Image Recommendation for Wikipedia Articles

Thesis · January 2020

DOI: 10.13140/RG.2.2.17463.27042

2 authors, including:

Oleh Onyshchak
Ukrainian Catholic University
6 PUBLICATIONS **0** CITATIONS

SEE PROFILE

Ukrainian Catholic University

Master Thesis

Image Recommendation for Wikipedia Articles

Author:
Oleh Onyshchak

Supervisor:
Miriam Redi

A thesis submitted in fulfillment of the requirements for the degree of Master of Science

in the

Department of Computer Sciences
Faculty of Applied Sciences

Lviv 2020

Declaration of Authorship

I, Oleh Onyshchak, declare that this thesis titled, “Image Recommendation for Wikipedia Articles” and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

UKRAINIAN CATHOLIC UNIVERSITY

Faculty of Applied Sciences

Master of Science

Image Recommendation for Wikipedia Articles

by Oleh Onyshchak

Abstract

Multimodal learning, which is simultaneous learning from different data sources such as audio, text, images; is a rapidly emerging field of Machine Learning. It is also considered to be learning on the next level of abstraction, which will allow us to tackle more complicated problems such as creating cartoons from a plot or speech recognition based on lips movement.

In this paper, we will introduce a basic model to recommend the most relevant images for a Wikipedia article based on state-of-the-art multimodal techniques. We will also introduce the Wikipedia multimodal dataset, containing more than 36,000 high-quality articles.

Acknowledgements

I wish to express my sincere thanks to:

- Miriam Redi for mentoring the project, giving valuable feedback, suggesting possible solutions to all the problems, and for cheering up along the way. It was a pleasure working together.
- Dmytro Karamshuk, who went out of his way to introduce me to Miriam, and thus making this cooperation possible.
- Jianfeng Dong, who made source code for his Word2VisualVec model[7] publicly available and well-documented, which helped to speed up the research process significantly.
- Irynei Baran for helping us to keep the pace of research because of regular Master's seminars.
- Vadim Ermolayev, who was one of the initiators of the Masters Symposium, which put me on track and allowed to get the valuable feedback early on.
- Oleksii Molchanovskyi, who was one of the creators of the Master's program in Data Science at UCU and still keeps developing it. Only because of this program I was able to enter the field of Data Science.
- All lecturers, staff, and donors of Data Science Program, who made my studying of Data Science and thus researching this project possible

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Domain Overview

Every day we perceive the world around us through multiple cognitive feelings such as sight, smell, hearing, touch, taste. Moreover, our ability to consolidate all the information from different sources into one complete picture helps us comprehensively understand the world.

With a trend to digitizing in the last few decades, more and more information is recorded in different kinds of media such as audio, image, video, text, and 3D modeling. That also created new challenges of efficiently processing significant amounts of recorded information, where we already have significant achievements. However, every type of digital storage only captures some subset of available information. For example, imagery only captures visual appearance, while audio - the sound, just as our eyes and ears do. Thus all the scientific progress in processing some data carrier is bounded by limitation of what that medium can capture. In other words, to represent a dog digitally, we have to have more than just a visual representation. Similar to humans, we need to combine all the information streams, which describe the same entity from different perspectives, into one comprehensive representation.

That is the motivation for multimodal representation learning, which aims to combine different types of data into a complete representation of a real-world entity. In that context, the word "modality" refers to a particular way of encoding information. Thus a problem in the domain of e.g., image processing is called unimodal, while a problem in the domain of multiple information encodings, for example image to caption generation, is called multimodal since it works with both image and text modalities [12]

By having a complete representation of an entity, which was created via multimodal data that captures complementary / supplementary information subsets of an object, we have more comprehensive computational "understanding" of that entity. That helps us to increase the precision of existing data science applications, and extend the limits to more abstract problems such as not only identify the objects in an image but understand the value. For example[12], early research on speech recognition showed that by involving visual modality of lips movement on top of sound modality, we get extra information that allows us to increase the quality of voice recognition task, just as it works for humans[24]

1.2 Problem Motivation

Wikipedia is the biggest collection of human knowledge containing more than 35 million pages and having nearly 9 billion views per month[1] And it continually growing, having more than 500 new pages per day[2], and all of that only in its English version.

As a part of 2030 strategy, one of the key goals is to break down any barriers for accessing free information[3]. By researching possibilities to automatically recommend images for Wikipedia editors, it will help to get better media enrichment of articles, which in turn will make information easier and faster to comprehend[36]. Also, it would be helpful as automation of time-consuming task to search for and add a proper article visualization.

In addition to motivation of making Wikipedia better, this work might present some useful insights to development of multimodal learning field. Since this is:

1. purely real-world problem, which might give us interesting insights of how to apply and adjust current academia progress

2. we have more complicated problem settings of one extensive article corresponding to multiple images, instead of a more simplified one-to-one relationship of images and their tags/descriptions

1.3 Problem Formulation

We are going to research how state-of-the-art multimodal learning techniques performs on a task of recommending images for Wikipedia articles. In other words, having a text with wiki formatting, we need to rank images from Wikimedia Commons database[38] by relevance.

That is, based on the article's text information, we need to recommend images describing the same notion. In other words, we need to create a high-level representation of some entity, described by both text and images. So that we can "understand" which image representation of the notion is the best suited for a given text description.

This high-level task consists of two main subtasks

1. collecting a multimodal dataset of Wikipedia articles

2. adjusting the state-of-the-art model to work on our real-world data

Dataset Collection

We need to collect a dataset which will have article's text content as well as all images associated with it. We will also need to include some of the useful metadata Wikipedia contains, such as image description.

[1] https://stats.wikimedia.org/v2/#/en.wikipedia.org
[2] https://en.wikipedia.org/wiki/Wikipedia:Statistics
[3] https://meta.wikimedia.org/wiki/Strategy/Wikimedia_movement/2017/Direction

Model Adjustment

We will take Word2VisualVec[7] multimodal model and apply it on our Wikipedia dataset. To do so, we will need to experiment how to process our data, and in what form to include it into the model. Then we will need to compare its performance with our baseline, which we will introduce later, and make a conclusion about the applicability of multimodal approaches to Wikipedia Image Recommendation problem.

1.4 Project Contribution

We made the following contributions, which, as we believe, will be valuable from both a research and an application perspectives.

- collected a dataset of multimodal Wikipedia articles of high quality. That is the first complete multimodal dataset of high-quality Wikipedia articles. We will describe how it differs from similar one in Chapeter 2
- adapted Word2VisualVec[7] to our real-world problem
- additionally developed text-based image retrieval in order to combine with our adapted Word2VisualVec[7] model to get the best performance

1.5 Thesis Structure

- Chapter 2, Related Work: here we will overview existing approached in multimodal domain and select the most appropriate for our problem. Then we will choose some specific model, which implement that approach, in order to apply it on our problem. We will also review specific researches made in this field with Wikipedia data and describe how our work differs to them.
- Chapter 4, Problem Approach: in this chapter we will describe our model in details, describing its architecture, feature extraction techniques and evaluation metrics. We will also specify how we adjusted our data to work with this model and describe the baseline for comparing with our model's results.
- Chapter 3, Data: here we will describe all details about dataset collection and processing. Specifically, we will tell 1) what articles we collected for our dataset, 2) how we ensured dataset quality, 3) details on how we collected text, images, and metadata; 4) how we cleaned and additionally processed our dataset; 5) and also how to it can be downloaded and 6) how collection process can be reproduced
- Chapter 5, Experiments: in this chapter we will describe what experiments were performed and will analyse their results, identifying our best model. We will also compare our model to the baseline and provide inferences on each experiment and our model overall. Additionally, we will present the output of the best-performing model and provide our hypothesis regarding what made the model perform strongly/poorly on specific examples.

