A Needle in a Data Hysack

Tweakipedia

Final Project

> *...If you tweak something such as a system or a design, you improve it by making a slight change...*

Raya Elsaleh, Naama Shamash Halevy, Arielle Barouch

1-3-2022

Project title: Tweakipedia

Team members info

1. Naama Shamash Halevy, naama.shamashhal@mail.huji.ac.il, neamash
2. Raya Elsaleh, raya.elsaleh@mail.huji.ac.il, rayae
3. Arielle Barouch, arielle.barouch@mail.huji.ac.il, arielleb

Problem Description

Wikipedia is one of the most accessible information sources used all over the world, Thus, it is necessary to keep it reliable and up to date. Editing Wikipedia pages is done mostly voluntarily by users. Therefore, some pages might be un-updated with recent events.
We are interested in finding a solution to the said problem by identifying whether a page is in need for updating. We are also interested in inspecting Wikipedia articles and gaining insights about their updating habits.

Data

We collected our data from three main sources:

1. Wikipedia Website: Wikipedia, as it's well known, is constructed from articles (we may as well refer to them by pages). Every article covers a certain topic. As for now, Wikipedia has a total of 54,754,527 pages. Among them 6,419,412 articles in English, while the Hebrew Wikipedia contains more than 307,000 articles. As of 20 April 2021, the size of the current version of all articles compressed is about 19.52 GB.
   Every Wikipedia page has an edit history page in which one can find information about every update it has undergone.
   By scraping Wikipedia page and its edit history page, we extract the following features:
   - Per page: Title, URL, Creation date, Word count, Categories , Section titles.
   - Per page per update: Date and time, Number of bytes added/deleted, Editor username.

   Our two main databases from Wikipedia are: (i) PagesDB: consists of the data per page. And (ii) UpdatesDB: consists of data per page per update and is indexed by the pages (for easier access).
   However, they do not contain all the pages and updates in Wikipedia because of lack of space and time to collect them all. So instead, we constructed a code that given headers of pages in Wikipedia it will fill the forenamed databases and return them. It also can draw the headers randomly.

2. GDELT: a platform that monitors the world's news media in over 100 languages, with daily updates. The total size of the GDELT 2.0 GKG table, when stored in its original compact nested delimited format, now stands at 1.47TB. All GDELT 2.0 tables are available in Google BigQuery, allowing interactive analysis.

3. Google Trends: a website by Google that analyzes the popularity of top search queries in Google Search across various regions and languages.

Solution and results

The main goal of our project is to extract features for a classifier. Given a Wikipedia page, the classifier should predict whether the page should be updated. Our main challenge is extracting the relevant features, either from Wikipedia itself or from external sources.

The first challenge we faced was defining what is considered a "significant" update in a Wikipedia article, and we present the way we dealt with this challenge. Then, we present the approaches we took to extract the needed data to construct a future classifier.

Significant & Insignificant updates

Many updates in Wikipedia pages are insignificant. For example, a recent update of Business article consisted of changing the sentence "Fish for sale for in Dhaka, Bangladesh with a price tag of 395 Bangladeshi taka per kilogram" to "Fish for sale in Dhaka, Bangladesh, with a price tag of 395 Bangladeshi taka per kilogram." In this update the word "for" was deleted and two punctuation marks were added. Such updates are negligible and should not be considered. In our research, if needed, we disregard those updates to attain clearer and better insights.

Negligible updates can be diagnosed in two ways:

1. Update size
   On each page history, there is a feature per update called "update size", which is the number of added chars minus the number of deleted chars in the update. We noticed that most of the updates with small size in absolute value are negligible updates. Based on that, we need to set a threshold that determines the lower bound of a significant update size. Observing our data, we set the threshold to be 210 bytes.
2. Bag- of-words
   Another feature per update represents two bags of words: one that contains the deleted words and one that contains the added words in the update. After deleting the stopwords in the bags, we noticed that most of the negligible updates end up with two very similar bags (i.e., the added words are similar to the deleted ones, a property that may raise the possibility that the update didn't change a lot overall). On the other hand, non-negligible updates end up with two different bags. So, to identify negligible updates we calculated the cosine similarity between the two said bags and recognized them as negligible updates if their cosine similarity is higher than 0.1.

We focused on the first filter - update size - due to running time considerations.

To show the effect of our filtration, we took the 41 main articles that correspond to one of Wikipedia's major topic classifications. We calculated how many times those main articles were updated in the last 5 years and sorted them by this value. In Figure 1, the result of the mentioned calculation in shown, with and without filtration. This shows that the number of updates that got filtered by our definition is significant.

