WassersteinQA - An Exploration of Generalization using Wasserstein Distance

Matteo Karl Donati UCL
Amir Ghomeshi UCL
Neil Leiser UCL
Zacharie Rozenberg UCL

Abstract

Defining a distance between distributions or datasets is crucial for various machine learning tasks including transfer learning and generalization from one dataset to another. Current methods used in Natural Language Processing (NLP) usually rely on different parameters that have to be optimised or require to train a model. With the recent success of Optimal Transport in computer vision, several approaches have proven their efficiency in identifying similarities between distributions. Some methods are used in NLP for document representation. However, Optimal Transport is still relatively new when it comes to comparing entire text datasets. This can be explained by the fact that the task becomes significantly more complex as the distribution of each dataset strongly depends on the embedding approach. In this paper, we propose a new method based on the Wasserstein Optimal Transport distance which (i) evaluates the distance between Question-Answering (QA) datasets, (ii) is efficient and (iii) is based on strong theoretical footing. (iv) We validate the correctness of our method by showing its compatibility with a transfer learning task.

1 Introduction

To get a notion of distance or similarity between datasets can be useful for multiple supervised and unsupervised Natural Language Processing (NLP) tasks. Some examples include classification, clustering as well as transfer learning.

In transfer learning, a model is trained on a source task and evaluated on a new target task, with or without additional fine tuning. In the latter case it is called generalization. In other words, we store knowledge acquired while trying to solve a specific problem and use this knowledge to tackle new ones. When using transfer learning to evaluate a model on a target dataset, it intuitively seems more appropriate to pretrain this model on a dataset that is to some extent close to that task. This shows the importance of defining a suitable distance between datasets.

Several approaches analyse similarities between datasets to estimate the performance of algorithms on transfer learning tasks. Current methods in Machine Learning involve comparing the learning curves of datasets (Leite and Brazdil, 2005) or embedding datasets based on the Fisher information matrix (Achille et al., 2019) as well as other approaches (Khodak et al., 2019; Tran et al., 2019). The downside of most of the papers mentioned above is that a model has to be trained for each dataset and optimal parameters have to be computed. In order to overcome those issues, Alvarez-Melis and Fusi (2020) propose an optimal transport distance between datasets (OTDD) and apply their method on a range of datasets including text data. However, the NLP section of their paper focuses on transfer learning across text classification datasets only. Their approach does not mention any application on more challenging NLP datasets like QA where contexts as well as questions have to be considered as relevant features.

In this research, we propose a new approach to compute the distance between QA datasets based on the optimal transport Wasserstein distance (Villani, 2016). The Wasserstein distance has recently been very popular in training Generative Models (Arjovsky et al., 2017; Gulrajani et al., 2017; Liu et al., 2018; Dukler et al., 2019) but is not often used in the domain of NLP. We apply this distance to the embedded features of ten QA datasets in order to quantify their similarities.

We also compare our results with those obtained by Talmor and Berant (2019). In their paper, the authors evaluate generalization and transfer learning over ten QA datasets using two different models: DocQA and BertQA. However, their analysis

requires to train a model on every single dataset. We show that our approach, while being relatively simple and computationally efficient, can quantify generalisation from one dataset to another.

To summarise, we make the following contributions:

- We propose a new method to compute the distance between QA datasets.
- Our approach is simple, computationally efficient and is based on strong theoretical footings.
- We show that the Wasserstein distance between QA datasets can be related to transfer learning by comparing our results with Talmor and Berant (2019). Datasets which are closest to each other will tend to generalise better on transfer learning tasks.

2 Related Work

2.1 Document Representation

Measuring the distance between two distinct datasets in the domain of NLP is closely related to documents representation. Early work in this field include Salton and Buckley (1988) as well as Robertson and Walker (1994) who study efficient termweighting systems for improved documents indexing. Blei et al. (2003) describe Latent Dirichlet Allocation, a generative probabilistic Bayesian model which given a document, estimates the probability of generating a word.

Later on, deep unsupervised learning techniques are used in order to learn representation of texts. Glorot et al. (2011) implement a Stacked Denoising Autoencoder which learns to extract meaningful characteristics of documents. Chen et al. (2012) propose an mSDA which is more scalable and reduces computational costs. With the success of word2vec (Mikolov et al., 2013), Le and Mikolov (2014) propose paragraph vector embeddings (doc2vec) which can learn fixed lengths representations of sentences, texts and documents and surpasses the bag of words as well as several other approaches.

The Word Mover's distance (Kusner et al., 2015) measures the minimum cumulative distance between words of two documents. This Optimal Transport approach can be described as an example of the Earth's Mover Distance where each document is represented by a list of embedded words. Other variants include S-WMD (Huang et al., 2016) and the faster RWMD (Atasu et al., 2017). However, in Kusner et al. (2015), the WMD distance is based on word2vec which is able to create only one embedding per word, regardless of the different contexts in which the word may be found.