- Chapter 6, Conclusions: here we will sum up all the work which was done in scope of this project and provide conclusions on models performance and dataset value. We will also specify what future improvements are still planned and give an overview what further work is possible in scope of this project.

Chapter 2

Related Work

2.1 Overview

While during the last decades there was much progress in a field of unimodal representation, research in multimodal learning was moslty limited by simple concatenation of unimodal features[6]. However, during recent years, the scientific landscape in this domain has been rapidly evolving[1]. One of the triggers for it was the success of deep learning models, which have a powerful representation ability with multiple levels of abstractions. Thus they were also incorporated in multimodal learning. As Guo et al. suggested[12], we can divide all the multimodal learning approaches into three categories:

1. joint representation, which aims to integrate modality-specific features into some common space
2. coordinated representation, which aims to preserve modality-specific features, while introducing a space to measure multimodal similarities
3. intermediate representation, which aims to encode features of one modal to some intermediate space, from where we later generate features of another modal.

In this chapter, we will cover available techniques to extract features from text and image modalities, overview available solutions in each type of multimodal learning, and then summarise their applicability for our problem. We will also review specific works which address similar problem.

2.2 Approaches Review

2.2.1 Unimodal Representation

Image

The most popular model used in feature extraction from images are different types of Convolutional Neural Network(CNN), such as AlexNet[21], VGGNet[32] and ResNet[15]. When working with big datasets, it is preferable to use pretrained version of chosen CNN. This field has tremendous development in recent years, and thus currently we already have well-defined solution for most problems.

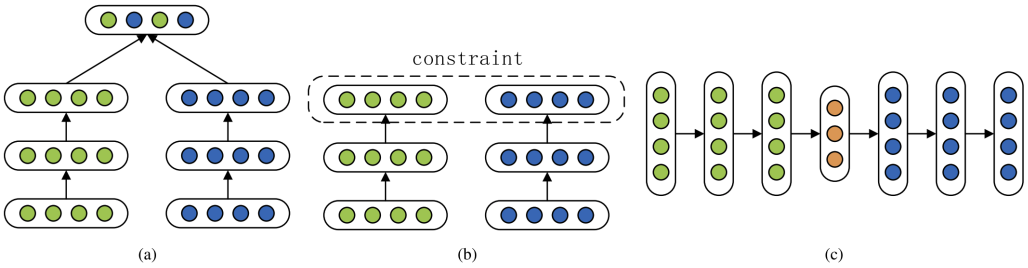

Figure 2.1: Three types of frameworks about deep multimodal representation. (a) Joint representation aims to learn a shared semantic subspace.(b) Coordinated representation framework learns separated but coordinated representations for each modality under some constraints. (c) intermediate representation framework translates one modality into another and keep their semantics consistent.[12] © 2019 IEEE

Text

A popular way to extract features from the text is to encode it to vector, as is done in word2vec[25] or Glove[28] algorithms. They map words into one-hot encoded vector space of language vocabulary. Although, the common problem with those approaches is when some words are not present in vocabulary or out-of-vocabulary error. However, there are also a variety of solutions to this problem, such as character embeddings[20].

An alternative and more powerful tool for dealing with text is recurrent neural network(RNN)[8], which is more context-aware and can make better encoding of the n-th word, knowing what was already in a sentence. One of the most successful realizations of RNN is long short-term memory(LSTM)[17].

2.2.2 Joint Representation

The main idea of joint representation is to integrate multimodal features into a single input, which we then process as some artificial unimodal input with well-known machine learning techniques. More formally, it aims to project unimodal representations into a shared semantic subspace, where the multimodal features can be fused[1], as shown in Figure 2.1(a). Up until recently, that was the primary technique in multimodal learning, where shared features were fused by concatenating them together. However, now, the most popular choice is to use a distinct hidden layer, where modality-specific features will be combined into a single output vector.

This approach was historically the first one and is still commonly applicable in video classification[19], event detection[13] and visual question answering[10]. However, its main disadvantage is neglecting the fact that different modalities have not only supplementary information, that is which show the same notion from different perspectives, but also complementary information, where one modality captures the information which another cannot. For example, lips movement and audio of a speech are mostly supplementary sources, while images

of some bird and audio of it singing are mostly supplementary sources. Because of that, much information gets lost in that shared space.

Although it has advantages of being a simple method and producing modality-invariant common space of features, it cannot be used to infer the separated representations for each modality[12]. Thus methods from this category are not applicable to our problem

2.2.3 Intermediate Representation

Intermediate Representation models aim to encode features of one modality to some intermediate space, from which later features of another modality can be generated(or decoded), as shown on Figure 2.1(c). To prevent the intermediate space from being related only to a source modality, during encoder-decoder training we maximize, e.g., the likelihood of target sentence given source image, so that error function employs the error of decoding. Subsequently, the generated intermediate representation tends to capture the shared semantics from both modalities[12].

Some interesting application of that model was proposed by Mor et al.[26], where algorithm encodes a musical track into intermediate space, which then will be decoded by multiple decoders into a space of some specific instrument. In other words, encoder extracts instrument-invariant generic musical features, which then each decoder transforms into features of its target instrument.

The general advantage of such approach is that it is one of the best ways to generate new features in a target domain. Thus this technique is used in Image Caption[35], Video Description[34], and Text to Image[31] generations. The disadvantages of that model are that 1) it can only encode one modality, 2) complexity of designing a feature generator should be taken into account[12] and 3) intermediate space also extracts only shared subspace from two modalities. Moreover, because we need to query existing information rather than generate one, those methods are also not suitable for our problem solution.

2.2.4 Coordinated Representation

The last type of multimodal learning is a coordinated representation. Instead of learning from a joint representation, it learns from modal-specific representations separately but with a shared constraint, which is some loss function identifying cross-modal similarity/correlation. Since different modalities hold unique information about an object, that approach operates with all available knowledge. A visual explanation can be seen in Figure 2.1(b). Regarding constraint function, a commonly used option is cross-modal similarity functions, where learning objective is to preserve both inter-modality and intra-modality similarity structure. In other words, it would force cross-modal distance for elements with the same semantics be as small as possible, while with dissimilar - as big as possible.

The cross-modal ranking is a widely used constrain, where the loss function is defined in the following way

$$\sum_{i}\sum_{t^-} max(0, \alpha - S(i,t) + S(i,t^-)) + \sum_{t}\sum_{i^-} max(0, \alpha - S(t,i) + S(t,i^-)) \qquad (2.1)$$

where (i,t) is a matching image-text pair, α is margin, S is a similarity function, i^- is mismatching pair to t and vise versa. Frome et al.[9] used a combination of dot-product similarity and margin rank loss to learn a visual-semantic embedding model(DeViSE) for visual recognition[12]. DeViSE trains deep networks for both image and text features, and then adjust features based on above mentioned ranked loss, though in more simplified form.