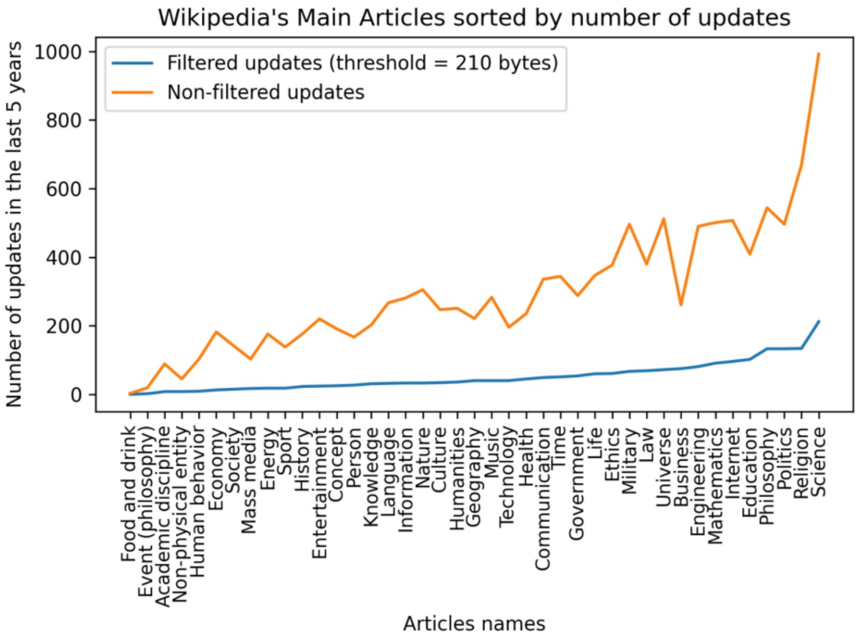


Figure 1

Regulation of Updates

To examine how often pages are updated, we took ~200 randomly selected pages and checked for each the number of significant updates it had in 2021. From Figure 2 we conclude that most articles do not get updated very often (in a significant manner, according to our definition). It is not surprising, since most pages in Wikipedia are very small and relate to esoteric topics. Therefore, uniformly random selection is likely to yield this sort of pages. Moreover, as shown in the next section, we observed a connection between the size of a page to its updating tendency.

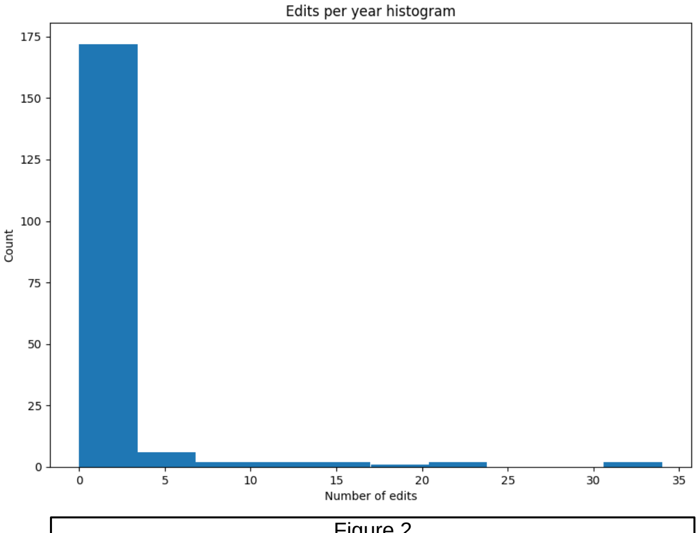


Figure 2

Relation to Page Size

We now show that the page size (defined as the number of words in the article) may be an important feature to predict the number of updates of an article. We demonstrate that by plotting and correlating an array of the updates count in the last 5 years, to the size of the article. We checked 500 randomly generated pages from

Wikipedia, and the resulting correlation was 0.55 with p value = 1.42e-41. We plotted the page size in relation to the number of updates and fitted a regression line that shows the trend, as shown in Figure 3.