With the recent improvements made in machine learning and NLP, several embedding techniques are now producing contextualised word representation like ELMo (Peters et al., 2018), GPTv1 (Radford, 2018) and the famous BERT (Devlin et al., 2019) which achieves state-of-the-art in a range of NLP tasks. Those embeddings can be used to further improve WMD and other approaches. Recent progress in document representation using BERT include BERTgrid (Denk and Reisswig, 2019) for 2D document representation, DocBERT for document classification (Adhikari et al., 2019) and BERT-AL (Zhang et al., 2020) for long document understanding.

2.2 Optimal Transport Distances Between Datasets

In Optimal Transport, datasets are represented by complex distributions which are then compared using the notion of distance. Optimal Transport has had a recent success, particularly in computer vision with Courty et al. (2014) using OT for domain adaptation, Salimans et al. (2018) who propose OT-GAN and measure the distance between the generator and the discriminator distribution to improve GANs, Snow and lent (2018) who use the Wasserstein distance for image comparison as well as Bunne et al. (2019) who rely on the Gromov-Wasserstein distance to learn generative models across different feature spaces.

With regards to datasets containing text, only a few papers consider optimal transport for dataset comparison. Alvarez-Melis and Jaakkola (2018) apply the Gromov-Wasserstein distance to measure similarities between pair of words across different languages and use this approach for unsupervised language-translation tasks. Yurochkin et al. (2019) use a Hierarchical Optimal Transport distance between documents where each document is modeled as a distribution of topics and each topic is represented as a distribution of words. More related to our work is the paper from Alvarez-Melis and Fusi (2020) on Geometric

dataset distances via optimal transport. They define the optimal transport dataset distance (OTDD) to measure the similarity between different sets of feature-label pairs and investigate the relations between the OTDD distance and transfer learning tasks. However, the authors focus their work on news and reviews data while we focus our research on QA datasets using a different approach.

Other interesting distances between datasets which do not rely on Optimal Transport include discrepancy distances (Mansour et al., 2009; Cortes and Mohri, 2011) as well as distances based on summary statistics (Tatti, 2007).

3 Background On Optimal Transport

Let $\mathcal{M}_1^+(\mathcal{X})$ be the space of probability measures on $\mathcal{X}$, where $\mathcal{X} \subset \mathbb{R}^d$. In our case, $\mathcal{X}$ is the embedding space. We consider discrete uniform measures of the form $\mathbb{P}_x = \frac{1}{N}\sum_{i=1}^{N} \delta_{x_i}$, such that each x_i is a sentence embedded in $\mathcal{X}$. Optimal transport distances lift the idea of closeness between points in the Euclidean sense to the estimation of the closeness between probability measures. For instance, the Wasserstein distance between measures $\mathbb{P}_x, \mathbb{P}_y \in \mathcal{M}_1^+(\mathcal{X})$ is of the following form (Kantorovich, 1958):

$$W(\mathbb{P}_x, \mathbb{P}_y) = \min_{\pi \in \Pi(\mathbb{P}_x, \mathbb{P}_y)} \mathbb{E}_{(x,y)\sim\pi} c(x, y) \quad (1)$$

where $c : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a ground cost function defined on the data space, and $\Pi(\mathbb{P}_x, \mathbb{P}_y)$ is the set of joint measures with marginals $\mathbb{P}_x$ and $\mathbb{P}_y$. More intuitively, the Wasserstein distance consists of the minimal expected cost of transporting mass from $\mathbb{P}_x$ to $\mathbb{P}_y$, and the ground cost is typically euclidean.

However (1) is typically extremely challenging to estimate, and Cuturi (2013) introduces entropic regularization which smooths the objectives and enables the use of the massively parralelizable Sinkhorn algorithm. The entropic-regularized Wasserstein is of the following form:

$$W_\epsilon(\mathbb{P}_x, \mathbb{P}_y) = \min_{\pi \in \Pi(\mathbb{P}_x, \mathbb{P}_y)} \mathbb{E}_{(x,y)\sim\pi} c(x, y) - \epsilon h(\pi) \quad (2)$$

where $h(\pi) = -\int_{\mathcal{X}\times\mathcal{Y}} \pi(x, y) \log(\pi(x, y)) dx dy$ is the entropy of the transport plan π, and we use the algorithm from Cuturi (2013) to compute it.

4 Motivation

The purpose of this paper is to explore how the Wasserstein distance can be applied to QA datasets.

First, we consider the relatively simple task of taking the Wasserstein distance between sets of documents from the 20 Newsgroups corpus. This is to confirm our intuition that the Wasserstein distance can indeed be used to quantify the similarity between different NLP datasets. The 20 Newsgroups dataset contains 20000 documents spread evenly across 20 topics. By using the methods that we describe in section 5, we display a visualization of the distances in Figure 1. We can observe that all the topics belonging to a specific category are indeed close to each other on the graph. We can thus infer that by choosing suitable embeddings, the Wasserstein distance can be used to distinguish similarities between sets of features in NLP datasets.

