Search Engine Entity Cards

Yash Kalia , Claudia Hauff
[1]TU Delft

Abstract

Search engine Entity Cards(ECs) display concise information from the web about a topic or subject in response to a user query. The topic or subject can be a person, an organization etc. and is referred to as an "Entity". The specific topic under research is how to determine which entity is most relevant for the query in terms of helping the user find the information he/she is looking for. and what information about the chosen entity to display to answer the query. The information can be in the form of but it not limited to text, images and hyperlinks. Research into the concepts of EC focuses on different components of the EC widget for example entity linking, tagging, extraction and fact summary generation. In the developed "EC algorithm" these concepts are combined into an implementation of an Entity Card widget and then evaluated. The EC algorithm utilizes tools such as DBPedia, DBPedia Spotlight and the Bing Web Search API to generate an entity ranking for a query. The results of evaluating the top ranked entity imply that the EC algorithm retrieve on average a slightly to moderately relevant entity to the user. The fact retrieval algorithm had predictably worse results given the complexity of finding truly relevant facts about entities.

1 Introduction

Search Engine ECs are a concise way to display dense complex information to the user in an easily readable and easy to navigate fashion [2]. They are a common functionality in search engines like Google that form a part of the search results displayed by the browser in response to a certain query. These entity cards offer utility in efficient information retrieval and navigation. The hyperlinks to related information about the entities that are displayed in the EC allow the user to navigate swiftly.

From Figure 1 and 2 of ECs from Google and Bing, it is apparent that Entity Cards follow a semi strict structure of contents across different queries. For example Google displayed quotes from Albert Einstein while Bing displayed a timeline of his life. They were however alike in the fact that they both have images and introductory paragraphs, along with entity attributes. Ideally EC's should adapt to the query topic and the specific terms(if they are provided) in the query to display exactly the summary the user is looking for[6]. This importance of query dependent summaries is demonstrated in the results of [6], and they note "query-dependent summaries (DynES) are preferred over query-agnostic ones (DynES/imp)" for about half of the queries".

The process of finding an entity to display for a certain query is called entity retrieval[5], and ranking a list of retrieved entities in order of relevance to the query is called entity ranking[7]. The ranking is done by means of a ranking function.

The research question is "How to determine which entity to display information for, based on a query?" since picking the correct entity(assuming one is appropriate to the query and information need) to display information for is essential to aid the user in his search. One important concern with the question is whether a query pertains to a single entity, example "who was Albert Einstein", see Figure 3 or not, example "list of American presidents", see Figure 4. Since EC's have to display a limited amount of information it is necessary to consider which information specifically to include in the EC and which to filter out as unnecessary.

Entity extraction and ranking for a query is not equally easy for all queries.For example for queries such as "Capital city of Netherlands" where the entity mentioned in the query text is not actually the entity the user is primarily interested in. These type of queries make extracting the entity from the query text itself difficult. This is where looking at top retrieved web documents also plays a role in finding the correct entity. There are many other examples of queries where entity extraction using query text itself if difficult. "list of American presidents" is a query where the relevant entity is not mentioned by name in the query at all.

First an analysis on useful relevant literature is done, followed by an implementation phase. Finally the implementation is evaluated in the form of questionnaires and the results are explained.

