Project Report: Data Science 2021 - Group 01

Ahmed Amine Cherif
Technische Universität Berlin, Germany

Mohamed Amine Dhiab
Technische Universität Berlin, Germany

Joanna Lenkiewicz
Technische Universität Berlin, Germany

Martin Zehetner
Technische Universität Berlin, Germany

ABSTRACT

With the increasing use of social networks by people and the speed with which information is circulated on these networks, traditional newspapers have to be more and more effective and meaningful. In order to attract new users and to convert interest into clicks in an increasingly competitive market, the choice of the right combination of article image and text, title and content, is particularly important. Academically, the investigation of this task can be reduced to a re-matching problem between article images and text, which is investigated in this paper using real world data from an online newspaper. In this work, we use a "Self-Attention Embeddings for Image-Text Matching" (SAEM) model to embed images and texts in the same semantic space and thus match similar images and texts by using simple metrics such as cosine similarity. Furthermore, in the course of this work, methods from the transfer learning and data augmentation domain and their performance impact were investigated. A general suitability of the approach for the solution of the problem could be recognized, among other things by outperforming various baselines, and in particular the transfer learning approaches could be ascribed a noticeable added value. The results of this work encourage further research on multimodal embedding approaches and transfer learning strategies in the news re-matching domain.

KEYWORDS

MediaEval, image-text matching, self-attention, bottom-up attention, context sensitive embedding

1 INTRODUCTION

Online news media is, nowadays, undoubtedly one of the most important source of information and developed rapidly in recent years. Online news articles usually try to convey their content distinctively and direct by using expressive titles, texts and images. Images try to visualise the content and give the reader a strong first impression and understanding of what a particular article is about. Together with headlines, these are the two elements readers first pay attention to. Understanding the relationship of textual content and images of news article is crucial in multimedia studies to improve the credibility of the media and to achieve more interesting content for the user. In this case, when the pictures used for a given article have little to do with its content, it may appear as fake news or as low-quality content to the reader. Investigation of these relationship comes down to image-text matching problems, which are unquestionably a challenging, but important, task. In this work, we present a method which attempts to solve this problem, in form of a real world re-matching task between article texts and images of a real online news paper, using multimodal embeddings, meaning embeddings that map images and texts into the same vector space, to measure semantic similarity between text and image. For this purpose we employee a "Self-Attention Embeddings for Image-Text Matching" (SAEM) model [44] to generate directly comparable text and image embedding. Furthermore we use transfer learning techniques and text data augmentation methods to investigate, if either of these domains show promise in attempting to solve a re-matching task. To compromise for the relatively small size of the training data available compared to the complexity of the problem tackled, we will use the mentioned text augmentation techniques to try to generate additional training examples, for example by creating paraphrases of available article titles.

The work is now structured as follows. First, a short description of the problem is given and general aspects are discussed. After that, interesting works regarding the topic are highlighted and then the methods used by us are introduced theoretically and in respect to the task. After that, the data used and the experimental setup are discussed before the results are analyzed and a conclusion is drawn.

2 PROBLEM DESCRIPTION

2.1 General

The problem considered in this work is to re-establish the links between news articles and images that were originally posted with the articles in question [16]. The images are a mix of original pictures, that for example come from the event itself described in the article, and stock images that were picked out from existing databases. The main challenge is to achieve the highest possible proportion of correctly matched pairs, based on the provided information.

2.2 Baselines

We designed four baselines to compare our solution with. First two baselines are based on matching articles and images using cosine similarity of feature vectors. In the first baseline, text feature vectors are calculated for the article texts using term frequency–inverse document frequency (TFIDF) [35], which is a numerical statistic that indicates how important a word is to a document in a corpus. To calculate image feature vectors, pretrained Convolutional Neural Networks (CNNs) are used and the feature vectors are represented by the outputs of non-output dense layers. In our baseline VGG19 [37] and the second dense layer before the output layer is used. The baseline results are then achieved in the following way. Cosine distance is calculated for each feature vector from evaluation data with every feature vector from training data. Then for each article

Data Science '21, Summer Term 2021, Berlin, Germany

in the training set, one article from the evaluation set is assigned, which has the smallest distance and thus is considered to be most similar, together with its corresponding image. In the following step, for each of those images, cosine distance between their feature vectors and each feature vector of images in the evaluation set is calculated, to find and assign the 100 most similar images, which are then matched to the matched images for a certain article. The second baseline is based on the same principle, however instead of using TFIDF to calculate text's feature vectors, features are extracted using Doc2Vec [33], which is trained on English Wikipedia Dump. In this case, titles of the articles were translated from German to English using the DeepL Online Translator and then used. The third and forth baselines are based on the model for Multimodal Embeddings (SAEM) pre-trained with image captioning data sets without training on article data.