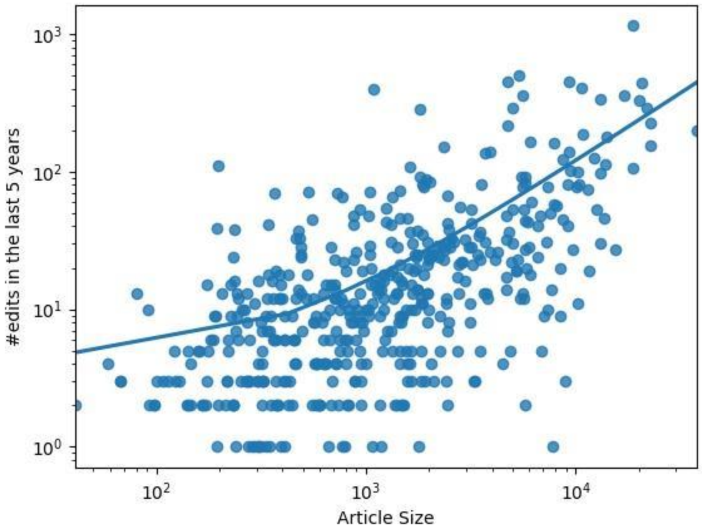


Figure 3

Next, we wanted to observe how long it takes to update a specific event, i.e., examine the time passing from the occurrence of the event until it is reported on its corresponding Wikipedia page. An easy and interesting (albeit dark) way was to check the update of death occurrences.

Death update statistics

We parsed a few hundred names of deceased people from the yearly Wikipedia category (for example: 2019 deaths). For each departed, we extracted the exact date of death by using a regular expression (regex) and plotted the Wikipedia updates around that time. Then, we approximated how long it took for Wikipedia to update the death, by measuring the days between the death and the next update. Here we did not filter updates by their length.

In Figure 4 we present the updates of the articles of Sergey Smiryagin, a Russian freestyle swimmer, and of Yawovi Agboyibo, a Togolese politician. Their date of death is marked as a blue line on the plot. We observed that in most articles, including the two presented, there is a peak in updates in the day or following days after the demise. For all the downloaded data (100 randomly selected pages of people who died during the last 5 years), the mean number of days to update was 2.7 days, with standard deviation of 5.92. The maximum observed amount of time was 30 days. We conclude that the English Wikipedia is probably reliable in terms of updating this information.

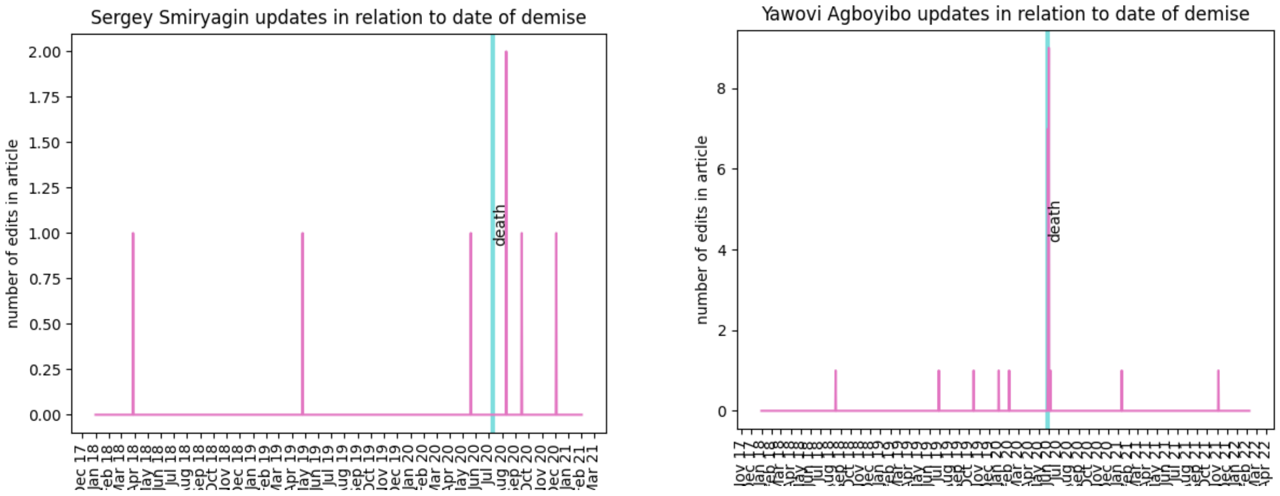


Figure 4

GDELT and Google trends

To gain information about the need for an update of a Wikipedia page, we used two outer sources- GDELT and google trends. Both are public databases that monitor attention to topics, by calculating a value that represents the attention a certain topic received per unit of time:

- GDELT calculates the percentage of monitored media that mentioned the topic in a given time period, resulting in values between 0 and 100. We call this percentage *the media percentage.*
- Google trends calculates the proportion of a search of the topic out of all searched topics (in a given time unit), resulting in values between 0 and 100. We call this percentage *the search percentage*

At first, we had to demonstrate that these values correlate to the number of edits of a Wikipedia article. To achieve that, we selected several topics and time intervals, and compared the resulting GDELT and google trends values to the number of Wikipedia edits. Indeed, we managed to show that such a relation exists for some pages. The following table shows the Pearson correlation for selected articles:

Topic/article	Correlation GDELT	P value GDELT	Correlation google trends	P value google trends
Ukraine	0.57	$4.52e-94$	0.64	$5.34e-122$
Attack on Titan	0.32	$1.59e-21$	0.53	$5.81e-60$
Pfizer	0.3	$1.49e-17$	0.12	0.0009

We also plotted a smoothed version of the data and observed if peaks correlated to known events. For example, in Figure 5 we can see the page Ukraine which has a peak in the last week in all our measured values.