Alternatively to cross-modal ranking, another widely used constraint is Euclid distance, which is also used for ensuring that similarity structure for both intra-modality and inter-modality is preserved. That is, for inter-modality, we map text and image features into low-dimensional space, where we can calculate the distance between feature vectors. The idea here is to ensure that inter-modality features of the same semantics are as close as possible[27]. While for intra-modality, we want to preserve the similarity between neighborhood items, that is:

$$d(m_i, m_j) + m < d(m_i, m_k), \forall m_j \in N(m_i), \forall m_k \notin N(m_i) \tag{2.2}$$

where m is data point of any modality, m_i point of interest, N(m) - denotes neighborhood of m[37].

So, Coordinated Representation preserves all modality-specific information. It also explicitly compares features from different modalities, thus having data from one, we can identify the closest data point from another modality. Because of those properties, it is used for cross-modal retrieval[37], retrieval-based visual description[33], and transfer knowledge across
modalities[27]. Thus it can be applied for our problem of Image Recommendation for articles, and we will proceed with those methods.

2.3 Work Review

A similar problem was researched by Rasiwasia et al.[30] in 2010, where a cross-modal retrieval model for Wikipedia articles was designed. For that purpose, they also collected a multimodal dataset of subset of featured articles[1] with corresponding images. Then, using a derived method from latent Dirichlet allocation model[2] as text features and scale-invariant feature transformation(SIFT) descriptors model[22] as image features, they performed a correlation analysis of features in text and image modalities in order perform cross-modal retrieval.

We are going to collect multimodal Wikipedia dataset as well but on a bigger scale. That is, it will include not only all featured but also good[2] articles. We will also additionally collect metadata for each image such as its title and description so that our model would have more data to train on. Additionally worth noting that in ten years the size of featured articles collection itself becomes two times bigger, which justifies its updating.

From the model perspective, there was a significant shift of state-of-the-art approaches to a variety of problems since the time of the paper because of the popularity of deep neural network models. Thus we will heavily leverage them for feature extraction in our work.

[1] https://en.wikipedia.org/wiki/Wikipedia:Featured_articles
[2] https://en.wikipedia.org/wiki/Wikipedia:Good_articles

More recently, Huang et al.[18] also made a research using Wikipedia multimodal dataset introduced in [30]. They proposed a Cross-modal Hybrid Transfer Network, for addressing the problem of knowledge transfer from a single-modal source domain to a cross-modal target domain. While they work is also in the scope of multimodal domain with Wikipedia dataset, we have a different goal of developing a model for cross-modal retrieval.

Another relevant work was done by Hessel et al.[16]. They collected a vast Wikipedia multimodal dataset of 192K most popular article, where the condition was at least 50 views on the date of collection for article to be collected. They also presented the algorithm for automatically computing the visual concreteness of topics within a multimodal dataset. In our work, we will collect the dataset based on quality rather than popularity because the high-quality dataset is paramount for our project. Here as well we will solve a different problem though in similar settings. While they were assessing visual concreteness of different Wikipedia topics, we are trying to match particular text to relevant images.

Another interesting paper was recently published by Dong et. al[7] where they developed a cross-modal retrieval model. That Word2VisualVec model combines a variety of state-of-the-art approaches to extract text and image features, heavily leveraging deep neural network approaches. The model showed impressive results on Flickr[29] dataset, where every image is associated with five crowdsourced descriptive sentences. In our work, we will reuse architecture of Word2VisualVec and apply it on our newly collected multimodal Wikipedia dataset. We want to asses how good will this model perform in the more complicated real-world setting of our Wikipedia problem. We will also amend the model to fully exploit available Wikipedia metadata for each image in order to improve its precision.

Chapter 3

Data

3.1 Overview

We need to collect and adequately preprocess a dataset with articles and relevant images in order to train our model. While the proper multimodal dataset of high-quality Wikipedia articles does not exist, all Wikipedia data is publicly available, so we have a way to collect it on our own.

Specifically, we have almost 6 million Wikipedia pages[1] and Wikimedia Commons image dataset[38] contains more than 57 million images[2]. That is the real-world data, where, ultimately, the solution should be applied.

3.2 Article Selection

The enormous growth of Wikipedia is caused by its business model of crowd-sourcing article editing. Although, this is also the reason why it can contain incomplete or even false information[3]. Thus we need a way to identify whether an article is of high quality in order to collect a useful dataset for our task.

Fortunately, Wikipedia has notions of 1) good articles[4], which contain only high-quality and well-illustrated articles, and 2) featured articles[5], which contain articles of exceptional quality, one tier better than good articles. As of the time of writing, there are more than 30K good articles with 159K images and more than 5,500 featured articles with 57K images out of almost six million available at English Wikipedia. In other words, only about 1 in 150 articles is of good enough quality to become a "good article", although it still leaves us with plenty of data to train on. Please note, that image numbers mentioned above might contain some duplicated or not publicly available entries. Thus the real number of unique images is somewhat smaller. For example, for featured articles, there are 45K of unique publicly available images out of 57K associated with them.

Each page in either category goes through a thorough manual review procedure by the Wikipedia community and represents the best Wikipedia can offer. All together it leaves us with a significant dataset of manually selected articles with a theoretically the best possible quality for machine learning algorithms.

[1] https://en.wikipedia.org/wiki/Special:Statistics
[2] https://commons.wikimedia.org/wiki/Special:Statistics
[3] https://en.wikipedia.org/wiki/Reliability_of_Wikipedia
[4] https://en.wikipedia.org/wiki/Wikipedia:Good_articles
[5] https://en.wikipedia.org/wiki/Wikipedia:Featured_articles

3.3 Collection

Source code for dataset collection is available at:

https://github.com/OlehOnyshchak/WikiImageRecommendation/tree/master/article_reader

First of all, we need to obtain a list of all articles of our interest. Wikipedia API[6] allows us to list all pages of a specific category, which we have used. Then we will process the list with pywikibot[7], which is a convenient python wrapper around Wikipedia API.

More details regarding how to download and work with the collected datasets can be found in the Appendix A.2

3.3.1 Text Collection

Having a list of articles, we can download its underlying wikitext[8]

3.3.2 Image Collection

For each page, we will retrieve a list of all image handlers and then download each of them in a unified fashion. That is, all images will be of JPEG type with width equals 600 pixels, while the height will be adjusted to preserve the aspect ratio. In that way, we will be able to work with images uniformly without losing any information as well as reduce the size of dataset significantly. Wikipedia API allows downloading images specifying restrictions mentioned above in the query itself so that all transformation will be performed on the server-side and we will pass through network only efficiently-sized resulting images.

Along with image, we collect some useful metadata Wikipedia supply us, such as image title and description. And while title can be trivially queried, for description we need to scape Wikimedia HTML page for each image and then parse out its description.

Although, some images used in Wikipedia does not exist in Commons dataset. The common reason for the is that image copyright protection allows its usage only for specific pages and are not generally free. Those images are also mentioned but their raw data is unavailable due to above mentioned constrained.

3.4 Preprocessing

Before working with our collected data, we also need to do some additional preprocessing, which are described in details in the following sections.

3.4.1 Text Cleaning

Since wikitext auxiliary markup mostly used to specify rendering details, which is irrelevant for our problem, and links to other pages, which is not currently

[6] https://www.mediawiki.org/wiki/API:Main_page
[7] https://www.mediawiki.org/wiki/Manual:Pywikibot
[8] https://en.wikipedia.org/wiki/Help:Wikitext

used in the scope of this research, only core text is extracted with the help of MWParserFromHell library[9]

3.4.2 Image Cleaning

Each article also uses standard icons to identify some Wikimedia resource or to identify that it is a good or featured article. Such images are not related to the content of any particular page so we need to purge them to avoid noisy information in the dataset. As the most trivial way to do so, all image with "SVG" were removed. Beware that it also removes some useful images like country flags, but it is a very small chunk of all removed images, thus it should not affect overall performance.