2.3 Evaluation metrics

The metrics used in this work will be shortly introduced here. When trying to find the match of a given article text, we predict for each given article a possible ranking of the most likely article images. In our case, we predicted a maximum of 100 images for each article. To evaluate the prediction quality, we use two different evaluation metrics. The first metric used is Accuracy@N (ACC@N), which is calculated using the following formula

$$Accuracy@N = \frac{1}{K}\sum_{i=1}^{K} \mathbb{1}(\text{ image "i" is in the top N}) \tag{1}$$

The internal accuracy term is equal to 1, if the true image lies within the number of N predictions, otherwise the term is equal to 0. It is calculated for each article and added together and then averaged over their number of articles. The motivation for using this metric is as follows. In general for retrieval problems, the maximum true elements among the N retrieved elements is N, which is why some Accuracy@N definitions also sometimes include a division by N, which is for normalization purposes. In our case, we do not include this division, since the maximum true images among the N retrieved images is 1, therefore a normalization over the number of retrieved images is not needed. The second metric we use is MRR@N, that is Mean Reciprocal Rank, which is a metric for evaluating ranked list. To each prediction a rank between 1 and N is given which represents the place ρ in the predicted ranked list of matchings, if the image is present in the ranked list. Otherwise, If the prediction is not within N predictions, the internal term is set to 0. The formula to calculate the metric is then:

$$MRR@N = \frac{1}{K}\sum_{i=1}^{K} \frac{1}{\rho(\text{ image "}i\text{"})} \tag{2}$$

It can be seen that matchings with a smaller rank, that means seen as more likely to be the match in the ranked matching list, contribute a larger value in the sum. The MRR@N is then calculated by averaging these reciprocal ranks of the true images.

3 RELATED WORK

3.1 News Media Tasks

Data science in the field of media and entertainment has become a prerequisite to drive decision-making and each task in this field tries to solve a certain problem based on the multimedia data given like the "Predicting Media Interestingness" task [6] which is part of the MediaEval 2017 Benchmarking Initiative for Multimedia Evaluation, the objective is to automatically select images and/or video segments that are considered to be the most interesting for a common viewer. Interestingness of media is to be judged based on visual appearance, audio information and text accompanying the data, including movie metadata. In addition to "The Fake News Detection" task [3] with the goal of detecting if an article is fake news or legitimate news.

3.2 Image-Text Matching

One of the most important tasks in the field of News and Media is Image-Articles/Texts matching and a lot of the existing methods proposed for such task, where the key issue is measuring the visual-semantic similarity between a text and an image, are related to our work. In fact, Learning a common space where text and image feature vectors are comparable is a typical solution for this task.

Features Embedding : A large amount of research papers has been done on learning multimodal representations of images and text. Popular approaches include learning joint image-text embeddings as well as embedding images and sentences into a common space. From the examples of such works we mention Kiros et al. [19] who use LSTM [13] for the learning of text representation . Frome et al. [10] a feature embedding framework that uses Skip-Gram [12] and CNN to extract feature representations for cross-modal. Then a ranking loss is adopted to encourage the distance between the mismatched image-text pair is larger than that between the matched pair. Also one of the most popular baselines for image-text embedding is Canonical Correlation Analysis (CCA), which finds linear projections that maximize the correlation between projected vectors from the two views, recent works using it include Ba et al. [4]. To obtain a nonlinear embedding, other works have opted for kernel CCA like [28] which tries to find a projection that maximizes the correlation of kernel Hilbert spaces with corresponding kernels. Despite being a classic textbook method, CCA has turned out to be a surprisingly powerful baseline.

Similarity Prediction : Learning a measure of similarity between pairs of objects is a fundamental problem in machine learning. It stands in the core of classifications methods like kernel machines, and is particularly useful for applications like searching for images that are similar to a given image or finding videos that are relevant to a given video. In these tasks, users look for objects that are not only visually similar but also semantically related to a given object. For example Karpathy et al. [15] propose to detect and encode image regions at object level with R-CNN, and then infer the image-text similarity by aggregating the similarity scores of all possible region-word pairs. While Lee et al. [21] propose stacked cross attention (SCAN) to align image regions and text words. It first calculates cosine similarity between all image regions and words of sentence to get attended sentence vectors which are

weighted combination of word representations, then it calculates cosine similarity between all attended sentence vectors and image region features and uses LogSumExp pooling or average pooling to obtain the final image-sentence similarity. However, this strategy may lack efficiency involving preparing all the image-text pairs to predict the matching score at the test stage.[44]

3.3 Attention Mechanism

Our work is also inspired by bottom-up attention [38], [1] mechanism and recent image-text matching methods based on it [21]. Bottom-up attention refers to salient region detection at stuff/object level can be analogized to the spontaneous bottom-up attention that is consistent with human vision system [30], [1]. Bottom-up attention models in the visual context can be applied to various tasks like image classification [40], object detection [9], image generation [47], image captioning [45], etc. The textual attention approach had a lot of success in tasks like Sentiment Analysis [22], Text Summarizing, [34] and Machine Translation [25]. Recently, Vaswani et al. (2017) [39] proposed the Transformer architecture for machine translation. It relies only on attention mechanisms, instead of making use of either recurrent or convolutional neural networks. This architecture contains layers called self-attention (or intra-attention) which allow each word in the sequence to pay attention to other words in the sequence, independently of their positions. The usage of these transformers in our work was established through the new language representation model developed by Google Research called BERT [8], which stands for Bidirectional Encoder Representations from Transformers. Unlike recent language representation models, BERT is designed to pre-train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers. Unlike Word2Vec [33] and GloVe [31] which are context insensitive, the word embeddings produced by Transformer are context sensitive representations. Context sensitivity means giving different representations according to the sentences. For example, "tie" in the context of clothes and in the context of sports would not have the same representation.

4 APPROACH

The following section describes the core ideas behind and the structure of the components contained in our approach. In broad terms the approach uses text data augmentation steps, a model to generate multimodal text and image embeddings and a cosine similarity function to solve re-matching tasks involving news texts, i.e., titles or parts of article texts, and associated images. To this end, the three core ideas of our approach are first briefly summarized and then explained and motivated in the following subsections.

The steps of our approach can briefly be summed up as the use of the "Self-Attention Embeddings for Image-Text Matching" (SAEM) model [44] to generate directly comparable text and image embeddings in a common vector space, transfer learning to exploit general learning successes on exhaustive data sets and a subsequent specialization to our specific task domain, i.e. online news texts and images, and the augmentation of considered article titles and texts to generate additional training examples.