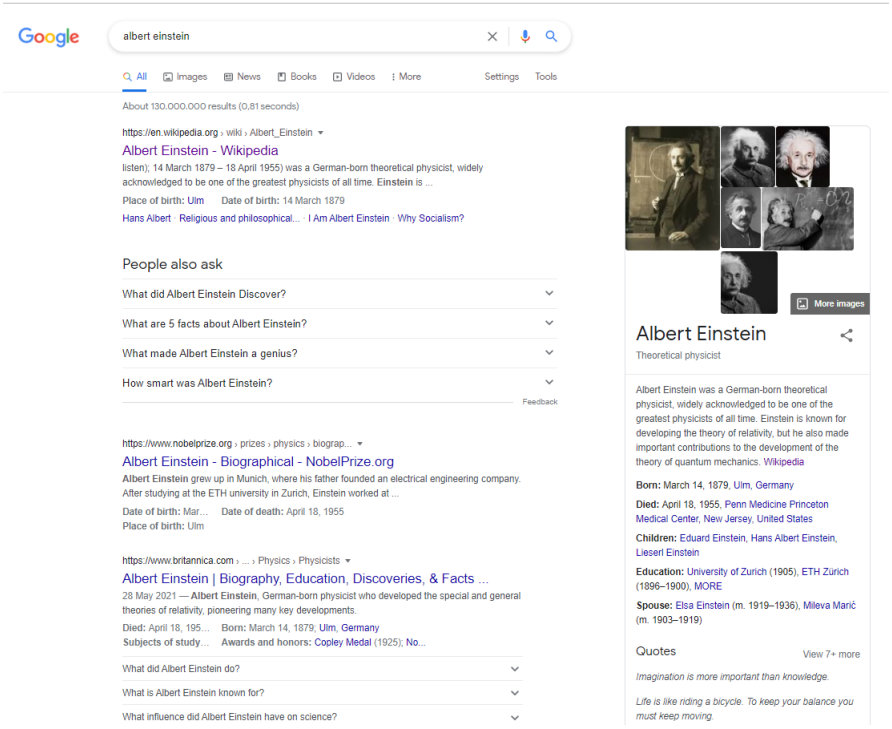

Figure 1: Example of an Entity Card(EC) displayed on the right side for search query ”Albert Einstein” on Google Search.

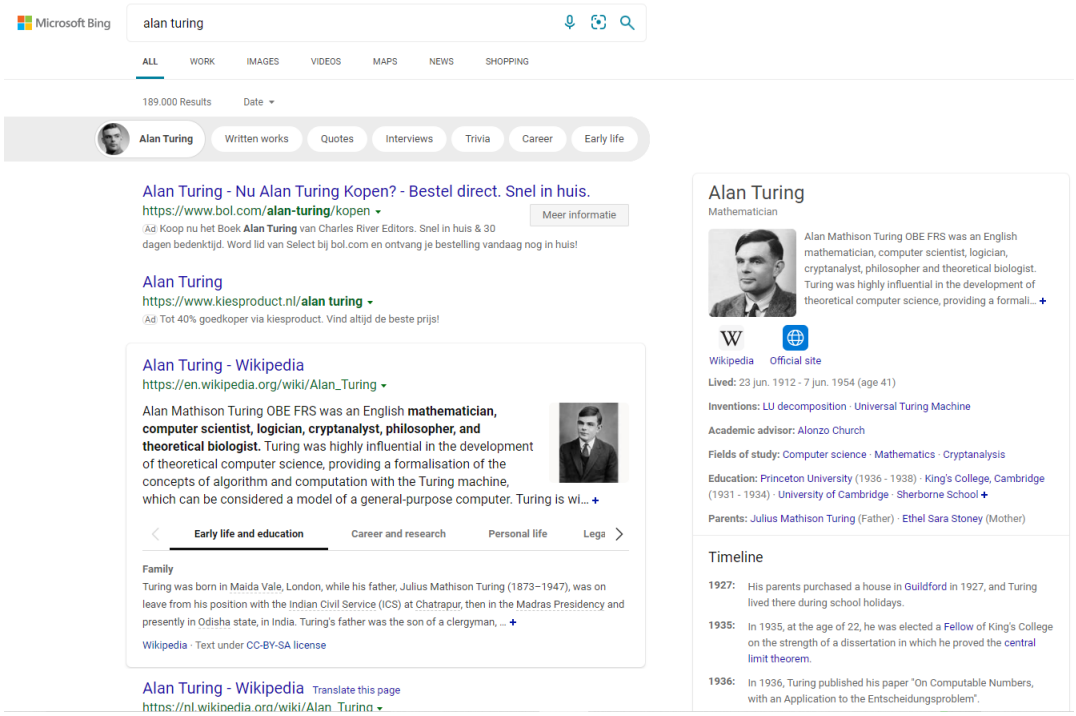

Figure 2: Example of an EC displayed on the right for search query ”Alan Turing” on Bing.

