CS60092 Information Retrieval Project Options

Instructor: Somak Aditya Spring 2022

[†]TAs: Abhilash Nandy, Ankan Mullick, Neeraj Saini, Ravi Pratap Singh, Vaibhav Saxena

Project 1: Enhanced Query-based product retrieval with the help of extraction of key phrases and vague phrases in the query (proposed by Abhilash)

Domain: E-commerce/English/Text

Description: The **dataset** published by Papenmeier et al. 2021 has annotated key phrases and vague facts per natural language query. Steps - (1) annotation of a few queries towards product retrieval from Amazon, and then bootstrapping. (2) Incorporating the key and vague phrases in product retrieval, and comparing it with normal baselines

Library: pytorch, simpletransformers, huggingface

Reference:

1. https://andrea.papenmeier.io/res/papers/2021_CHIIR_Dataset.pdf
2. Bootstrapping - https://aclanthology.org/2020.ecnlp-1.7.pdf

Project 2: E-commerce Clothing Product Categorization with limited data (proposed by Abhilash)

Domain: E-commerce/English/Text and images(Multimodal)

Description: The task contains matching images with categories (that belong to a taxonomy). For example, Clothing > Pants > Jeans is a hierarchy and you may have images associated with it. The dataset is available along with well performing models. However, the task is to train models with limited amounts of data.

Suggestions - (1) Checking out contrastive learning-based self-supervised methods that potentially reduce the amount of data required (2) Utilizing well-known taxonomies concepts to enhance the models (ConceptNet/YAGO).

The task can be of two forms. One can input an image and retrieve a serialized categories sequence; or otherwise, one can query a serialized sequence of categories and retrieve a list of images.

Library: Transformers, pytorch

Reference:

1. https://sigir-ecom.github.io/ecom2020/ecom20Papers/paper9.pdf
2. https://github.com/vumaasha/Atlas

Project 3: Content-based Image Retrieval System (proposed by Neeraj)

Domain: Image search using image

Description: This task gives us a flavor of the reverse image search, where one can use certain images to find related images. Reverse image search has many applications such as in fashion industries. The dataset and baseline methods can be found in https://arxiv.org/pdf/2002.07877v1.pdf .

Library: Transformers, Pytorch

Reference:

1. https://github.com/abhinav23dixit/Text-and-Content-Based-Image-Retrieval

Project 4: Cross-lingual Information Retrieval (CLIR) (proposed by Ravi)

Domain: Queries in English and 7 European languages

Description: A CLIR system usually includes two steps, the first step is the translation step, which includes translating either the queries into the language of the document collection, or translating document collection into the query language. After translation is done, the task can be reduced into a monolingual IR task.

However, current multilingual models (such as mBERT, XLM-R) embeds many languages into the same vector space and allows for cross-lingual semantic similarity. There are multiple possibilities. Comparing the methods using translation vs., using semantic similarity matching using mBERT is a start. The question here would be how to improve models and make it more resource-efficient, while maintaining the state-of-the-art accuracy (or F1 scores) for retrieval.

Libraries: transformers, bert, multilabel-classification

References:

1. https://aclanthology.org/2020.acl-main.613.pdf
2. https://github.com/suamin/multilabel-classification-bert-icd10

Project 5: Evidence Retrieval for Fact Verification (proposed by Neeraj)

Domain: Wikipedia/En

Description: Fact or claim verification is a two-step process. First, you retrieve supporting or refuting evidence related to a claim. Then based on the set of evidence snippets, the task is to determine whether the claim is true or false. In this project, we are interested in the first step, i.e., evidence retrieval.

Dataset: FEVER Dataset

References:

1. https://arxiv.org/pdf/1908.01843.pdf
2. Baseline: https://github.com/thunlp/GEAR

Project 6: Offensive query detection (on reddit/Twitter dataset) and generalization to multi-lingual setting (proposed by Vaibhav)

Domain: Social Media/En (and other languages)

Description: Offensive queries have become an important avenue for search engine companies. Often new socio-political events trigger new searches, which if shown as suggestions, can be deemed offensive to the users. Offensive text detection and generalizing to multilingual settings, hence, is of high relevance to many companies.

In the below datasets, the task is to propose an approach to automatically classify the tweets into 3 classes : hateful, offensive and clean. Test and compare various models for Hate-Speech detection on basis of Precision, Recall and F1 score. Most importantly, show how similar methods can generalize to multilingual settings. Here, we expect you to propose methods that do not use multilingual models (or ULMs such as XLM-R and mBERT).

Dataset: https://github.com/sayarghoshroy/Hate-Speech-Detection , https://github.com/mohit19014/Hindi-Hostility-Detection-CONSTRAINT-2021 , https://github.com/renuka-fernando/sinhalese_language_racism_detection

References:

1. https://arxiv.org/pdf/2010.12472.pdf

Project 7: Efficient and Fast Image Retrieval (proposed by Ravi)

Domain: Images and Text

Description: Regarding content-based image retrieval, it is often claimed that visually similar images are clustered in this feature space. However, there are two major problems with this approach: 1) Visual similarity does not always correspond to semantic similarity. 2) The classification objective does not enforce a high distance between different classes, so that the nearest neighbors of some images may belong to completely different classes.