4.1 Self-Attention Embeddings for Image-Text Matching (SAEM)

The main steps of our approach involves the use of the SAEM model introduced by Wu et al. [44]. In the following, the architecture of the SAEM model, see Fig. 1, will be roughly outlined. Furthermore, the corresponding internal model processing steps of the input data are highlighted. Afterwards the usage of the SAEM model in the context of our approach will be shortly described.

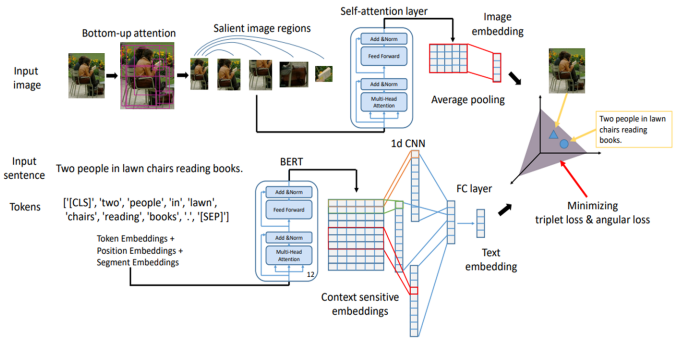

Figure 1: An illustration of the two-branch structure, an image (top) and a text embedding branch (bottom), of the SAEM model for generating multimodal embeddings in a common vector space [44]

The SAEM model can be understood as a neural network divided into two branches, where the individual branches cover the processing of inputs from two different modalities, i.e. image and text. In the first of these "branches", salient regions in input images are identified and converted to feature vectors using a bottom-up attention [2] mechanism consisting of a Faster R-CNN with ResNet-101 pre-trained on Visual Genomes [20], these region-related feature vectors and their intra-modal relations are encoded using specially defined self-attention layers, and the final normalized image embeddings are generated using an average pooling step. In the second "branch", text inputs are processed by first tokenizing the texts using the WordPiece tokenizer [7], afterwards token representations are converted to continuous context-sensitive word embeddings using the encoder of a transformer initialized with the pre-trained BERT (Bidirectional Encoder Representations from Transformers) [7] architecture. The continuous representations are then feed as uni-, bi- and tri-grams into three distinct 1D Convolutional Neural Network Layers [17] and a following Dense Layers and a final normalization step subsequently generate the global text embeddings.

The described structure and self-attention mechanisms make it possible to depict the relations between text and image fragments intra-modally and thus to encode and aggregate the information obtained in the text and image embeddings. However, to also enable learning of representing inter-modal semantic relations, i.e., to guarantee the mapping of semantically similar images to texts, Wu et al. [44] introduce a weighted combination of a triplet loss [42] and angular loss [41]. Here, the "scoring" function used in the losses to compute the immediate similarity explicitly corresponds to cosine similarity, so that in later comparisons of generated text and image embeddings similarities can be determined directly using

cosine similarity.

We use the SAEM model in our approach as the crucial component to transform lists of potential article text elements and article images into multimodal semantically meaningful image and text embeddings. The individual images and texts can subsequently be compared by applying the cosine similarity and matches between the most similar images and article text elements can be computed. Important to point out here is that, unlike in the introduction of the SAEM model described in [44], in which the focus was only on matching semantically identical images and texts, i.e., the matched texts should exclusively describe exactly what can be seen on the images and vice versa, the focus in our approach lays rather on the somewhat more abstract task of mapping the similarity of image and text elements in the semantics of the online news article context, e.g., which image would be selected to match a given article title. In our approach, we therefore train the model not by using mutually descriptive text-image pairs, like the Flickr30k [46] or MS-Coco [23] data sets used in [44], but by using specifically extracted article-image and article-text element combinations. For the direct use of the MediaEval data in the SAEM model, it is also necessary to translate the German article text elements into English, because among other factors, the BERT model used in the text processing branch of the SAEM was pre-trained on English data.

4.2 Transfer Learning

Transfer learning in the fields of machine learning and neural networks describes in broad terms the transfer or exploitation of knowledge learned and stored in one scenario to improve training flow, results, or generalization in another domain [11]. In this subsection, we now briefly describe a popular transfer learning strategy and highlight potential benefits. This is followed by an explanation of the transfer learning strategy explicitly performed in our approach.

Transfer learning encompasses a variety of different approaches and strategies for using knowledge acquired previously in new task domains. Often these strategies are used when in real world scenarios there is not enough immediate training data available for a task, but related scenarios offer a larger amount of labeled available data [26]. One of the most used and most researched types of strategies in the field of neural networks are various pre-trained model approaches. These are characterized by the initial selection of an already trained model from another task domain, which is similar or related to the currently considered task domain [48]. Subsequently, this pre-trained model is used in whole or in part as a starting point in the current task domain. Often this is done in the form of fine-tuning, i.e. by using the pre-trained model as initial model state in training with the data points of the new task domain [27]. In this context, various experiments have shown that the chance of successful transfer learning, i.e. a succesful exploitation of previously learned knowledge, is greatly increased by initial training on a large-scale dataset and a general nature of considered features [29].

The use of transfer learning approaches in training a neural network offers a number of tangible benefits in this regard. These can be divided into three areas and include a possible better initial model quality, since previously achieved learning results can be exploited instead of random initialization of the models, a possible faster training process, in which a potential optimum can be found more quickly, and a possible higher optimum compared to a training scenario without transfer learning [24].