We also use a RedditScore[10] word tokenizer on the image title to extract meaningful terms from the image title. That is, as we discovered during experiments, image title commonly consists of a few words joined together without spaces, e.g. "helloworldjpg". Then word tokenizer, based on natural word frequency, parses this title in the most probable sentence, which would be "hello world jpg". Then we additionally remove any mentioning of image extensions, which results in "hello world" parsed title.

3.4.3 Storing Computed Features

Our model should learn the mapping from text to visual feature space. Since the extraction of image features is computationally expensive and also images itself occupy far more space than resulting features vector, we extract them once and save for further usage. As features vector the output of last hidden fully connected layer of ResNet152[15] trained in ImageNet was taken. That output is later max-pooled to a vector of 2048 elements, in order to be compatible with Word2VisualVec model, which we use as a core of our solution. That optimisation allowed to save a dozen hours of training time as well as reduced dataset size by more than ten times.

[9] https://mwparserfromhell.readthedocs.io/en/latest/
[10] https://github.com/crazyfrogspb/RedditScore

Chapter 4

Problem Approach

4.1 Overview

After the overview of related work, Coordinated Representation approach was identified as the most prominent direction for our problem's solution because it aims to exploit modality-specific features fully.

We will focus on integrating recent Word2VisualVec model[7] to our more broader and more realistic problem settings. It showed impressive results but was evaluated on a more narrow problem. More specifically, it was working with Flickr dataset[29] where one image corresponds to 5 descriptive sentences. In our settings, we have one article corresponding to multiple images, where all of them having additional metadata such as category, name, description.

In Coordinated Learning, the general pipeline is to discover correct feature representation for each modality, while knowing how to map them into some common space. Word2VisualVec model[7] solves opposite problem, which is far more computationally efficient. That is, we fix some feature representation for each modality and learn how to map them into common space correctly. Moreover, since the task is to identify images with text, we simplify the model even further by mapping directly from the text to image space.

Although, the more simple model comes in expense of its quality. So here we might lose precision by assuming that existing pretrained feature extractors will represent our data for Image Recommendation task in the best possible way. But when making proof of concept model, this possible loss is acceptable since with more simple model we can do far more experiments.

This paper is accompanied by Github repository with all experiments

https://github.com/OlehOnyshchak/WikiImageRecommendation

4.2 Challenges of our Real World Scenario

The task of this project becomes testing the feasibility and improving Word2VisualVec in the practical scenario of recommending images to Wikipedia articles.
Word2VisualVec was originally trained on a much simpler task, namely caption retrieval for still images. The original dataset is made of 30K Flickr imagest[29], and each image is associated to five crowd-sourced descriptive sentences. Caption sentences describe generic actions of objects such as "dogs" or "mountains" in the image. The model is trained to associate an image to one of the five captions.

Our Wikipedia image recommendation task poses the following challenges:

- Semantics: in a generic caption retrieval task, the model learns associations between text and generic visual objects, for example "cars". Here we need the system to be able to capture the fine-grained semantics of an article, e.g. we want to be able to retrieve images related to the concept "Maserati", rather than to a generic concept "car". This can be solved partially by training on entity-specific data. More importantly, considering image metadata is crucial to solve this challenge.

- Retrieval: in a caption retrieval task, an image needs to be associated to one or more sentences. In our Wikipedia image retrieval task, we want to retrieve one or more images that should be assigned to one article, which is made of multiple sentences. In this scenario, we want not only to adjust the evaluation metric to reflect the goal of our task, but also carefully evaluate how we represent the notion of "article": are we looking at article's title, summary, or the whole text?

4.3 Architecture

Now we will describe each part of the model in details, which is also visualised on Figure 4.1

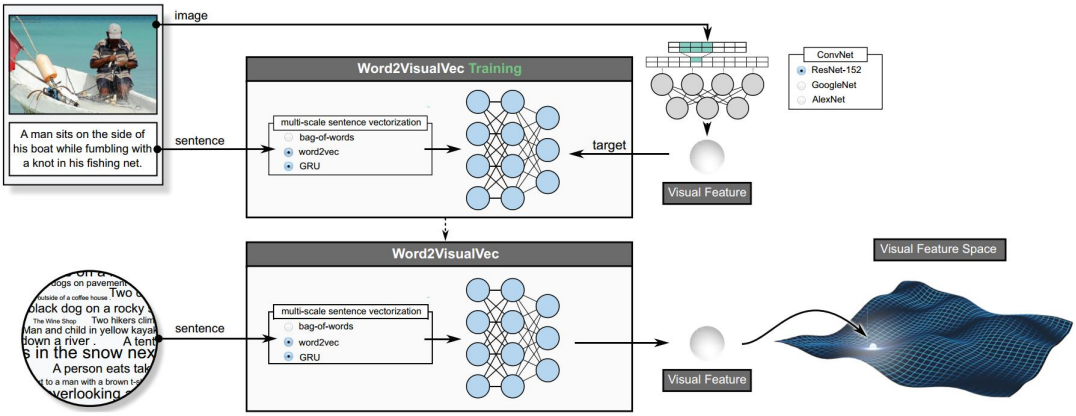

Figure 4.1: Word2VisualVec network architecture[7] © 2018 IEEE

4.3.1 Image Features

For image representation we will use the output of last hidden fully-connected layer of ResNet-152[15] pretrained on ImageNet dataset[5]. The motivation for this choice was that ImageNet is the biggest general-purpose image dataset and ResNet is one of the recent and popular deep convolutional neural networks. The biggest depth of 152 layers was taken in order to extract the most information.

Then ResNet-152 output is mapped to vector with size equals 2048, in order to compare it with Word2VisualVec's output. While we might lose some information here, we benefit from our simplistic proof of concept model.

4.3.2 Text Features

For text representation, several state-of-the-art models were applied together, which was experimentally the best option for the task[7]. The features from the following models were jointly used as a text representation:

- Bag of Words[14]: represents text in a space where each dimension is a specific word, and its values correspond to how many times the word occurred in a text. Although, it is limited by its vocabulary size, which is, in turn, limited to a training dataset. That is, if we do not have an image representation for some words, those will not contribute any information to text mapping into visual space.
- word2vec[25]: represents text in a space where contextual similarity of words is preserved. That is, given a large text corpus, word2vec assign unique vector to each word so that words often appearing in the same context are located close to each other in a mapped vector space. This model compensates for the Bag of Words lack of semantic relationship in a mapped space.
- Gated Recurrent Unit[3]: represents text with respect to a relative order of words appearing in a text. That is, it additionally exploits the information hidden in relative order of words.

Then we glue those fixed-size representations together and let model pick up what is the most relevant.

4.3.3 Text to Image mapping

The mapping model itself is a simple neural network with three hidden layers, each having a ReLU[11], as activation function, followed by a Dropout layer. As authors discovered, it is as deep as it can become without losing generalisation power.

4.4 Baseline

As a baseline for our model on our dataset, we will take real-world alternative to this project - text similarity model. That is, having article data we will look for relevant images with text-based search within image metadata. That resembles how it is usually performed manually nowadays: by querying with the article title, we get images ranked based on co-occurrence of words between query and image description.