2 Background

2.1 Entity Extraction

DBPedia is a free structured knowledge base derived from Wikipedia where information is structured in a tag-info or

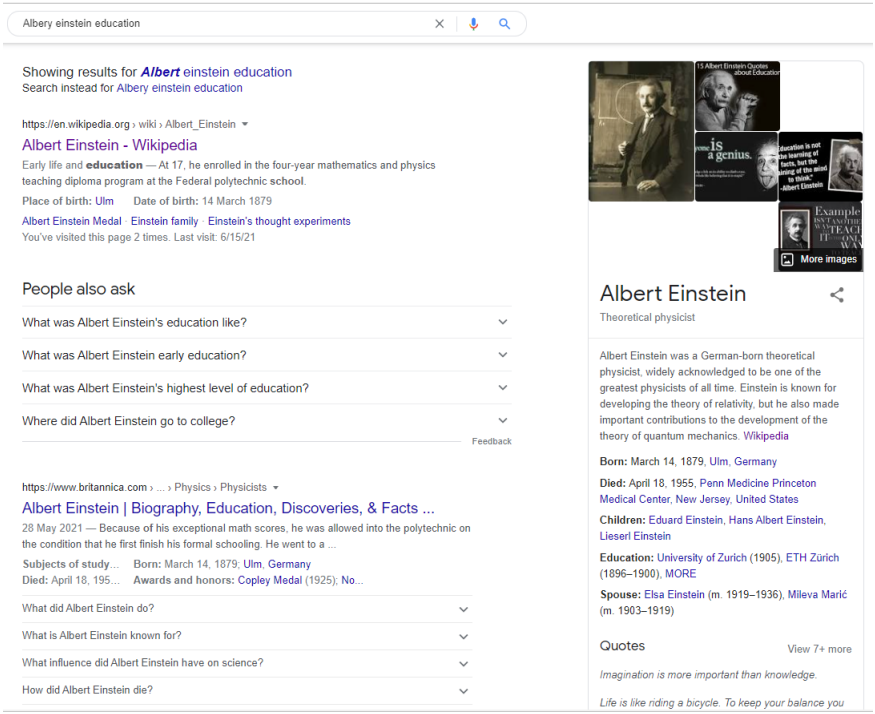

Figure 3: Query which specifically requires education information concerning Albert Einstein

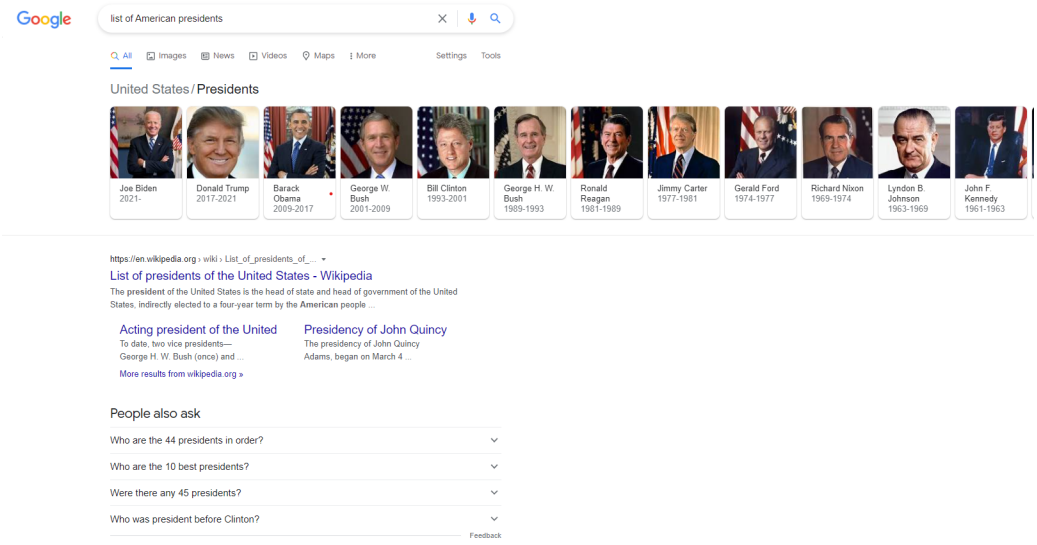

Figure 4: Google deems list form of results as more appropriate for the query ”presidents of America”

key-value form which makes it very appealing for extracting pinpoint information for an EC[8]. In addition it also offers the tool DBPedia Spotlight, which is used in the entity extraction.