In our approach, we leverage the fact that the SAEM model to be applied to the MediaEval dataset has already been tested and evaluated on the two large general datasets, Flick30K and MS-COCO, and produced high-quality results with text-to-image matching recall@10 metrics of 88.1 on the Flick30K dataset and 94.9 on the MS-COCO dataset, respectively [44]. In this regard, two aspects advocate the use of transfer learning in this use case. First, as mentioned in the previous subsection, the task domains between the original introductory purpose of SAEM in [44], matching semantically identical texts and images, and the current task of rematching news article text elements with article images, although not identical, are highly related due to their nature as semantic matching tasks, and second, the enormous size of the Flick30K dataset, with 31,783 images each with five human-annotated captions, and the MS-COCO dataset, with 123,287 images each with five text captions, compared to the available MediaEval dataset with 13,478 images, each with only an article title and text, makes the exploitation of potentially learnable generalizable knowledge from the two larger datasets highly desirable.

Therefore, in our approach, we use a direct pre-trained model strategy, where we first reproduce the SAEM models described in [44] trained on either the Flick30K or the MS-COCO dataset, and then fine-tune the SAEM models by using the best versions of the pre-trained models as initial model states, and then train the models using the article image and the selected translated article text elements.

4.3 Data Augmentation & Finalization

This subsection now focuses on augmenting the data, specifically the article text elements.

In general, it can be stated that the effectiveness and performance of neural networks can be influenced to a large extent by the size and quality of the training data sets [32], and that an increase in, for example, the amount of training data for models of sufficient complexity usually leads to better results. At the same time, too little training data can lead to overfitting, i.e. optimization of performance on the training data with simultaneous loss of the ability to generalize [36].

However in many real world tasks, relatively few data points are available for the training process, such as in our MediaEval task. A potential solution on the data level is data augmentation, i.e. the generation of new training data points from existing ones. Data augmentation is frequently used in various computer vision and speech processing tasks, but is also applied in NLP tasks, where general rules of text transformation and general data augmentation methods are difficult to define [43].

To address the problem of the relatively small size of the available MediaEval data set, we defined two different text data augmentation processes to increase the number of training data points. These apply respectively to the translated article title and article text.

The first data augmentation process of our approach now consists of generating a new paraphrased title for each translated article title, where the original image is mapped to the original and newly generated title. By doing so, we naturally doubled the amount of training data. The second data augmentation approach used consists of splitting the article texts into the complete contained sentences of the article text. Again, the corresponding article image is assigned to all sentences found. The number of resulting training data obviously varies in this approach based on the number of contained sentences in the article texts.

Given these two techniques for data augmentation, our approach now distinguishes between three different "settings" of the dataset when training the SAEM model. Where the data points in each setting consist of one image and one text element. The first setting S_T simply represents the translated titles together with the corresponding image. In the second setting S_P there are two data pairs for each image, one with the original article title and one with the paraphrased title, and in the third setting S_S there are at least one, but possibly several data pairs for each image, where the text elements represent the complete sentences of the article text.

5 EXPERIMENT

This section discusses the details of the experiment conducted in the course of the project and further elaborates on the data used, details of the implementations, and the experiment protocol used. The goal of the experiment was to evaluate the effects of the transfer learning strategy and data augmentation methods described in Sec. 4, that is, specifically, to investigate how the performance, measured in terms of Accuracy@N and MRR@N, of the SAEM model to be trained relates to either no previous training or previous training on the image captioning datasets and the type of processed text inputs.

5.1 Experiment Data & Metrics

The datasets relevant for the experiment are the image captioning datasets, i.e., images with associated descriptive sentences, Flickr30K and MS-COCO used for the pre-training of the SAEM model, as well as the MediaEval dataset consisting of articles of the "Kölner Stadtanzeiger", which contains online news article image and text element, i.e. partial article texts and article title, pairs.

5.1.1 Flick30K and MS-COCO. The two datasets Flickr30K and MS-COCO are among the most used datasets in the field of image captioning, as they have both a large number of data points in the form of images and high quality descriptive text captionings, in the case of Flickr30K the annotations were even generated by humans. Both datasets have the same structure, in which five image descriptive text annotations are assigned to each of the 31,783 (Flickr30K) and 123,287 images (MS-COCO). We follow the same public split [21] as Wu et. al [44] in their training of SAEM, in which the Flickr30K dataset was split into 29,783 training, 1000 validation and 1000 test images, and the MS-COCO dataset was split into 113,287 training, 5000 validation and 5000 test images.

5.1.2 MediaEval. The MediaEval dataset provided for the rematching task consists of several batches of article elements and metadata, whereby only the existing combinations of individual images, titles and texts of the articles are relevant for the task and our choosen approach. The article images have a resolution of approximately 300x150px and the article texts are cut off after 255 characters. Both article titles and texts are in German and the data used in the experiment comprises three batches containing articles from a German online news platform. The batches roughly represent the articles of the months January, February, and March 2019, respectively. In total, the datasets we considered contain 13,478 articles.

Table 1: Statistics: MediaEval Experiment Data Set - Rough time frame of publication of the articles, number of articles and train, validation or test split association.

	Batch 1	Batch 2	Batch 3.1	Batch 3.2
Time Span	January	February	March	March
No. Articles	4688	4676	2057	2057
Split	Training	Training	Validation	Evaluation

The exact number of items in each batch and their purpose in the experiment are shown in Tab. 1. We use the first two batches as training data and split the last batch chronologically to create validation and test datasets that have a clear temporal separation from the training dataset. This was done to mimic the actual use cases of matching images with text articles, where past news data is used to find relevant images for specific news articles. The result is a training set with 9364 artiles representing approximatly 70 % of the articles and validation and test sets with 2057 articles each representing roughly 15 % of the data.