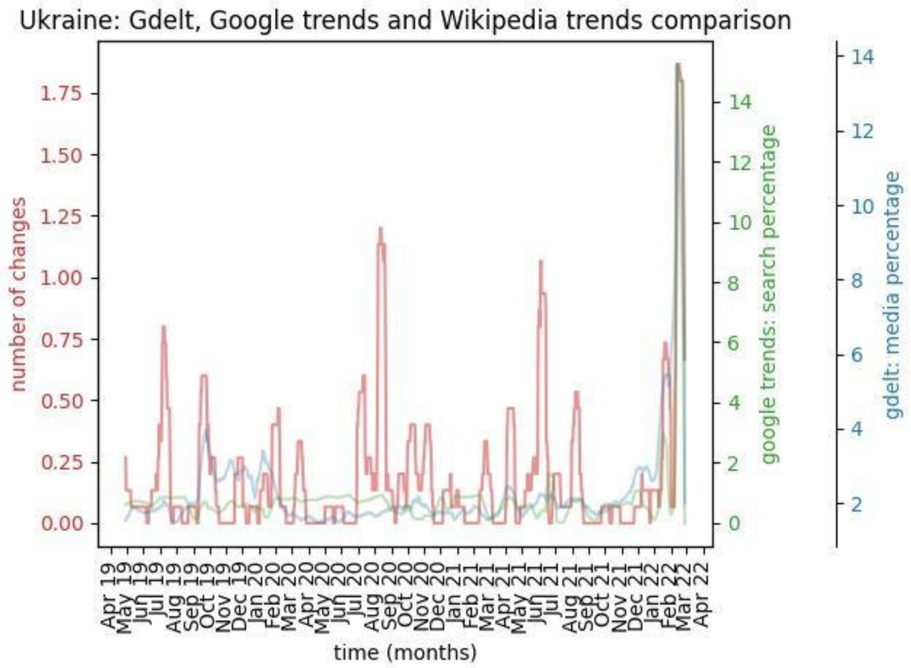


Figure 5

Pages update pattern

A strong page feature to indicate its need to be updated is its update pattern. We observe three different update patterns:

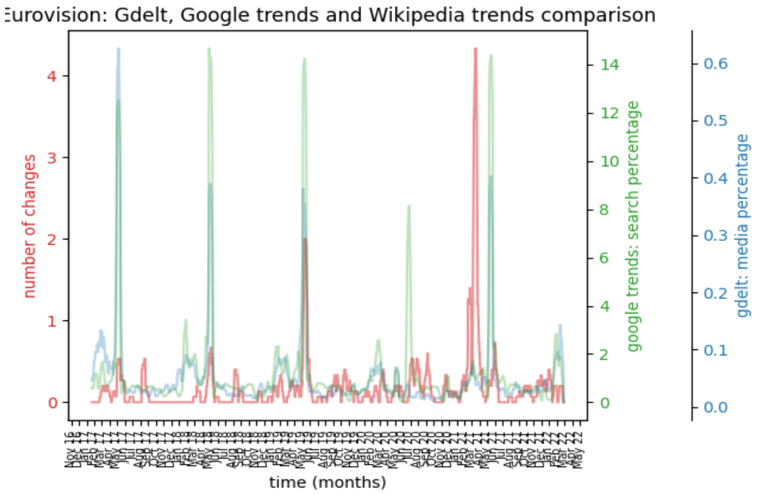


Figure 6

1. <u>Active pages:</u>
   Those pages are updated regularly. They are characterized by constant or periodical updates. For example, Figure 6 shows the “Eurovision Song Contest” which get updated annually around the date of the competition, with the exception of 2020 in which the competition was canceled.

2. <u>Inactive pages:</u>
   Those are the pages which were once active (mostly when they were created) and now they are not. A reason for that may be that they handle settled issues. Take for example the page “Doug Naysmith”, a local British politician, for whom there has not been any significant updates in the past 5 years.

3. <u>Sleeping beauties:</u>
   Those pages have not been active for a long time, and suddenly they become active (or need to become active). Such pages are harder to identify and our cue to identify them is News and Trends databases: GDELT & Google trends. Take for example the page “Pfizer” and its compatibility with GDELT and Google

trends (Figure 7). Pfizer released the first vaccines in December 2020, which was also the time of the highest peak for the Wikipedia edits and GDELT. However, the highest peak in google trends occurred 3 months later in March 2021. We assume it might be related to the vaccine reaching more people and therefore the global discussion and search around it increased.