We will investigate a few models which enable us to identify similarity between different text description such as: word2vec[25], inferText[4], wikipedia2vec[39]. Those techniques map words into model-specific vector space, where words with similar semantics are mapped close to each other and vice versa. For example, word2vec builds its space by investigating which words often appear in the same context, while wikipedia2vec additionally consider cross-references between different Wikipedia pages.

We will also develop our implementation of simple word co-occurrences similarity model, which directly provides similarity scope for two text input based on have many times words from one input appeared in another input. That is, the similarity of "hello world" to "hello my dear world, hello" will be equal three, since the latter has words from former input three times.

We believe that text-based model is partly complementary to our multimodal model, and thus we will experiment whether we can get better precision by combining two models later on.

4.5 Evaluation

4.5.1 Settings

In original work, each training pair consisted of image and five descriptive sentences, describing it. The idea is that all five different sentences describe the same entity, and thus should map into that entity representation in visual space.

In our settings, we have one article having a varying amount of included images. Each image, in turn, might have some additional metadata such as description. Thus for our experiments we will associate information from both article and metadata as a textual description of entities described on every image.

We will also consider two different approaches regarding train-test split dataset, that is:

- image-level split: with this approach we take all the images, associate them with their article and metadata text representation, and then split on train-test subsets. In this way images from the same article might appear in both subsets thus avoiding the situation of processing completely unseen fine-grained query. That is, when in training dataset we do not have any image of Albert Einstein there is a little chance that model will identify him correctly during testing. So while we still keep different images in train-test subsets, this approach helps the model to get some insight on different fine-grained queries

- article-level split: here we will first split all the articles into train-test subsets, and only then extract images from articles and associate them with textual representation. This is more complicated settings for our model but it better outlines the real-world performance. We believe that with significantly bigger dataset and with a leverage of text-similarity model, article-level precision will tend to the image-level precision.

4.5.2 Metric

For evaluating results, we process all textual descriptions of images from a test subset with Word2VisualVec[7] model. Internally, it firstly maps captions into a space of text features, and then it gets mapped into a visual space. Then the performance is evaluated based on caption ranking[23]. That is, for each mapped to visual space point, we rank all dataset images based on their similarity and report rank-based R@K (K = 1, 3, 5) precision. For example, R@3 shows the percentage of test pairs with the target image within top-3 ranked images for a mapped point in visual space.

Chapter 5

Experiments

5.1 Table Abbreviations

In this Chapter we will use the following abbreviations in tables to reports results of experiments.

Table 5.1: Table Abbreviations

Table Abbreviations		
	Abbreviation	Meaning
1	A.	article
2	I.	image
3	I. description	description of an image, if present. Otherwise, its title
4	I. description (parsed)	description of an image, if present. Otherwise, its parsed title. That is the title, which often has a few words glued together without spaces, was converted into separate words removed with image extension out of it
5	A. summary	first 1000 characters of an article. It is an approximation of article summary because extracting title precisely will require much not-trivial work

5.2 Training Details

All training was done on Kaggle[1] where environment is set up from scratch on each run. In other words, it should be easily reproducible with any hardware, since models run in cloud.

For training constant global parameters were as following 1) learning rate = 1e-4, 2) dropout = 0.2, 3) optimizer = RMSprop, 4) loss function = MSE, 5) similiarity function = cosine similiarity.

All the training was performed on subset of Featured articles, which are of the best quality.

[1] https://www.kaggle.com/jacksoncrow/w2vvtraining

5.3 Baseline Experiments

We have tried four different models with different text inputs such as article summary, article title, image description. As we can see, inferText and wikipedia2vec showed inferior performance in all cases. Word2vec was significantly better than previous competitors but, surprisingly, the simple co-occurrence model showed the best results. And thus we will take model nine with results equal to 4.9, 11.8, 25.7 as our baseline.

Table 5.2: Text-Similarity Experiments

Text-Similarity Experiments				
	Model	Query	Image Meta	Precision
1	word2vec	A. summary	I. description(parsed)	1.3, 2.5, 5
2	word2vec	A. title	I. title(parsed)	3.5, 10.3, 17.8
3	word2vec	A. title	I. description(parsed)	3.8, 9.4, 18.6
4	inferText	A. summary	I. description(parsed)	0.9, 1.4, 2.5
5	inferText	A. title	I. title(parsed)	2.9, 6.3, 12.9
6	wikipedia2vec	A. summary	I. description(parsed)	0.5, 1.5, 2.7
7	wikipedia2vec	A. title	I. title(parsed)	1.5, 3.1, 6.7
8	wikipedia2vec	A. title	I. description(parsed)	1.5, 3.2, 6.5
9	co-occurrence	A. title	I. description(parsed)	4.9, 11.8, 25.7

5.4 Word2VisualVec Experiments

Tables in this section would have columns abbreviation described in Table 5.3.

Table 5.3: Column Abbreviations

Column Abbreviations		
	Abbreviation	Meaning
1	Text Representation	what data was used to generate textual representation of an image
2	B	minimal number of times a word should appear in training corpus for it to be included in Bag of Words vocabulary
3	R	the output size of Gated Recurrent Unit, which is one of the models used to extract text features
4	E	number of epoch the model was train
5	Precision	caption ranking precision formatted as R@1, R@3, R@10

5.4.1 Image-Level Split

In order to identify what would make the best textual representation of an image, we held various experiments, as is showed in Table 5.4.

We tried a lot of different models in the first five experiments and discovered that using article summary with image description, which is the fifth model, significantly outperform alternatives. We then discovered that a lot of image titles,

Table 5.4: Image-Level Experiments

Image-Level Experiments					
	Text Representation	B	R	E	Precision
1	A. first sentence + A. title	5	32	10	3.6, 12.4, 19.3
2	A. first sentence	5	32	10	5.1, 16.0, 25.9
3	A. first sentence + I. description	5	32	10	8.3, 25.1, 37.8
4	A. first sentence + I. description	20	32	10	7.6, 23.8, 36.2
5	A. summary + I. description	5	32	10	11.8,29.8,40.9
6	A. summary + I. description (parsed)	5	32	10	13.9,31.9,42.7
7	A. summary + I. description (parsed)	5	100	10	7.2, 20.2, 28.3
8	A. summary + I. description (parsed)	5	32	24	18.2,38.4,47.2
BASELINE					4.9,11.8,25.7

which are used when description is not available, is a short descriptive sentence without spaces. For that reason, all our feature-extraction model from texts cannot recognise anything and map everything into zero-vector. By correctly parsing titles back into words, we were able to increase precision on one more level, as can be seen by comparing models five and six.

We also did a few experiments with changed model hyperparameters, as can be seen in experiments four and seven, but Word2VisualVec defaults performed the best.

In the end, we allocated more time for training to our best-performing model and got the final precision of 18.2, 38.4, 47.2, as is shown in model eight. In other words, this model can correctly identify the target image within top-10 results every second time, which significantly outperform our baseline. Please note, that model is commonly evaluated on fine-grained queries such as "Maserati" instead of general concept "car", which makes identification of the image far harder.

5.4.2 Article-Level Split

We chose the best-performing model from image-level experiments and tested it on the article-level split, as is shown in Table 5.5. We also did a few experiments trying to adjust model hyperparameters better but default one performed better here as well.