After introducing the used datasets in more detail, we will now specify the metrics used for evaluation in the experiment. In the case of pre-training, i.e., training and evaluation on the image-captioning datasets, we again follow Wu et. al. [44] by determining the proportion of validation or test results in which at least one relevant image was ranked among the top 1, 5, and 10. The best model is then determined by taking the arithmetic mean of the three metrics.

For the evaluation on the MediaEval data, however, we use the metrics ACC@N and MRR@N introduced in Sec. 2. For our experiment we focus on evaluating the more general performance of the generated models by measuring the performance using ACC@N and MRR@N with $N = 1, ..., 100$. The best models are again determined by averaging over the respective metrics.

5.2 Implementation Details

In this subsection, the details and specifications of the various processing steps performed, model structures implemented, and hardware components used are given to ensure later reproducibility.

5.2.1 Text Processing & Augmentation. Four external services and libaries were used during the experiment to process and augment the article titles and texts before feeding them into the SAEM model. The desktop app of the commercial DeepL software was first used to translate the titles. In contrast, due to limitations in

the length of the translatable texts, the article texts were translated using the Google Translate API.

For the text augmentation steps, the commercial Quillbot Pro web platform was afterwards used to generate paraphrases for the article titles, with the generated phrases being created in standard mode with the highest possible abstraction level. The text augmentation based on splitting the article texts was then performed using the NLTK [5] sentence tokenizer.

5.2.2 Image Pre-Processing. Image preprocessing is mentioned separately here, as generating the image features of the salient image regions using the bottom-up-attention approach is not included in the published version of the SAEM implementation [44] and has to be implemented separately using the SCAN [21] and Bottom-up-Attention [2] Github projects. Processed features are available for MS-COCO and Flickr30K [21], but for new datasets, such as the MediaEval images, they have to be generated separately. In this sense, we set the number of salient image regions the Bottom-up-Attention ,mechanism is suppose to find to a fixed 36 to mimic the shape of the given MS-COCO and Flickr30K feature vectors.

Required for this process on the software side is the Caffe library [14], OpenCV and Nvidia's NCCL library, whereas on the hardware side a GPU with at least 12 GB of internal RAM is needed. We use a Docker container for our experiment to provide the required software setup and used a Tesla P100 with 16GB of internal memory as the GPU.

5.2.3 Model Implementation. For the experiment, most of the hyperparameters and details in the model structures of the SAEM from Wu et. al. [44] were adopted. Thus, among other things, the PyTorch framework was obviously used and the Adam [18] optimizer with an initial learning rate of 0.0001 and Bachtsize of 64 was employed in the learning process. During pre-training, the decay rate of 0.1 was adopted after every 10 epochs, but when training on the MediaEval data, it was modified so that the learning rate was only updated after every 15 epochs. Furthermore we set the gradient clipping threshold to 2.0 while training. In terms of the structure of the model, the dimension of the internal word embedding was set to 300 and the dimension of the extracted features of the image region was set to 2048. The dimension of the final multimodal embedding was furthermore fixed to 256. For pre-training on the image captioning datasets and training on the MediaEval data, a GeForce GTX 1070 GPU was used.

5.3 Experiment Protocol

The protocol used to conduct the experiment is explained below. The first phase of the experiment refers to the training on the image captioning data and the selection of the best "pre-trained" model for the further course of the experiment. In a second phase, the training and evaluation on the MediaEval data is described.

5.3.1 Training & Evaluation of pre-trained models. The process described below is performed once separately for Flickr30K and MS-COCO. Initial input of the first experiment phase are lists of image-annotation pairs $(I_{Image},\ I_{Text})$, where the data is split according to the split described in Sec. 5.1.1 into training, validation, and test data. For each image I_{Image}, there are five pairs, with the corresponding five annotations forming the respective I_{Text} elements. The models are then trained for 30 epochs and after each epoch the ratio of recommendations in which at least one relevant image was ranked among the top 1, 5, and 10 is determined on the validation data. At the end of the 30 epochs, the model with the highest mean between these 3 metrics is then selected as the best "pre-trained" model.

5.3.2 Training & Evaluation of MediaEval Models. The core of the second phase of the experiment is the comparison of the performance of the models with different configuration combinations. The "Initial Model State" M_{Init} and the "Textual Input Type" T_{Type} can be seen as possible variables of the configurations. "Initial Model State" M_{Init} can take values representing initialization of the SAEM model randomly, based on the best pre-trained Flickr30K model or based on the best pre-trained MS-COCO model. "Textual Input Type" T_{Type}, on the other hand, can take variables representing the use of only translated article titles without data augmentation as text elements, that use of translated titles together with paraphrased titles as text elements, or the use of translated, decomposed into its sentences, article texts as text elements.

The initial input of the second experiment phase is then lists of triplets $(I_{Image},\ I_{ArticleTitle},\ I_{ArticleText})$ consisting of the article image and translated titles and text. The data is then split according to the split described in Sec. 5.1.2 into training, validation, and testing data. Afterwards, for each of the possible nine (M_{Init}, T_{Type}) $\in$ "Initial Model State x Textual Input" combinations, the following steps are performed.