DBPedia Spotlight can recognize mentions of DBPedia entities in text, and returns links to the DBPedia pages on the same entities[9].What is is missing to our knowledge is an evaluation of the usefullness of DBPedia Spotlight as a tool in Entity Cards. We have therefore evaluated DBPedia Spotlight with web queries to evaluate if the correct entity is retrieved by Spotlight.

2.2 Entity Ranking

Given query, top retrieved web documents and a list of entities, the mention frequency:

$$MentionFrequencyIdf = tf_e * log(N/df(e))$$

is a strong ranking method by itself[12] for ranking entities. Here the tf_e is the term frequency of entity ”e” ie the num-

ber of occurrences of the entity "e" within the fixed number of web documents under consideration "N", and $df(e)$ is the number of documents in which of entity "e" occurs.

2.3 Content of EC's

The problem of dynamic entity summarization deals with generating query dependent summaries[6]. Their evaluation results indicate that dynamic (query dependent) entity summaries are preferred by users over static (query ignorant) summaries and also demonstrate the effectiveness of their approach to the presented problem. In addition to a novel and tested approach to Dynamic Entity summaries that they refer to as "DynES", they also make available a benchmark for the fact ranking sub task of generating dynamic entity summaries which can be used to evaluate different approaches to the same problem. The benchmark can be used to evaluate both dynamic and static new approaches to the paper's original problem.

3 Methodology

The task of developing an entity card widget can be divided broadly into 2 main problems. The only user provided input is a query string. The overview of the EC algorithm is given on Figure 5,

3.1 Entity Extraction

The input for the entity extraction algorithm is the search query string entered by the user and the top retrieved documents retrieved by Bing for the query. Since the entity in question may be mentioned in the query and the web documents returned for that query, a tool is required to recognize mentions of entities stored in a huge corpus in text. This problem is referred to as Named Entity Recognition(NER) or simply "entity tagging" as the occurrences of entities are tagged in the query text [10]. The entity corpus being used here is DBPedia.

DBPedia was chosen as the corpus since it is derived from Wikipedia, which is a huge collection of information. According to the official DBPedia web blog the DBpedia 2016-04 release describes over 6 million entities and over 9.5 billion pieces of information. An example of what a DBPedia page looks like is given in Figure 6.

Once the mentions of entities have been identified and an entity ranking has been produced we need to display the entity information to the user. For this, URLs to the entity information pages for the entities recognized in DBPedia are needed. These DBPedia pages can be accessed to extract the entity facts that will be displayed to the user in the entity card. This task is referred to as "entity linking" as the entities recognized in text are linked to their DBPedia pages.

DBPedia Spotlight is responsible for both the tasks outlined above. According to Mendes et al. Spotlight is "a system to perform annotation of DBpedia resources mentioned in text."

This tool has been evaluated on random web queries from MS MARCO in the experiment section. DBPedia Spotlight was chosen over other tools like the Staford-NER[4] and TagMe[3] because it is offered by DBPedia itself and so all the entities will be extracted from the same corpus as the one where the entity facts are drawn from.

One of the biggest advantages Spotlight has over other NER implementations is that where other NER systems are restricted in the resource types they can identify, Spotlight can identify entities from over 272 resource classes in the DBPedia Ontology. Stanford NER on the other hand can only recognize a small fraction of them.Furthermore Spotlight being a tool offered by DBPedia can directly return links to DBPedia entity pages which is the knowledge base that was picked for information extraction of entities. The choice of Spotlight therefore removes the need to locate the corresponding DBPedia page for an entity extracted by some other NER tool.

An approach with simply using DBPedia Spotlight to extract entities from the query string itself was evaluated. The details of this experiment can be found in the evaluation section. The results however demonstrated that this would not retrieve the correct entity in most cases. So simply using the query string for entity extraction does not provide sufficient information to the algorithm to find the right entity for the user. Therefore, the scope of input information for the algorithm needs to be broadened. This is done by taking into account the content of the top retrieved documents, returned by the Bing search engine for the query.