As expected, article-level model shows worse results, which is only around 60% of the precision of the image-level model with respect to R@10. And the best article-level precision is 8.4, 20.1, 29.6

Table 5.5: Article-Level Experiments

Article-Level Experiments					
	Text Representation	B	R	E	Precision
1	A. summary + I. description (parsed)	5	32	14	5.2, 14.6, 22.5
2	A. summary + I. description (parsed)	5	32	24	7.2, 18.0, 28.0
3	A. summary + I. description (parsed)	5	32	38	8.4, 20.1, 29.6
4	A. summary + I. description (parsed)	5	100	14	4.7, 14.9, 22.3
5	A. summary + I. description (parsed)	10	100	14	1.4, 5.6, 10.9
BASELINE					4.9,11.8,25.7

5.5 Additional Experiments

So now we will combine the best-performing article-level model with our baseline model. We use article-level model since it truly shows how model will perform on unseed data. To do so, we will also need to adjust our evaluation metric properly. So for this experiment, we will use only article summary as input and evaluate the percentage of articles where at least one true image was within top 1, 3 and 10 images respectively. Please note, that we are using a model trained on a different metric. Thus Word2VisualVec precision will be worse than it should be. The point here is to see whether we can compound two models.

As shown in Table 5.6, we indeed have a significant increase in performance when compounding models. An especially big improvement is in respect to R@1 metric. Since one model exploits text information and another mostly rely on image information, we believe that is the reason why "predictive powers" of models can sum up. For example, text-based model might improve precision when we have an article about some particular person because in this case, the most relevant information is in the text where we use the name of that person. While in case of, for example article about some landscape, more information might be encoded in the image and so image-based model will show higher confidence.

Table 5.6: Compound Model Experiments

Compound Model Experiments		
	Model	Precision
1	Word2VisualVec	13.2, 21.3, 32.2
2	Text Similarity	26.4, 40.8, 55.7
3	Compound	40.2, 55.2, 64.4

5.6 Model Demonstration

Here we will demonstrate some interesting model results on particular articles. Specifically, we used the best-performing article-level model. Results of each run are illustrated with ten top-ranked images for each input, all of them from public domain so can be freely illustrated. Moreover, each image is outlined with green colour when it was guessed correctly, and red - otherwise. Each figure also has a link on the original article to examine all expected images. Also, each article illustrated here is taken from the test subset, so model sees this data for the first time.

As we can see from Figure 5.1, the model performed really well on Jupiter article, because here we have 50% of correctly identified images. Furthermore, we can notice that 4-th and 5-th images are Jupiter as well, so performance is actually at least 70% here. The problem is that some images of Jupiter might be used in other articles but not in this one. So our evaluation metric does not recognise it as a correct image. We will need to address this problem of evaluation metric in the future when the model is trained on a significantly bigger image dataset.

On Figure 5.2, model correctly identified four images, which is a good example as well. A good sign is also that other mismatched images are also mostly sports cars, just as for Jupiter it were other planets. It shows that the model

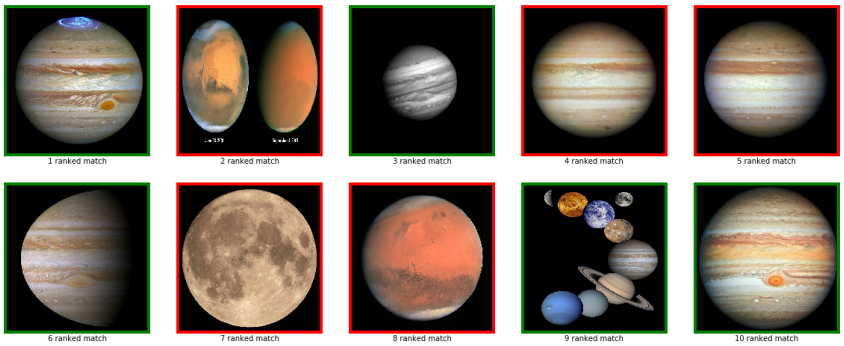

Figure 5.1: Article-level model output for "Jupiter" article. Green outline show correctly guessed images, red - incorrectly. https://en.wikipedia.org/wiki/Jupiter

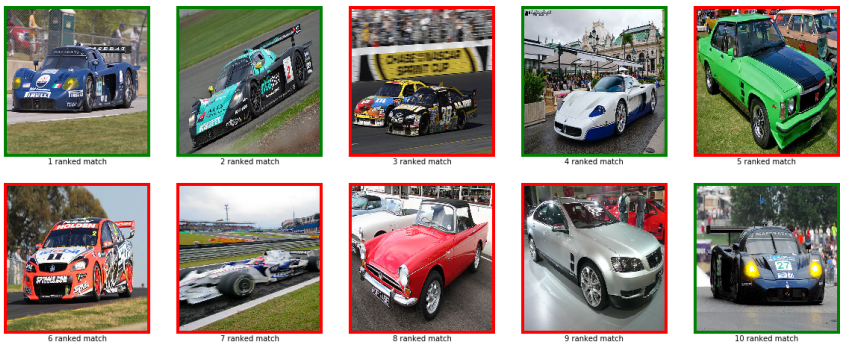

Figure 5.2: Article-level model output for "Maserati MC12" article. Green outline show correctly guessed images, red - incorrectly. https://en.wikipedia.org/wiki/Maserati_MC12

Figure 5.3: Article-level model output for "Emma Stone" article. Green outline show correctly guessed images, red - incorrectly. https://en.wikipedia.org/wiki/Emma_Stone

succeeded in identifying general concepts of the article fully but cannot correctly

guess the fine-grained query of a sports car of a specific brand. As we mentioned in Chapter 4, we believe it is the matter of data quantity when the model can grasp better those precise concepts.

As a failing example, we can look at Figure 5.3. Even though the model correctly identified the concept of human described in "Emma Stone" article, it did not have a way to identify the particular previously unseen person. We believe that the performance of the model on such specific cases can only be improved with the help of additional text similarity model.

Additional model output examples can be found in Appendix B

Chapter 6

Conclusions

6.1 Conclusions

So in this work, we created a model for recommending images based on Wikipedia articles. We showed that even basic deep neural network model, which exploits multimodal information, significantly outperforms our baseline of a simple text-based search upon image tags or description. Moreover, those results are for article-level model. When we manage to increase its performance to image-level, which we believe is the matter of dataset size, results will even better.

Additionally, text-based model and the model which strongly relies on visual information are partly complementary to each other. Thus, as we showed in experiments when combining both of them, we can achieve even higher quality than any of them shows separately.

We also created a multimodal dataset of more than 36K high-quality Wikipedia articles, which is publicly available and might be useful for further researches in this field.

All work which was made in the scope of this project is available from Kaggle. Because of that, anyone can instantly reproduce results without a need to download anything or to set up the environment.

6.2 Future Work

As we described, this is only a simple multimodal model to recommend images. The idea was to create a minimum viable product to showcase that our problem can be successfully solved with multimodal techniques. Thus there is plenty of required work and research to be done before we can get to the final real-world solution. The main directions for the project are:

- use more complex model, which will learn the feature representation for our data rather than specifying them in advance
- exploit additional metedata provided by wikipedia to increase prediction strength of the model. Such metadata might be categories associated with each article and image or additional image description specified in each referenced article.
- train model on bigger datasets of "good articles" in order to increase quality.
- test the model in real-world scenario of entire Commons image dataset

- adjust evaluation metric to recognise photos of the same entity as correct output, not just one mentioned in the article. That is, if we have three pictures of Tower Bridge from different sides of the river, we should acknowledge any of them as a correct match, not just the single image used in the article.

