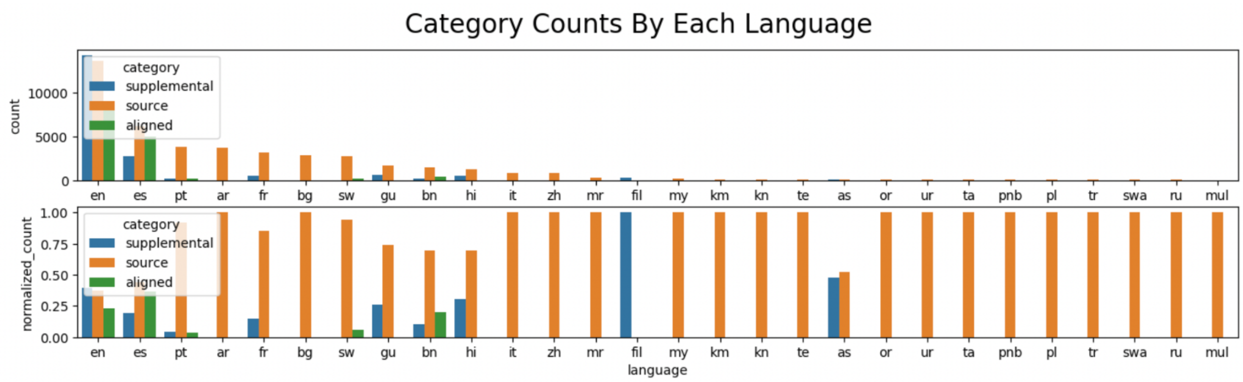
	has_content
True	61517
False	15455

From the above screenshots it follows that the majority of all topics and content items are given in English.

Next, the types of content items in English are more diversified than other languages (see the figure below):



The same applies to types of origin of topics: they are the most diverse among the languages represented (see the figure below).



Also, topics and content items given in English have more complete information (considerably less NaN fields than topics in other languages). Finally, as follows from the description of the **content.csv** and **topics.csv** datasets, the language of a topic **will always match the language of any associated content items.** **Hence, I decided to use only the English part of the data (both topics and content).**

From the analysis of the **correlations.csv** file it follows that topics with assigned content on average are aligned with 4-5 content items.

	content_ids_len	
	mean	median
has_content		
True	4.550271	3.0

Algorithms and techniques

As I already mentioned in the Solution strategy section, in my solution I'm using a well-known approach based on fine-tuning two pretrained transformer models. This is a very common approach based on the *information retrieval* interpretation of the problem. On a greater scale, the main technique used in the solution is *knowledge transfer*.

Benchmark

I am not aware of a specific benchmark result or threshold for this specific problem. The Kaggle competition "[Learning Equality - Curriculum Recommendations]" organizers used a hidden test set to measure performance based on the average F2 metric. Since I didn't have access to the test set, I used the F2 score on the validation set to measure performance of my models.

3. Methodology

The whole approach to solving the problem and the basic re-ranker model configuration I borrowed from https://www.kaggle.com/code/ragnar123/lecr-xlm-roberta-base-baseline. I modified the model itself and adapted the code for training and inference in Sagemaker.

Data preprocessing

- As was already mentioned above, I discarded all non-English topic and content item rows.
- I decided to use only the "Title" field in all topic and content rows, so I dropped all rows with NaN in the "Title" field both in **content.csv** and **topics.csv.**
- Based on the refined datasets from **content.csv** and **topics.csv,** and the **correlations.csv** file, I created a new dataset **uns-train.csv** for fine-tuning the retriever model. Essentially, the 'uns-train' dataset is a list of pairs (topic title, content item title), where the topic is aligned with the content item based on the **correlations.csv** file. A sample is shown below.

	set
127528	['Business Writing', 'Vegetarian Lunch Options...
127529	['Business Writing', 'Mid-Project Report on Hi...
127530	['Introduction', 'Introduction to ratios']
127531	['Scalar Projections', 'Scalar Projections']
127532	['Scalar Projections', 'Scalar Projections Pra...