The algorithm was then changed to instead extract entities based on the contents of the web documents returned by Bing. This content had to be extracted from the documents in a string form so that it could be passed over to DBPedia Spotlight. This is where Trafilatura is used. Trafilatura is a open source tool which extracts the main content from top retrieved web pages so that entity extraction can be carried out on the contents[1]. Once the content of all the web pages under consideration is extracted using Trafilatura, DBPedia Spotlight is used for finding entities in all those texts. From there a list of entities is prepared which finishes the entity extraction part of the EC algorithm. Using the list of entities, data on the entities like term frequency and inverse document frequency is computed to calculate the ranking of each entity in the list according to the formula for mention frequency:

$$MentionFrequencyIdf = tf_e * log(N/df(e))$$

.

The top ranked entity according to formula is used as the entity to be displayed in the EC.

3.2 Fact Retrieval

Fact retrieval refers to finding the information about the top ranked entity that should be displayed to the user. The EC algorithm currently only accesses the DBPedia page for the top ranked entity and retrieves the information corresponding to the abstract tag of the entity. The abstract was chosen as among all the other tags because as a general heuristic it is obvious that the information in the abstract would contain a summary of the most relevant facts about the entity. An example of what a DBPedia abstract looks like is given in Figure 6. These facts are then displayed to the user.