First the input data is augmented and processed according to the current T_{Type} so that the given triplets are reduced to image-annotation pairs (I_{Image}, I_{Text}). Then the SAEM model is initialized according to the current M_{Init}. Thereafter, the SAEM model is trained for 50 epochs using the generated (I_{Image}, I_{Text}). Subsequently, the best models of the current configurations for the specific metrics are then selected by calculating *ACC@N* and *MRR@N* for $N = 1, ..., 100$ for the matchings of the trained model on the validation data set and choosing the model iterations with the highest computed mean values.

Afterwards, the corresponding metrics are also calculated on the test data and the performances of the baselines are determined on the test data sets.

6 EVALUATION

The results of the experiment introduced in the previous section are now presented in this section. First some illustrative information and examples are given regarding the success of the text data augmentation. The main focus of the section will then be on the comparison of the performance of the different re-matching approaches and an attempt is given to answer the two core questions asked when defining the experiment: how the use of pre-trained models and the different types of text inputs affect the performance.

6.1 Text Data Augmentation

Before the analysis of the actual experiment results it should be shortly highlighted that both data augmentation methods were able to achieve a significant increase at least in the number of training examples. From the original 9364 training examples, 9364 additional training data points could be generated by generating paraphrases,

two paraphrase examples and their original corresponding translated headlines can be found in Tab. 2, and 14682, an increase of 5318, training data points were created by splitting the article texts into complete sentences.

Table 2: Two examples of with Quillbot generated paraphrases and their original (translated) article headlines.

Original (translated) Headline	Paraphrase
Car bomb explodes in Assad stronghold Latakia .	In Assads stronghold of Latakia, a car bomb blasts .
Green Week kicks off: focus on animal welfare and digitization .	The first day of Green Week is dedicated to animal welfare and digitisation .

6.2 Experiment Results

The first results to be shown display the performance of the best models of the configuration combinations and SAEM baselines, that is SAEM models trained exclusively on MS-COCO or Flickr30K, on the validation data set. Fig. 2 represents the ACC@N for $N = 1, ..., 100$ while Fig. 3 shows the MRR@N metric for $N = 1, ..., 100$ for the model candidates of the configuration combinations.

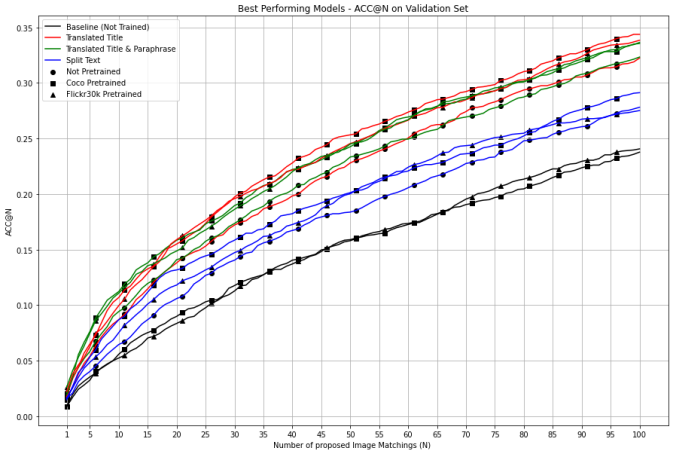

Figure 2: The ACC@N for $N = 1, ..., 100$ results on the validation data set achieved by the best model iterations of each possible model configuration and two SAEM Baselines.

In general, both in Fig. 2 as well as in Fig. 3 it can be seen that on the validation dataset all best iterations of the investigated model configurations can sometimes significantly outperform the best iteration of the examined SAEM baselines. Regarding our core questions, to what extent the use of pre-trained models and the different types of text inputs affect the performance, interesting insights can also be gained. For both metrics, for all three input text types, the worst performing models are the non-pretrained model candidates. With respect to the different text input types, it can also be seen that the article text-based model variant is generally performs worst in both metrics, whereas for the ACC@N a model variant with the

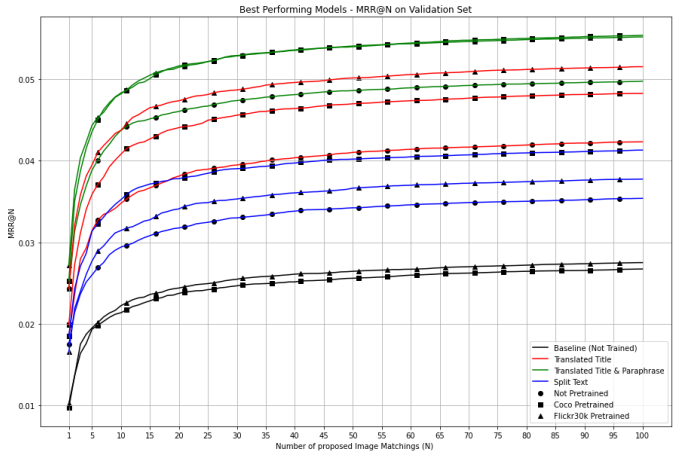

Figure 3: The MRR@N for $N = 1, ..., 100$ results on the validation data set achieved by the best model iterations of each possible model configuration and two SAEM Baselines.

translated titles alone performs best, while regarding MRR@N two model variants trained using translations and paraphrases achieve the best results.

After determining and selecting the best model variants separately for the ACC@N and the MRR@N scenario, the performance of the specific models was subsequently also evaluated on the test data set. Additionally the already used SAEM baselines and the similarity-based Baselines are also evaluated. The corresponding results are now shown in Fig. 4 with respect to the ACC@N for $N = 1, ..., 100$ and in Fig. 5 with respect to MRR@N for $N = 1, ..., 100$.