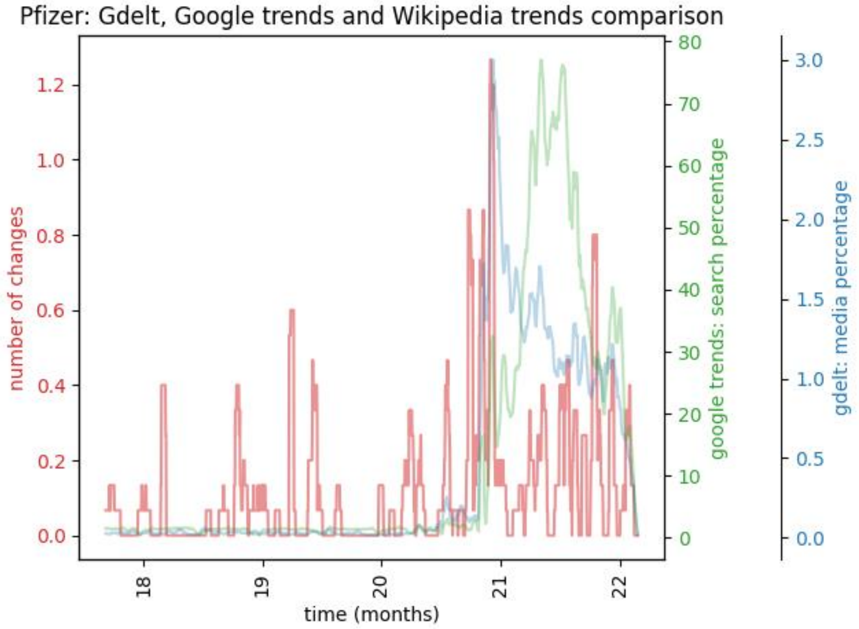


Figure 7

Page rank

Page rank is an algorithm we encountered in the course: given a set of pages with links between them, it will rank each page according to its connections the other pages in the set.

As a sanity check, we ran the algorithm on a set of articles from the biology field. Edges have direction and are defined as a link from source article to target article. In Figure 8, the node size corresponds to its PageRank. As expected, we can observe that the more general topics have higher page-rank values. Topics such as Genetics, Evolution and Plant have higher values than amino acids or rhesus macaque.

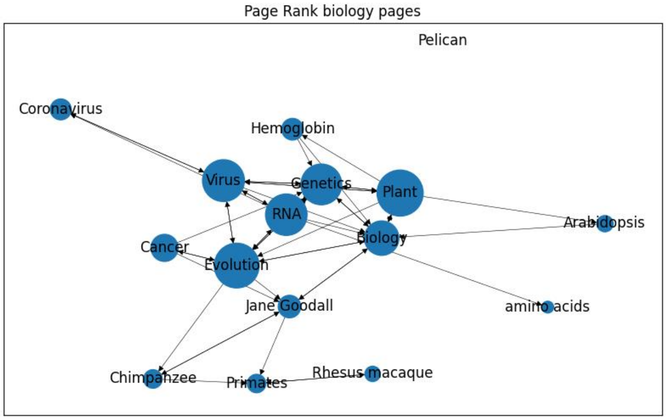


Figure 8

Pages Communities

We think of a set of Wikipedia pages as nodes in an undirected graph, in which the edges can be defined in one of many ways. For example, two pages are connected iff one of them has a link to the other, or iff they are textually similar. On the graph we can apply methods to community detection, such as the **Girvan-Newman algorithm**. The motivation to detect those communities is mainly to infer what pages are highly related. Then, given a query page, our classifier should be able to determine whether articles that are highly related to the query have been updated recently – information that can raise the possibility that the target article should be updated too.

Communities by synchronic updates

The motivation of this section is to check whether related pages (by topic) tend to be updated together.

For a page p, we define its *update vector* v_p as follows:

$$v_p[i] \ = \ \#\, significant\ updates\ of\ p\ in\ week\ i$$

For every week i in the last 10 years.