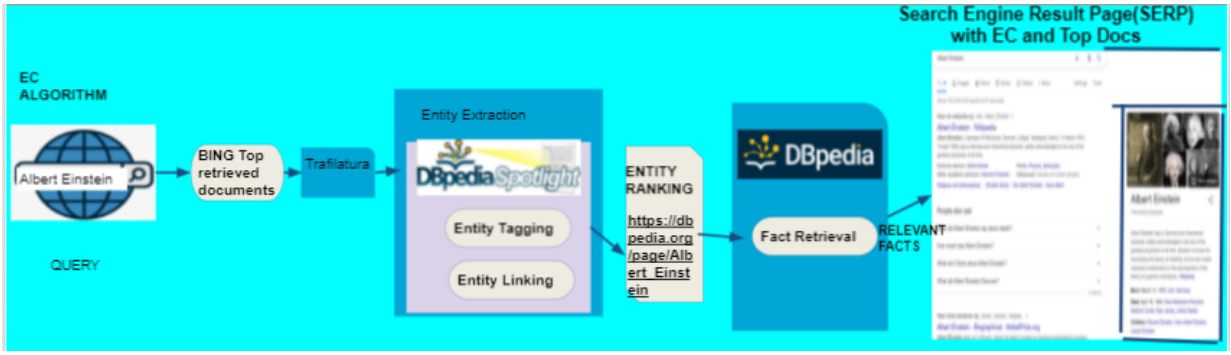

Figure 5: Diagram representation for the EC algorithm

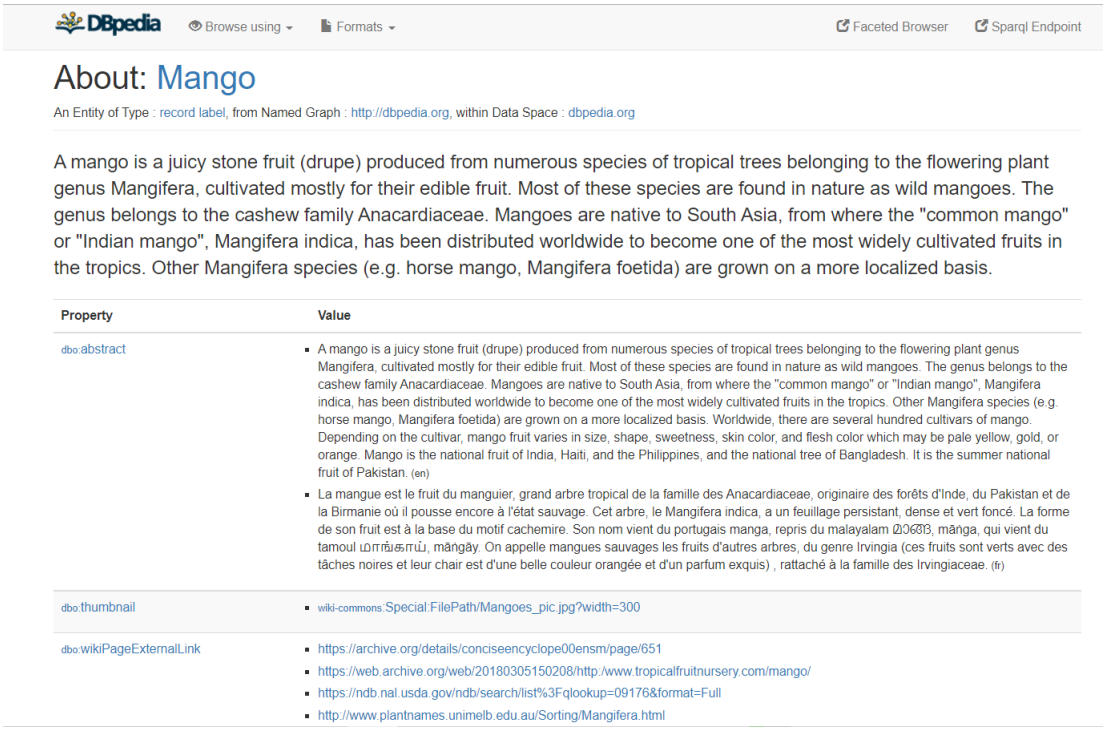

Figure 6: DBPedia page for ”Mango”

4 Experimental Setup and Results

4.1 DBPedia Spotlight

The performance of DBPedia Spotlight was evaluated to check if it would be a suitable tool for entity tagging and linking. It was evaluated against manual annotations of what entities should be displayed for 30 random web queries from the MS MARCO data set, alongside Bing entity card results for reference to what a modern search engine would result. MS Marco, which is short for ”MAchine Reading COmprehension” dataset is a free of charge collection of datasets, comprising of 1,010,916 anonymized questions— sampled from Bing’s search query logs[11]. Additionaly, ”The size of the dataset and the fact that the questions are derived from real user search queries distinguishes MS MARCO from other well-known publicly available datasets for machine reading comprehension and question-answering. We believe that the scale and the real-world nature of this dataset makes it attractive for benchmarking machine reading comprehension and question-answering models. ”[11] makes MS MARCO a viable option for queries to evaluate DBPedia Spotlight. Bing is used as reference because for entity extraction the top retrieved documents are extracted using the Bing Web Search API which uses the Bing Search Engine. Hence, using Bing is the right search engine to serve as reference.

The table can be found online here[1].

The evaluation metric for DBPedia Spotlight is whether or not DBPedia Spotlight returns the entity with the same name as the one in the manual annotation. This is so because if the

[1]https://docs.google.com/spreadsheets/d/1HIgFdUib_nzoBhQGgm5xEUtwPgdnAyt4KGY2FPY9w98/edit?usp=sharing$

Figure 7: The abstract of the DBPedia page for Microsoft Windows as the top ranked entity.

entity was different it is clear that the entity return is different from the corresponding one in the manual annotation. The manual annotations were derived from researching the results of the queries and formulating whether entities should be displayed or not, and if so which entity should be displayed, for every query. Out of 30 queries for only 6 of them did Spotlight return exactly the result that was expected in the manual annotation. Given this it was clear that merely looking at the query is insufficient to decide which entity should be displayed, if at all. This is why scanning top retrieved documents for which entity is most relevant by the Mention Frequency is quintessential to getting more accurate results. One observation is that Spotlight returned entities for 6 queries even when no entity should be displayed according to the manual annotations. See Figure 8 for details of the results for every query.

4.2 Evaluating Entity Extraction and Ranking Algorithm

The entity extraction algorithm by using a Likert scale to evaluate how relevant the top ranked entity in the ranking was for over a 100 queries to the user. The Likert scale comprised of 5 values ranging from

- Not Relevant
- Slightly relevant
- Moderately relevant
- Fairly Relevant
- Highly Relevant

The queries for the evaluation were randomly drawn from the same dataset as the Spotlight evaluation dataset but over 98 queries were taken for this evaluation as larger set was needed to sufficiently test the algorithm.