Also, the planned work for near future is to make the model accessible in real-time via public API.

Appendix A

Data

A.1 Structure

A.1.1 High-Level Structure

```
.
+-- page1
|   +-- text.json
|   +-- img
|       +-- meta.json
+-- page2
|   +-- text.json
|   +-- img
|       +-- meta.json
:
+-- pageN
|   +-- text.json
|   +-- img
|       +-- meta.json
```

where:

- pageN - is the title of N-th Wikipedia page and contains all information about the page
- text.json - text of the page saved as JSON. Please refer to the details of JSON schema below.
- meta.json- a collection of all images of the page. Please refer to the detals of JSON schema below.
- imageN - is the N-th image of an article, saved in ‘jpg‘ format where width of each image is set to 600px. Name of the image is md5 hashcode of original image title.

A.1.2 text.json Schema

```
{
  "title": "Naval Battle of Guadalcanal",
  "id": 405411,
  "url": "https://en.wikipedia.org/wiki/Naval_Battle_of_Guadalcanal",
  "text": "The Naval Battle of Guadalcanal, sometimes referred to.. ",
}
```

where:

- title - page title
- id - unique page id
- url - url of a page on Wikipedia
- text - text content of the article escaped from Wikipedia formatting

A.1.3 meta.json Schema

```
{
  "img_meta": [
    {
      "filename": "d681a3776d93663fc2788e7e469b27d7.jpg",
      "title": "Metallica Damaged Justice Tour.jpg",
      "description": "Metallica en concert",
      "url": "https://en.wikipedia.org/wiki/File%3
          AMetallica_Damaged_Justice_Tour.jpg",
      "features": [123.23, 10.21, ..., 24.17],
    },
  ]
}
```

where:

- filename - unique image id, md5 hashcode of original image title
- title - image title retrieved from Commons, if applicable
- url - url of an image on Wikipedia
- features - output of 5-th convolutional layer of ResNet152 trained on ImageNet dataset. Features taken from original images downloaded in ‘jpeg‘ format with fixed width of 600px. Practically, it is a list of floats with len = 2048.

Please note that some images are not embedded on Wikipedia page from Commons, thus we can only download them in original type & size. If you want to use those as well, those images should be properly processed later. Each such image can be identified by suffix ‘.ORIGINAL‘ in a ‘filename‘ and absence of key ‘features‘. Raw images are available in complete version of dataset[1]

A.2 Dataset Links

- featered articles + raw images[1]
- featured articles: 500 pages subset[2]
- featured articles[3]
- good articles[4]

[1] https://drive.google.com/file/d/1l0Oyv2Y6LmPGN3lP9MB6i8WWCinqkYPk/view?usp=sharing
[2] https://www.kaggle.com/jacksoncrow/wiki-articles-multimodal
[3] https://www.kaggle.com/jacksoncrow/extended-wikipedia-multimodal-dataset
[4] https://www.kaggle.com/jacksoncrow/wikipedia-multimodal-dataset-of-good-articles

Appendix B

Model Results Demo

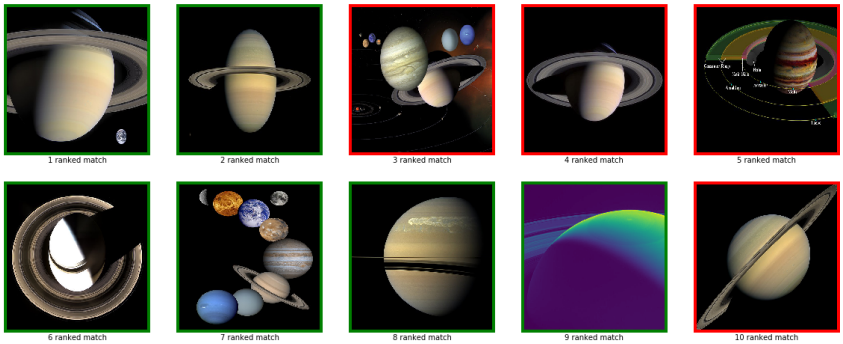

Figure B.1: Article-level model output for "Saturn" article. Green outline show correctly guessed images, red - incorrectly. https://en.wikipedia.org/wiki/Saturn

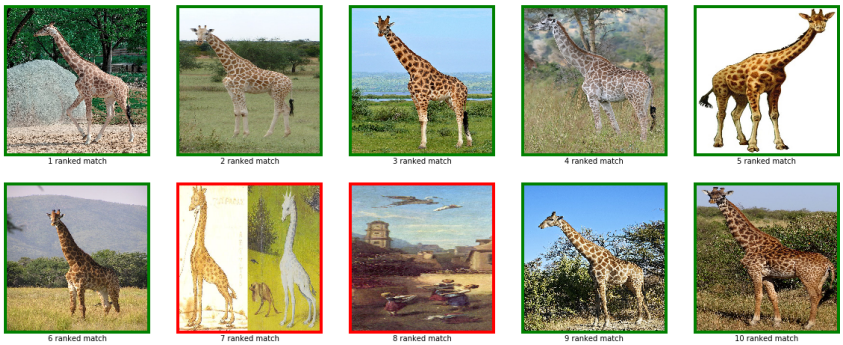

Figure B.2: Article-level model output for "Giraffe" article. Green outline show correctly guessed images, red - incorrectly. https://en.wikipedia.org/wiki/Giraffe

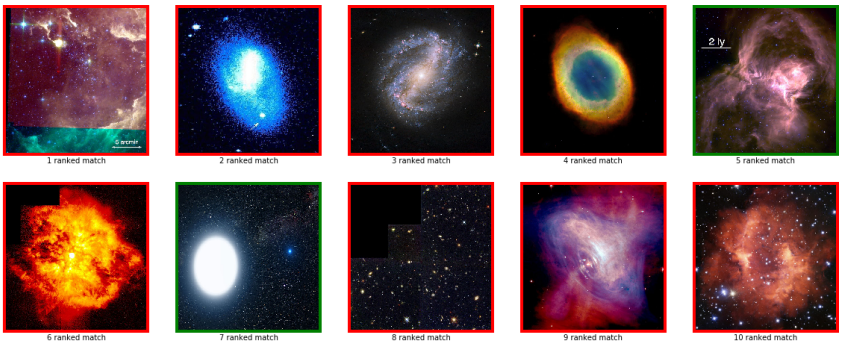

Figure B.3: Article-level model output for "Star" article. Green outline show correctly guessed images, red - incorrectly. https://en.wikipedia.org/wiki/Star

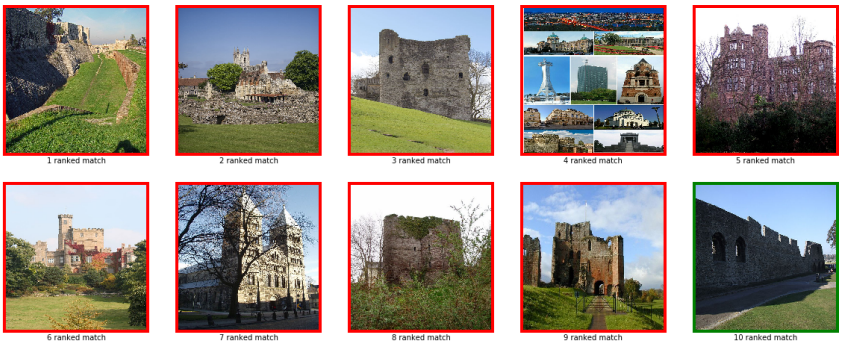

Figure B.4: Article-level model output for "Rochester Castle" article. Green outline show correctly guessed images, red - incorrectly. https://en.wikipedia.org/wiki/Rochester_Castle

Figure B.5: Article-level model output for "Kennedy Half Dollar" article. Green outline show correctly guessed images, red - incorrectly. https://en.wikipedia.org/wiki/Kennedy_half_dollar

Bibliography

[1] T Baltrušaitis, C Ahuja, and L Morency. “Multimodal Machine Learning: A Survey and Taxonomy”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 41.2 (Feb. 2019), pp. 423–443.