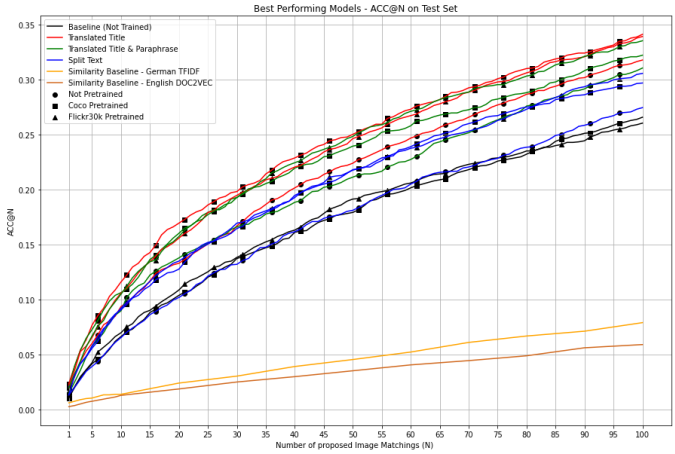

Figure 4: The ACC@N for $N = 1, ..., 100$ results on the test data set achieved by the best model iterations (on the validation data set) of each possible model configuration, two SAEM and two similarity-based Baselines.

Several insights can now be drawn from this. Firstly it can be seen that the similarity-based baselines perform much worse than

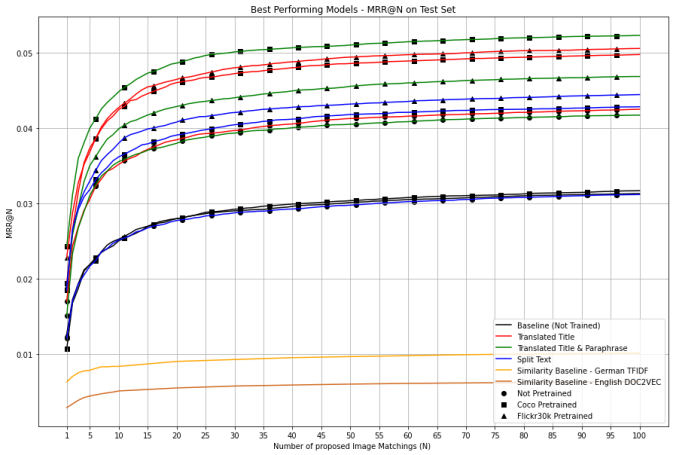

Figure 5: The MRR@N for $N = 1, ..., 100$ results on the test data set achieved by the best model iterations (on the validation data set) of each possible model configuration, two SAEM and two similarity-based Baselines.

any SAEM configuration. Furthermre, the non-pretrained article text-based model candidate obviously performs much worse in generalization on this new unseen dataset, as the performance drops to a level similar to the SAEM baselines. Further the impression is confirmed that the non-pretrained models generally perform worse than the pretrained variants, as also on the test dataset these model variants perform noticeably worse compared to the model candidates with the same input text type. This is especially evident for the MRR@N values in Fig. 5, where the three non-pretrained model variants represent the worst non-baseline models. In general, the variants using only titles and titles and paraphrases seem to perform better than the article text-based variants, whereby two models that already performed best on the validation data, for ACC@N the title-only variant pre-trained with MS-COCO and for MRR@N the title and paraphrase variant pre-trained with MS-COCO, also perform best on the unseen test data set.

7 CONCLUSION

In this final section, we use the observed results of our experiment to draw a conclusion regarding the questions we investigated. We approached this task to investigate in particular two key questions regarding re-matching tasks in general and, in the scope of this project, the MediaEval re-matching task in particular. These refer to the impact of transfer learning and the impact of the presented text data augmentation approaches on the final observable performance.

The results of our experiment now indicates three core findings. First, our results suggest that it is possible to use a SAEM-based approach to solve the proposed task. The metrics achieved by some models of about 34% ACC@100 and over 5% MRR@100 on the test data sets, as well as the partial substantial outperforming of the baselines support this impression. Second, we observe that at least one text augmentation approach, the use of paraphrases, led to increased performance measured by the MRR@100 metric, making text data augmentation approaches at least potentially still worth considering, even if the second, article text-based, approach led to a significant drop in performance. Lastly, our results provide clear indications that the use of transfer learning in this domain has an impact and could lead to improvements in the performance of similar tasks.

In conclusion, when using the SAEM model in combination with transfer learning and text data augmentation approaches to solve a re-matching task, we observed at encouraging results, which suggest potential for future research in this area.

REFERENCES

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 6077–6086.

[2] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering. In *CVPR*.

[3] Wissam Antoun, Fady Baly, Rim Achour, Amir Hussein, and Hazem Hajj. 2020. State of the Art Models for Fake News Detection Tasks. In *2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT)*. 519–524. https://doi.org/10.1109/ICIoT48696.2020.9089487

[4] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450* (2016).

[5] Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python* (1st ed.). O'Reilly Media, Inc.

[6] Claire-Hélène Demarty, Mats Sjöberg, Bogdan Ionescu, Thanh-Toan Do, Michael Gygli, and Ngoc Duong. 2017. Mediaeval 2017 predicting media interestingness task. In *MediaEval workshop*.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* abs/1810.04805 (2018). arXiv:1810.04805 http://arxiv.org/abs/1810.04805

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[9] Yuming Fang, Weisi Lin, Chiew Tong Lau, and Bu-Sung Lee. 2011. A visual attention model combining top-down and bottom-up mechanisms for salient object detection. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 1293–1296.

[10] Andrea Frome, Greg Corrado, Jonathon Shlens, Samy Bengio, Jeffrey Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. 2013. DeViSE: A Deep Visual-Semantic Embedding Model. In *Neural Information Processing Systems (NIPS)*.