The evaluators were given the query and the corresponding entity that was ranked at the first position by the EC algorithm and they were asked to rate how relevant it was. The average relevance for the entity extraction algorithm was 2.29/5 which is in the ”Slightly relevant ” to ”moderately relevant” bracket. The most relevant entities were obtained for queries like “how to check if your iphone is original” and “what is prevailing wage law in california”, both pertain to clear unambigous entities and so the web documents retreived are focused on the entities in question thus leading to relevant results.

The least relevant entities were obtained for queries like ”how long is a flight from miami to antigua”.There are multiple entities in this query which is a downside of the algorithm as the web documents will tend to focus(in terms of instances) approximately equally on both the entities or on some other entity entirely.This is what is happening in this case with the entity retrieved being the ”Australian National University”. This leads to irrelevant results.

	Query	Bing Results(EC's)	DBPedia(Entity(s) detected)	Manual Annotation(Ground Truth)
2	what is prescribed to treat thyroid storm	No Entity	Hyperthyroidism	No entity
3	what is presquipp	Requip	No entity	No entity
4	what is previcox for horses	Firocoxib	No entity	Firocoxib
5	treatment for gastritis inflammation	No entity	Gastrisis, Inflammation	No entity
6	what is prevailing wage law in california	Prevailing wage	Prevailing wage, law and California	No entity
7	how does windows firewall work	Windows Firewall	WIndows Firewall	Windows Firewall
8	who plays the father on the goldbergs	The Goldbergs	No entity	Jeff Garlin(EC)
9	who plays the hobbit	No entity	Hobbit	Martin Freeman
10	treatment of pseudogout of wrist	No entity	Calcium pyrophosphate dihydrate crystal deposition disease and Wrist	Colchine
11	who is igor pavlov	Igor Pavlov	Igor_of_Kiev and Ivan_Pavlov	Igor Pavlov
12	what is priligy	Dapoxetine	No entity	Dapoxetine
13	how long is a flight from chicago to new york	No entity	Chicago and York	No entity
14	netflix series keepers how many episodies	No entity	Netflix	No entity
15	how long is a flight from miami to antigua	No entity	Miami and Antigua	No entity
16	what is primary school in scotland	Education in Scotland	Primary School and Scotland	No entity
17	what county is park city, utah in	Park City	Park City and Utah	Summit County
18	how to check if your iphone is original	No entity	No entity	No entity
19	who plays zoe barnes	Kate Mara	No entity	Kate Mara
20	who produces optima batteries	OPTIMA batteries	Johnson Controls	Johnson Controls(HVAC company)
21	how long is a lunar month? quizlet	No entity	Lunar Month and Quizlet	No entity
22	largest natural gas supplier	No entity	Natural Gas	Gazprom
23	triop definition	Triops	No entity	Tadpole Shrimp
24	who published romeo and juliet	Romeo and Juliet	RoMEO, Juliet(from Romeo and Juliet)	Cuthbert Burby
25	who pulls the sun greek mythology	No entity	Sun and Greek Mythology	Helios
26	how long is a operational tour	Tour of Duty	No entity	No entity
27	what is processor_architecture	Computer Architecture	No entity	No entity
28	how long is a personal check valid	No entity	Cheque	No entity
29	who recommendation for post exposure arv prophylaxis	Post-exposure prophylaxis	Armoured Recovery Vehicle and Preventive healthcare	No entity
30	who recorded love potion number 9	Love Potion no. 9	Love Potion no. 9	The Searchers (band)
31	who recorded song chains?	No entity	No entity	The Beatles

Figure 8: Evaluation of DBPedia Spotlight with random web queries.

4.3 Evaluating Fact Retreival Algorithm

Like the Entity extraction and retrieval algorithm, the fact retrieval for the top ranked entity was evaluated using the Likert scale for 99 queries from the same dataset. For the quries a part of the facts that were retrieved were displayed in the questionnaire as the entirety was too large. The current fact retrieval approach only requests and displays the abstract for the entity identified. The average value for the relevance of the facts on the Likert Scale was 1.91. The performance can be explained because the fact retrieval algorithm does take access specific information about the entity from DBPedia.