[2] David M Blei, Andrew Y Ng, and Michael I Jordan. “Latent Dirichlet Allocation”. In: *J. Mach. Learn. Res.* 3.Jan (2003), pp. 993–1022.

[3] Kyunghyun Cho et al. “On the Properties of Neural Machine Translation: Encoder-Decoder Approaches”. In: (Sept. 2014). arXiv: 1409.1259 [cs.CL].

[4] Alexis Conneau et al. “Supervised Learning of Universal Sentence Representations from Natural Language Inference Data”. In: (May 2017). arXiv: 1705.02364 [cs.CL].

[5] Jia Deng et al. “Imagenet: A large-scale hierarchical image database”. In: *2009 IEEE conference on computer vision and pattern recognition*. researchgate.net, 2009, pp. 248–255.

[6] S K D’mello and J Kory. “A review and meta-analysis of multimodal affect detection systems”. In: *ACM Computing Surveys (CSUR)* (2015).

[7] J Dong, X Li, and C G M Snoek. “Predicting Visual Features From Text for Image and Video Caption Retrieval”. In: *IEEE Trans. Multimedia* 20.12 (Dec. 2018), pp. 3377–3388.

[8] Jeffrey L Elman. “Finding Structure in Time”. In: *Cogn. Sci.* 14.2 (Mar. 1990), pp. 179–211.

[9] Andrea Frome et al. “DeViSE: A Deep Visual-Semantic Embedding Model”. In: *Advances in Neural Information Processing Systems 26*. Ed. by C J C Burges et al. Curran Associates, Inc., 2013, pp. 2121–2129.

[10] Akira Fukui et al. “Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding”. In: (June 2016). arXiv: 1606.01847 [cs.CV].

[11] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. “Deep sparse rectifier neural networks”. In: *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. jmlr.org, 2011, pp. 315–323.

[12] W Guo, J Wang, and S Wang. “Deep Multimodal Representation Learning: A Survey”. In: *IEEE Access* 7 (2019), pp. 63373–63394.

[13] Amirhossein Habibian, Thomas Mensink, and Cees G M Snoek. “Video2vec Embeddings Recognize Events When Examples Are Scarce”. en. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 39.10 (Oct. 2017), pp. 2089–2103.

[14] Zellig S Harris. “Distributional Structure”. In: *Word World* 10.2-3 (Aug. 1954), pp. 146–162.

[15] Kaiming He et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. openaccess.thecvf.com, 2016, pp. 770–778.

[16] Jack Hessel, David Mimno, and Lillian Lee. “Quantifying the visual concreteness of words and topics in multimodal datasets”. In: (Apr. 2018). arXiv: 1804.06786 [cs.CL].

[17] S Hochreiter and J Schmidhuber. “Long short-term memory”. en. In: *Neural Comput.* 9.8 (Nov. 1997), pp. 1735–1780.

[18] Xin Huang, Yuxin Peng, and Mingkuan Yuan. “Cross-modal Common Representation Learning by Hybrid Transfer Network”. In: (June 2017). arXiv: 1706.00153 [cs.MM].

[19] Yu-Gang Jiang et al. “Exploiting Feature and Class Relationships in Video Categorization with Regularized Deep Neural Networks”. en. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 40.2 (Feb. 2018), pp. 352–364.

[20] Yoon Kim et al. “Character-aware neural language models”. In: *Thirtieth AAAI Conference on Artificial Intelligence*. aaai.org, 2016.

[21] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems 25*. Ed. by F Pereira et al. Curran Associates, Inc., 2012, pp. 1097–1105.

[22] David G Lowe. “Distinctive Image Features from Scale-Invariant Keypoints”. In: *Int. J. Comput. Vis.* 60.2 (Nov. 2004), pp. 91–110.

[23] Lin Ma et al. “Multimodal convolutional neural networks for matching image and sentence”. In: *Proceedings of the IEEE international conference on computer vision*. openaccess.thecvf.com, 2015, pp. 2623–2631.

[24] H McGurk and J MacDonald. “Hearing lips and seeing voices”. en. In: *Nature* 264.5588 (1976), pp. 746–748.

[25] Tomas Mikolov et al. “Efficient Estimation of Word Representations in Vector Space”. In: (Jan. 2013). arXiv: 1301.3781 [cs.CL].

[26] Noam Mor et al. “A Universal Music Translation Network”. In: (May 2018). arXiv: 1805.07848 [cs.SD].

[27] Yingwei Pan et al. “Jointly modeling embedding and translation to bridge video and language”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. openaccess.thecvf.com, 2016, pp. 4594–4602.

[28] Jeffrey Pennington, Richard Socher, and Christopher Manning. “Glove: Global vectors for word representation”. In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. aclweb.org, 2014, pp. 1532–1543.

[29] Bryan A Plummer et al. “Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models”. In: *Proceedings of the IEEE international conference on computer vision*. openaccess.thecvf.com, 2015, pp. 2641–2649.

[30] Nikhil Rasiwasia et al. “A new approach to cross-modal multimedia retrieval”. In: *Proceedings of the 18th ACM international conference on Multimedia*. 2010, pp. 251–260.

[31] Scott Reed et al. “Generative Adversarial Text to Image Synthesis”. en. In: *International Conference on Machine Learning*. jmlr.org, June 2016, pp. 1060–1069.

[32] Karen Simonyan and Andrew Zisserman. “Very Deep Convolutional Networks for Large-Scale Image Recognition”. In: (Sept. 2014). arXiv: 1409.1556 [cs.CV].

[33] Richard Socher et al. “Grounded Compositional Semantics for Finding and Describing Images with Sentences”. In: *Transactions of the Association for Computational Linguistics* 2 (Dec. 2014), pp. 207–218.

[34] Subhashini Venugopalan et al. “Translating Videos to Natural Language Using Deep Recurrent Neural Networks”. In: (Dec. 2014). arXiv: 1412.4729 [cs.CV].

[35] Oriol Vinyals et al. “Show and tell: A neural image caption generator”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. cv-foundation.org, 2015, pp. 3156–3164.

[36] Douglas Rudy Vogel et al. *Persuasion and the role of visual presentation support: The UM/3M study*. Management Information Systems Research Center, School of Management . . ., 1986.

[37] Liwei Wang, Yin Li, and Svetlana Lazebnik. “Learning deep structure-preserving image-text embeddings”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. cv-foundation.org, 2016, pp. 5005–5013.

[38] Wikimedia Foundation, Inc. *Wikimedia Commons*. https://commons.wikimedia.org/wiki/Main_Page. Accessed: 2019-12-29. Sept. 2004.

[39] Ikuya Yamada et al. “Wikipedia2Vec: An Optimized Tool for Learning Embeddings of Words and Entities from Wikipedia”. In: *CoRR* (2018).