Two pages p_1 and p_2 are *synchronized* if $cosine(v_{p_1}, v_{p_2}) \geq 0.35$. For a set of pages P, the *synchronic graph* $G_P{}^{sync}$ is the graph with P as a set of nodes and every two nodes are connected iff they're synchronized. We ran Girvan-Newman on $G_P{}^{sync}$ with $P = \{p \mid p\ is\ a\ country\}$, i.e., the set of all countries. We would like to see related countries in the same community. The graph $G_P{}^{sync}$ can be seen in Figure 9 (we filtered out communities of 2 or less countries, and for readability added ~30% of the node labels):

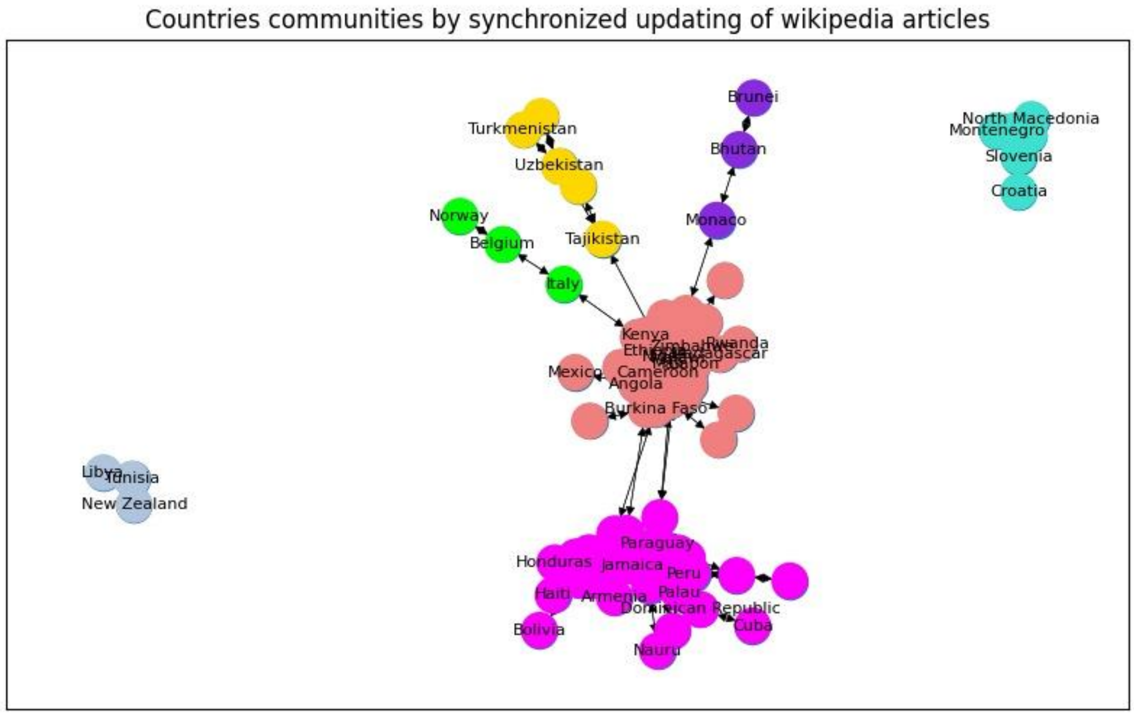


Figure 9

The algorithm resulted in a reasonable partition of the nodes, for example:

- The pink community consists mainly of countries from either central or south America, and some islands from the Pacific.
- The red community consists mainly of countries from Africa.
- The yellow community consists only of countries from central Asia ("stan" countries).
- The turquoise community consists mainly of countries from former Yugoslavia (south - east Europe).

Communities by Textual-Similarity

The motivation of this section is to check whether related pages (by topic) are textually similar.

The similarity metric we used is **cosine** similarity. We now describe how given a set of pages S and a threshold $t \in [0,1]$ we construct its corresponding graph: For each $i, j \in S$, we define $cos(i, j)$ to be the cosine similarity (on text, as seen in class) between the *summary* attribute (after removing stop words and stemming) of pages i and j (a page object of the Wikipedia package has an attribute called *summary*, which is a short description of the page content). The edge $\{i, j\}$ exists iff $cos(i, j) \geq t$. We now have a *cosine-graph* G_S^t, on which we can apply Girvan-Newman.

We used the above method to study pages of people, i.e., $S \subseteq \{p : \ p\ ia\ a\ human\}$.

We created a database of people pages as follows: we randomly chose 60 pages of people in the pop-culture categories, 60 pages of athletes, 60 pages of politicians, and 60 pages of mathematicians & computer scientists. For all those pages we created the cosine graph with threshold 0.1 (i.e., nodes represent pages, edge exists iff the cosine similarity of the summary attributes is higher than 0.1). On this graph we ran Girvan-Newman to detect communities. We were happy to see that the pages partitioned in the expected way, and people engaged in the same field were in the same community (Figure 10). The following colors correspond generally to the following communities (with few exceptions):

Musicians, athletes, culture figures, politicians (American), computer scientists, politicians (Israeli), politicians (others).