- Using the fine-tuned retriever, I created another dataset **sup-train.csv** to train the re-ranker model. Every row of this dataset contains a pair (topic title, content item title), where the vector embedding of the topic title is *close* to the vector embedding of the content item title based on the retriever model output. The label of the pair is 1 if the topic is indeed aligned with the content item based on the **correlations.csv** file, and 0 otherwise. A sample is shown below.

	topics_ids	content_ids	title1	title2	label
1838781	t_8b5ee088546e	c_f7be0b0ebd73	Rules for Dilations	Absolute Value of Integers Practice	0
1838782	t_8b5ee088546e	c_225713f4ec00	Rules for Dilations	The Combinations [ur] and [ər] Practice	0
1838783	t_8b5ee088546e	c_2788ef655019	Rules for Dilations	Rules for Dilations Practice	1
1838784	t_8b5ee088546e	c_8d198c06dff3	Rules for Dilations	Similar Squares	1
1838785	t_8b5ee088546e	c_bf65f5ccfeb8	Rules for Dilations	Rules for Dilations	1

Implementation

1. I considered several pre-trained sentence transformers as candidates for the retriever model. I tested all of them on 20% of the **uns-train.csv** dataset and obtained the following results (with respect to the average cosine metric):
    a. 'all-distilroberta-v1' - 0.49077263
    b. 'paraphrase-distilroberta-base-v2' - 0.5043382
    c. 'multi-qa-distilbert-cos-v1' - 0.509438
    d. 'all-mpnet-base-v2' - 0.5305708
    e. 'multi-qa-mpnet-base-dot-v1' - 0.6256573

    The last transformer 'multi-qa-mpnet-base-dot-v1' produced the best result, so I took it as the base model for the retriever.

2. I fine-tuned 'multi-qa-mpnet-base-dot-v1' on the **uns-train.csv** dataset using the **uns-train.py** script for training, deployed it, and tested it on the **uns-train.csv** dataset using the inference script **uns-inference.py.**
    Performance of the fine-tuned 'multi-qa-mpnet-base-dot-v1' model improved to 0.8861534 (on a random 1% sample of the **uns-train.csv** dataset - inference is expensive and takes a long time, that's why I took only a small sample).

3. I split all topics (only English, non-NaN rows) into training (**train_topics**) and testing (**test_topics**) datasets. Then using the retriever (the fine-tuned 'multi-qa-mpnet-base-dot-v1' model), I constructed embeddings of the topic titles into a 768-dimensional vector space. I did the same to all content item titles from the **content.csv** dataset.

4. Next, I ran the KNN algorithm on vectorized content item titles and then for each topic from **train_topics** I constructed a set of 50 nearest neighbors (among content item titles) based on the cosine metric. Using the **correlations.csv** file I labeled each (topic title, content item title) pair by 0 or 1. The result is the dataset **sup-train.csv.**

5. I constructed a model based on the pre-trained sentence transformer 'all-mpnet-base-v2' to output the probability that the pair (topic title, content item title) is correlated. Essentially, on top of the output of 'all-mpnet-base-v2' I added several linear, dropout, and activation layers. Then I trained the model (the re-ranker) on the **sup-train.csv** dataset.

6. Finally I deployed the trained model and tested it on the **test_topics** dataset.

On all steps discussed above the main difficulty was time and cost of training and inference of both models (retriever and re-ranker).

Refinement

In my initial approach I used the 'paraphrase-multilingual-mpnet-base-v2' sentence transformer as the base model for both the retriever and re-ranker. Eventually, I switched to the solution described above.

4. Results

Model evaluation and validation

My model shows a decent validation F2 score around 0.3 (trained only for 2 epochs). With more training, the score could be considerably improved, I believe, but each epoch takes ~80 min on 'ml.p3.2xlarge' instance and the improvement is going to be pretty costly. At the same time, I think my model closely follows the 'retriever - re-ranker' approach outlined in
https://www.sbert.net/examples/applications/retrieve_rerank/README.html
So I believe it's robust enough (assuming the approach itself is appropriate).

Justification

As I already mentioned, I don't have a benchmark model or F2 score to compare my model's performance with. The winning F2 score in the "Learning Equality - Curriculum

Recommendations" competition was 0.76450, but it was achieved on a hidden test set and all the data available (in all languages) was used in training. So, it is hard to compare.

I tested my model on a small random sample from the **test_topics** dataset (that wasn't used in the training of the re-ranker), but the best test F2 score I obtained was much lower than the best CV score: only around 0.05. I don't know how to explain that, my model is definitely not overtrained.

5. References

- Learning Equality - Curriculum Recommendations
- https://www.sbert.net/examples/applications/retrieve_rerank/README.html
- https://www.kaggle.com/code/hasanbasriakcay/learning-equality-eda-fe-modeling
- https://www.kaggle.com/code/ragnar123/lecr-xlm-roberta-base-baseline