Hierarchy-based semantic embeddings overcome these issues. The task is to learn semantic embeddings(more details are provided in paper) using the CIFAR-100 dataset.

Variation: An interesting **extension/variation** to above is utilizing the semantic embeddings; how do we create an indexing strategy which makes query processing fast and efficient. You may look at https://towardsdatascience.com/billion-scale-semantic-similarity-search-with-faiss-sbert-c845614962e2 and https://blog.vespa.ai/billion-scale-knn/ for reference.

Libraries: CNN, clustering

Dataset: https://www.cs.toronto.edu/~kriz/cifar.html

References:

1. https://arxiv.org/abs/1809.09924
2. https://github.com/cvjena/semantic-embeddings

Project 8: Efficient API/Code Snippet Retrieval (Somak)

Domain: Code-snippets and text

Description: Imagine, rather than waiting for answers in stack-overflow; you can query using natural language and the search engine comes up with a plausible code snippet. This facility can help ease many simpler tasks, which often a professional programmer needs to re-do. Such motivation has motivated the PL (programming languages) and ML community to come together and propose Deep neural-net based code search; which has seemingly become quite efficient. Here, based on the dataset, given a query task is to retrieve the most relevant code snippet.

Dataset: https://conala-corpus.github.io/ (data should be used in a retrieval setting)

References:

1. https://arxiv.org/pdf/2008.12193.pdf
2. https://people.eecs.berkeley.edu/~ksen/papers/ncs.pdf
3. https://github.com/nokia/codesearch

Project 9: Query-by-Example for Scientific Article Retrieval (Somak)

Domain: Scientific Publications/En

Description: Using background/objective, method or results as queries, the task is to retrieve most similar scientific papers. The dataset CSFCube contains an annotated test set for validation.

Dataset: https://arxiv.org/pdf/2103.12906.pdf, https://github.com/iesl/CSFCube

Project 10: Toponym Detection and Disambiguation in Scientific Papers (proposed by Ravi)

Domain: Scientific Papers/En

Description: Sentence boundary detection is an important task in NLP and may play an important role in semantic information retrieval. The dataset contains necessary annotation for detection and disambiguation subtasks as mentioned in the competition: https://competitions.codalab.org/competitions/19948#learn_the_details-task_details.

Project 11: Entity Extraction to help efficient Retrieval (Somak, inspired by Ankan's idea)

Description: The COVID-19 open research dataset was proposed by AI2 to facilitate research progress by processing COVID-related scientific articles faster. An interesting outcome of this was creating the SciBERTNLI model (and the SPIKE-CORD project). Such models were used for efficien semantic retrieval from this repository (as well as extractive summarization). The task here should investigate how entity extraction or entity tagging may increase retrieval efficiency. For biomedical entity tagging, you can look at tools such as https://github.com/strayMat/bio-medical_ner (YASET).

One possibility is to explore creation of a knowledge graph using such a tool and use KG embedding techniques alongwith SciBERT methods to retrieve efficiently.

This task may require annotation of certain articles for validation.

Project 12: Information Retrieval in a Code-mixed context (Somak)

Description: MSR India researchers recently published a large collection of tasks in different code-mixed languages under the umbrella of GLUECoS. Here, one can attempt to solve QA task in a limited resource scenario as a retrieval problem. Say, given a question and a paragraph (context), rank and retrieve relevant sentences. Track Recall@K, Precision@K (K=1, 5, 10). We mark a sentence correct if the annotated answer is present in the sentence

The challenge again here is to avoid large Universal Language models (ULMs), as student groups may not have such resources. How do we build plausible intuitive ML systems that can process such code-mixed data, and yet be competitive.

Library: pytorch, simpletransformers, huggingface

Reference:

1. https://aclanthology.org/2020.acl-main.329.pdf

Project 13: Entity Retrieval from Scientific Articles (Ankan)

Description: NLP/ML scientific articles often mention various entities, such as the programming language (Python, Matlab etc.), used libraries (Spacey Model, NLTK, Huggingface etc.), frameworks (Google Colab, Kaggle Notebook, Nvidia GPU Server.). Similarly, deep learning papers tend to contain details about parameters (Optimizer, Epoch etc.), model hyperparameters (Threshold Value etc.), and metrics (Precision / Recall / Accuracy / Mean Absolute Error etc.). A previous work (namely DLPaper2Code [1]) extracted such structures to automate generating the DL training code from a paper. This is hard for generic scientific articles. However, we can build a QA or a retrieval model to extract such entities and report Precision / Recall (Entity-wise and overall)

Dataset : Crawled Dataset of NLP / ML conference papers of the last 10 years.

This task will require annotation of certain articles for validation.

Library: pytorch, simpletransformers, huggingface

Reference:[1] https://arxiv.org/pdf/1711.03543.pdf, [2] https://arxiv.org/pdf/2009.06819.pdf