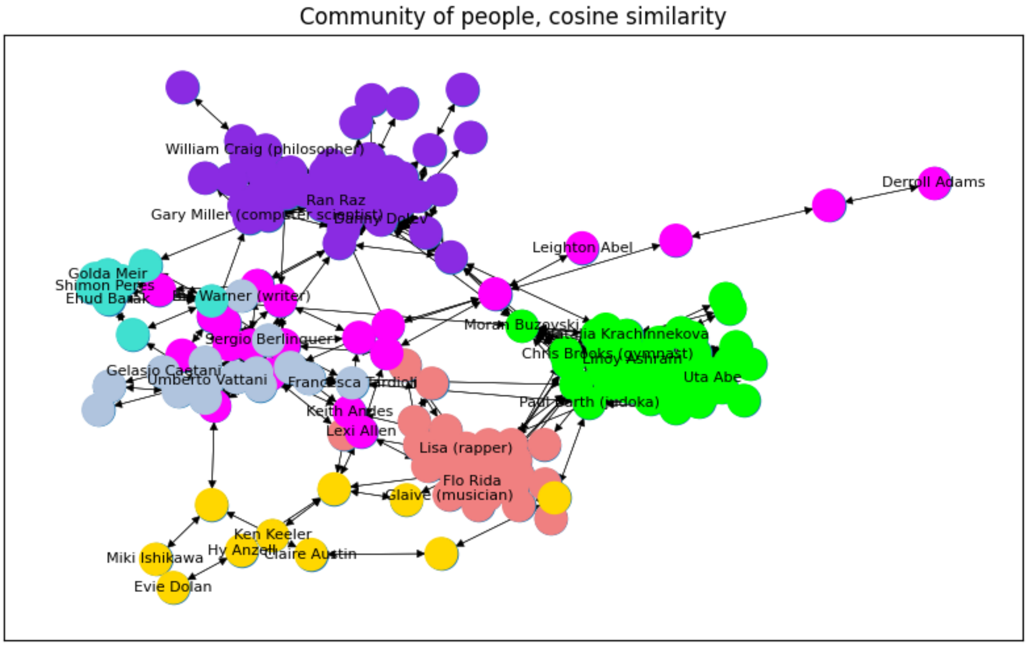
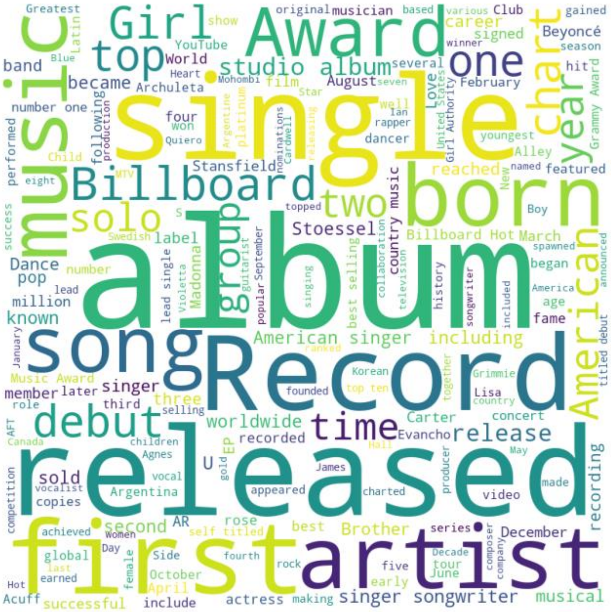


Figure 10

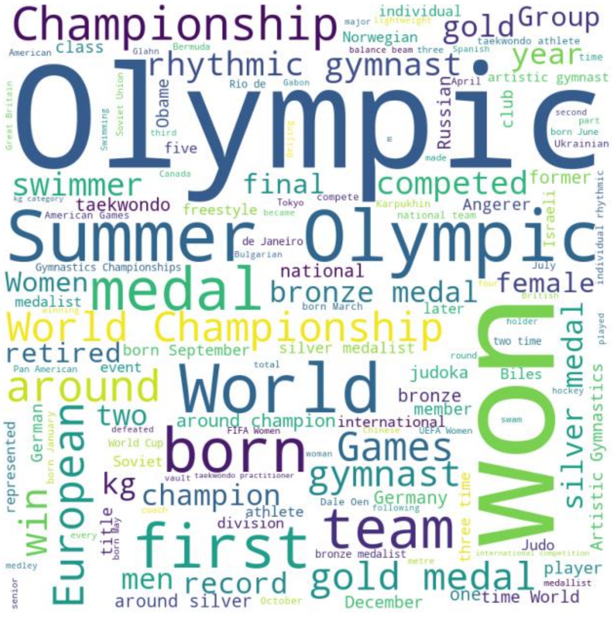
We note that the three different politicians' communities are close to each other in the figure, and to a lesser extent, so do the musicians and culture figures' communities.