[11] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep learning*. MIT press.

[12] David Guthrie, Ben Allison, Wei Liu, Louise Guthrie, and Yorick Wilks. 2006. A closer look at skip-gram modelling.. In *LREC*, Vol. 6. Citeseer, 1222–1225.

[13] Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for sequence tagging. *arXiv preprint arXiv:1508.01991* (2015).

[14] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. 2014. Caffe: Convolutional Architecture for Fast Feature Embedding. *arXiv preprint arXiv:1408.5093* (2014).

[15] Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3128–3137.

[16] Benjamin Kille, Andreas Lommatzsch, and Özlem Özgöbek. 2020. News Images in MediaEval 2020. In *Proc. of the MediaEval 2020 Workshop. Online.*

[17] Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. *CoRR* abs/1408.5882 (2014). arXiv:1408.5882 http://arxiv.org/abs/1408.5882

[18] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). http://arxiv.org/abs/1412.6980

[19] Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. 2014. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539* (2014).

[20] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Fei-Fei Li. 2016. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *CoRR* abs/1602.07332 (2016). arXiv:1602.07332 http://arxiv.org/abs/1602.07332

[21] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. 2018. Stacked cross attention for image-text matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 201–216.

[22] Gaël Letarte, Frédérik Paradis, Philippe Giguère, and François Laviolette. 2018. Importance of self-attention for sentiment analysis. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. 267–275.

[23] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: Common Objects in Context. *CoRR* abs/1405.0312 (2014). arXiv:1405.0312 http://arxiv.org/abs/1405.0312

[24] Jie Lu, Vahid Behbood, Peng Hao, Hua Zuo, Shan Xue, and Guangquan Zhang. 2015. Transfer learning using computational intelligence: A survey. *Knowledge-Based Systems* 80 (2015), 14–23. https://doi.org/10.1016/j.knosys.2015.01.010

[25] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025* (2015).

[26] Durjoy Sen Maitra, Ujjwal Bhattacharya, and Swapan K. Parui. 2015. CNN based common approach to handwritten character recognition of multiple scripts. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*. 1021–1025. https://doi.org/10.1109/ICDAR.2015.7333916

[27] Leeja Mathew and V R Bindu. 2020. A Review of Natural Language Processing Techniques for Sentiment Analysis using Pre-trained Models. In *2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC)*. 340–345. https://doi.org/10.1109/ICCMC48092.2020.ICCMC-00064

[28] Youssef Mroueh, Etienne Marcheret, and Vaibhava Goel. 2015. Asymmetrically Weighted CCA And Hierarchical Kernel Sentence Embedding For Image & Text Retrieval. *arXiv preprint arXiv:1511.06267* (2015).

[29] Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. 2014. Learning and Transferring Mid-level Image Representations Using Convolutional Neural Networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 1717–1724. https://doi.org/10.1109/CVPR.2014.222

[30] Sang-Jae Park, Jang-Kyoo Shin, and Minho Lee. 2002. Biologically inspired saliency map model for bottom-up visual attention. In *International Workshop on Biologically Motivated Computer Vision*. Springer, 418–426.

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

[32] Luis Perez and Jason Wang. 2017. The Effectiveness of Data Augmentation in Image Classification using Deep Learning. *CoRR* abs/1712.04621 (2017). arXiv:1712.04621 http://arxiv.org/abs/1712.04621

[33] Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. ELRA, Valletta, Malta, 45–50. http://is.muni.cz/publication/884893/en.

[34] Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. *arXiv preprint arXiv:1509.00685* (2015).

[35] Cong-ying Shi, Chao-jun Xu, and Xiao-Jiang Yang. 2009. Study of TFIDF algorithm. *Journal of Computer Applications* 29, 6 (2009), 167–170.

[36] Connor Shorten and Taghi M. Khoshgoftaar. 2019. A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data* 6, 1 (July 2019), 60. https://doi.org/10.1186/s40537-019-0197-0

[37] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[38] Jan Theeuwes. 2010. Top–down and bottom–up control of visual selection. *Acta psychologica* 135, 2 (2010), 77–99.

[39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

[40] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. 2017. Residual attention network for image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3156–3164.

[41] Jian Wang, Feng Zhou, Shilei Wen, Xiao Liu, and Yuanqing Lin. 2017. Deep Metric Learning with Angular Loss. *CoRR* abs/1708.01682 (2017). arXiv:1708.01682 http://arxiv.org/abs/1708.01682

[42] Liwei Wang, Yin Li, and Svetlana Lazebnik. 2015. Learning Deep Structure-Preserving Image-Text Embeddings. *CoRR* abs/1511.06078 (2015). arXiv:1511.06078 http://arxiv.org/abs/1511.06078

[43] Jason Wei and Kai Zou. 2019. EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks. *arXiv:1901.11196 [cs]* (Aug. 2019). http://arxiv.org/abs/1901.11196 arXiv: 1901.11196.

[44] Yiling Wu, Shuhui Wang, Guoli Song, and Qingming Huang. 2019. Learning fragment self-attention embeddings for image-text matching. In *Proceedings of the 27th ACM International Conference on Multimedia*. 2088–2096.

[45] Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. 2016. Image captioning with semantic attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4651–4659.

[46] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *TACL* 2 (2014), 67–78.

[47] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. 2018. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5505–5514.

[48] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. 2021. A Comprehensive Survey on Transfer Learning. *Proc. IEEE* 109, 1 (2021), 43–76. https://doi.org/10.1109/JPROC.2020.3004555