The fact retrieval performs well for queries like "what is processor architecture", for which the abstract of the query has the relevant facts. It does nor perform well for queries like "what is previcox for horses" when the information about the entity is so specific that it was not important enough to be mentioned in the abstract of the entity page.

5 Responsible Research

The queries used for evaluating DBPedia Spotlight are a random selection of 30 queries from the evaluation set of the 13/11/2018 dataset of queries. The queries were part of the dataset that was the focus of the 2020 and 2019 TREC Deep Learning Track. The link to the dataset can be found here[2].It is important to note that Trafilatura employs web scraping, which is only legal where the sites allow the data displayed on them to be scraped. Since Trafilatura is being used only for publicly available documents that are retrieved by Bing, there should be no danger of Trafilatura being used to scrape/pick confidential information from a website. In addition, no personal information is stored or accessed by the EC algorithm.It is also important to note that for the offline evaluation of the algorithm no personal or demographic data was collected for the people who were involved in the evaluation and that it was completely anonymous with no requirement of personal data.

[2]https://microsoft.github.io/msmarco/

6 Conclusions and Future Work

This paper introduces and evaluates an algorithm to implementing a search engine Entity Card widget and answers the research question- "How to determine which entity to display information for, based on a query?" DBPedia Spotlight has been evaluated as a tool for entity linking and tagging in the context of an Entity Card widget and given the low accuracy of results when using only the query for entity retrieval it is clear that the query alone cannot be used to extract the most relevant entity and that it is necessary to look at the contents of the top n retrieved web documents on the query instead. The evaluation results of this approach demonstrates that it is a much approach for entity retrieval than using only the query.

There is a lot of avenue available for improvement, specifically in the fact retrieval aspect of the developed widget. Some possible improvements being able to extract different entity summaries using useful keywords from the query that instruct the algorithm as to what part of the entity the user is interested in. This is the part of the widget that needs the most improvement in terms of extracting facts that are more relevant to the user. Additional future work can include retrieving

not only text but also images and hyperlinks to related entities to assist the user in his/her search and differentiating between queries where displaying an entity will be helpful or not.

References

[1] Adrien Barbaresi. Generic web content extraction with open-source software. In *KONVENS 2019*, pages 267–268. GSCL, 2019.

[2] Horatiu Bota, Ke Zhou, and Joemon M. Jose. Playing your cards right: The effect of entity cards on search behaviour and workload. In *Proceedings of the 2016 ACM on Conference on Human Information Interaction and Retrieval*, CHIIR '16, page 131–140, New York, NY, USA, 2016. Association for Computing Machinery.

[3] Paolo Ferragina and Ugo Scaiella. Tagme: On-the-fly annotation of short text fragments (by wikipedia entities). In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, page 1625–1628, New York, NY, USA, 2010. Association for Computing Machinery.

[4] Jenny Rose Finkel, Trond Grenager, and Christopher D Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 363–370, 2005.

[5] Faegheh Hasibi, Krisztian Balog, and Svein Erik Bratsberg. Exploiting entity linking in queries for entity retrieval. In *Proceedings of the 2016 acm international conference on the theory of information retrieval*, pages 209–218, 2016.

[6] Faegheh Hasibi, Krisztian Balog, and Svein Erik Bratsberg. Dynamic factual summaries for entity cards. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, page 773–782, New York, NY, USA, 2017. Association for Computing Machinery.

[7] Rianne Kaptein and Jaap Kamps. Exploiting the category structure of wikipedia for entity ranking. *Artificial Intelligence*, 194:111–129, 2013.

[8] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic web*, 6(2):167–195, 2015.

[9] Pablo N Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. Dbpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th international conference on semantic systems*, pages 1–8, 2011.

[10] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26, 2007.

[11] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. Ms marco: A human generated machine reading comprehension dataset. In *CoCo@ NIPS*, 2016.

[12] Michael Schuhmacher, Laura Dietz, and Simone Paolo Ponzetto. Ranking entities for web queries through text and knowledge. In *Proceedings of the 24th ACM international on conference on information and knowledge management*, pages 1461–1470, 2015.