For each community we created a word cloud of all words in the concatenation of all summary attributes of the community pages (after removing stopwords). The word clouds help us evaluate the quality of the partition to communities, and indeed the frequent words generally match the category. We give some examples:

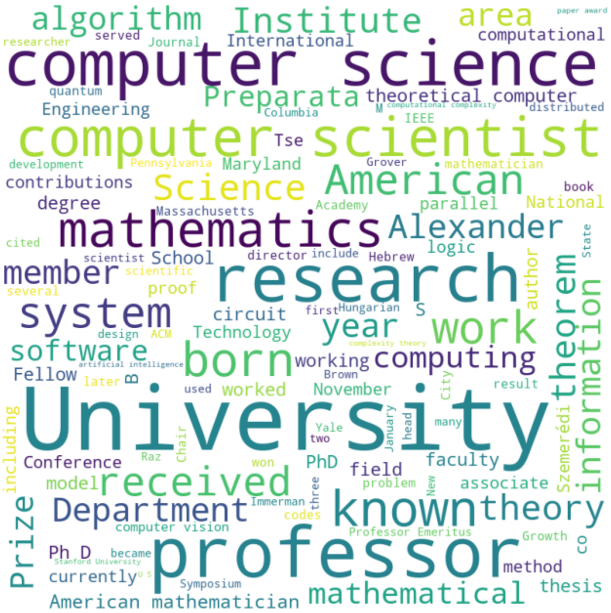
The musicians community word cloud:



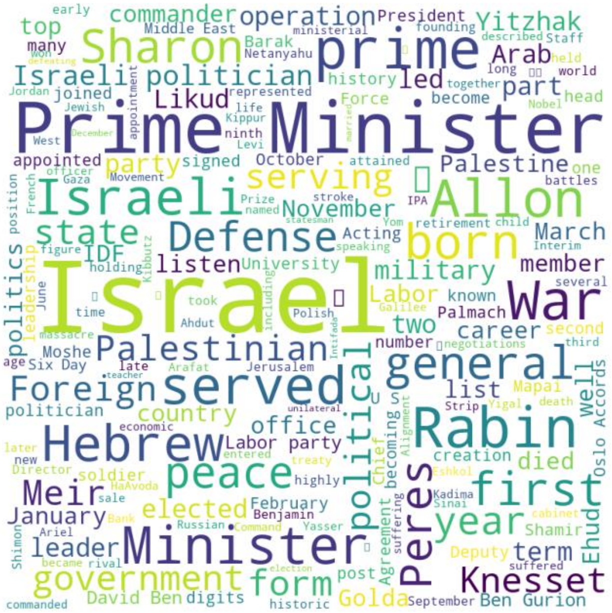
The athletes community word cloud:



The mathematicians & computer scientists community word cloud:



The Israeli politicians community word cloud:



Future Work

At first, we intended to produce a classifier that predicts the need of an update for a Wikipedia page. However, while examining and extracting the needed features for that goal, we realized that it would require much more work to be done.

If we had the needed resources, we would design a dataset that will be based on the following features, and possibly more:

1. Article size: we saw that there is a positive correlation between the size of the article and the number of times it was updated.
2. Page rank: we would like to examine the relation between pageRank of a page to the number of updates (if positive, could be a feature to imply need of update).
3. Communities: a page is likely to be updated if related pages were updated recently.
4. Media percentage by GDELT: since there is a positive correlation between media percentage and number of updates in some pages, we infer that the higher the media percentage - the higher the page is likely to get updated.
5. Search percentage by google trends: since there is a positive correlation between search percentage and number of updates in some pages, we infer that the higher the search percentage - the higher the page is likely to get updated.
6. Last update of the page (This feature can be used to decide whether a new update is needed. We will set the classifier predict this feature and if it’s nowhere near the real value then the page needs updating)

In addition, it would be interesting to examine the difference between Wikipedia languages in terms of updating. It is probable that the English Wikipedia will update more frequently than other languages, and it would be interesting to repeat our analysis for another language especially Hebrew.

Conclusion

In this project, we examined different ways to find connections between Wikipedia pages, with the aim of identifying important features for the prediction of an article’s update. We demonstrated the significance of methods such as cosine similarity and synchronic updates to find connections between pages. We also examined correlation between the edit frequency of an article and other features. To achieve that, we developed methods to extract and process information from Wikipedia in a meaningful way.