In the following sections of this paper we consider the Wasserstein distance to compare QA datasets which is a more challenging task. To illustrate the validity of these distances, we will show that there exists some correlation between the Wasserstein distance of two QA datasets and the quality of generalization from one to the other.

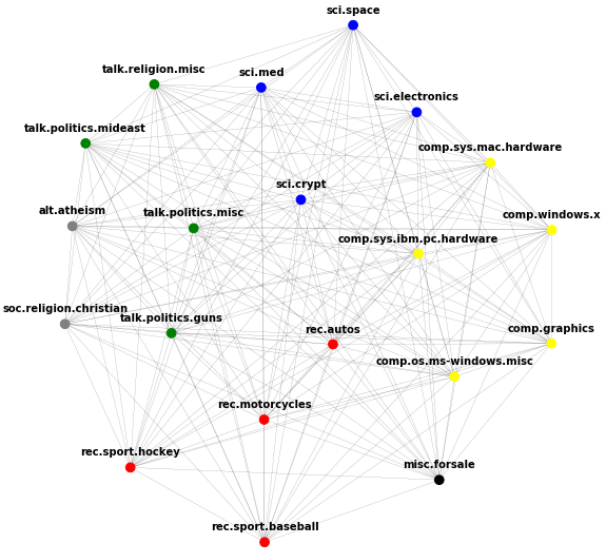

Figure 1: Graph representation of the distance matrix obtained using the Wasserstein distance for the 20newsgroups dataset. Each point represents a topic and each color represents a category.

5 Methodology

In this section we present different aspects of our study that have to be considered before these measurements can be carried out.

5.1 Reading Comprehension Datasets

For our research, we consider 10 RC (or QA) datasets which are also used by Talmor and Berant (2019) in their investigation of generalization across reading comprehension (RC) datasets. Information regarding those datasets can be found in Table 1. Each dataset provides context-question-answer triples.

Dataset	Size	Context	Question	Multi-hop
SQUAD	108K	Wikipedia	crowd	No
NEWSQA	120K	Newswire	crowd	No
SEARCHQA	140K	Snippets	trivia	No
TQA-W	95K	Snippets	trivia	No
HOTPOTQA	113K	Wikipedia	crowd	Yes
CQ	2K	Snippets	Web queries/KB	No
CWQ	35K	Snippets	crowd/KB	Yes
COMQA	11K	Snippets	WikiAnswers	No
WIKIHOP	51K	Wikipedia	KB	Yes
DROP	96K	Wikipedia	crowd	Yes

Table 1: QA Datasets (Talmor and Berant, 2019)

In their paper MultiQA, Talmor et al. explore the issue of transferability and generalization between those 10 RC datasets. For each dataset they pre-train on a source dataset, and then fine-tune the model using a small amount of data from the target dataset or simply test their model on a target dataset using zero-shot learning. In our research, we experiment with these same RC datasets, and explore the possibility of predicting generalization and transfer performance, using the Wasserstein distance measure.

5.2 Dataset as a distribution

The Wasserstein distance measures the distance between 2 distributions. In order to apply this distance to datasets, a way of representing them as distributions is needed. As implemented by Alvarez-Melis and Fusi (2020), it is common to represent a distribution as a matrix where each instance of the dataset makes up a row or a column of that matrix.

From now on, we focus exclusively on the RC datasets presented in the previous section. Each dataset is composed of multiple instances that are themselves subdivided in 3 structures as explained in 5.1 (triples). In our study, we only use two of these structures in order to calculate the distance : the context and the question. Our intuition was that similarity between different datasets would strongly depend on (1) the manner and style the contexts are written, (2) the way the questions are asked and (3) the way each one interfere with the other. Because all the datasets we analyze only have extractive questions, i.e. the answer is a span in the context, we believe that the answers have less impact on the distances and so we decide not to include them in our research without refuting their potential influence. In the following subsections we discuss different methods we explored to represent these instances as vectors that can be stacked together to form a matrix representation of the dataset.

5.3 Instances embeddings

To represent a whole document as a single vector is called document embedding and is still an active field of study. In recent years, the effort of producing a mapping from documents to informative vectors in $\mathbb{R}^n$ has resulted in numerous new methods with various innovative solutions to the problem and some notable breakthroughs. In our case, embedding the instances has yet another complication in that each one consists of two distinct structures that we must interpolate together in some way and embed as a single vector: the context and the question. We show here two simple and intuitive methods that we use to map our instances to vectors as well as the justifications which make these approaches suitable. Both methods proved themselves adequate to our purpose and allow us to demonstrate how we can indeed quantify the similarity between the datasets using the Wasserstein distance.

5.3.1 Averaging contextual word embeddings based on BERT

A classic approach to embed a document is to summarize its word vectors using some weighing scheme. As Palachy (2019) explains in his review "Document Embedding Techniques", this approach is valid especially when used with the most state-of-the-art word representations (usually by averaging instead of summing) and can stand its ground against more complex methods. Making use of this, we embed our instances using a pre-trained BERT model from the Transformers library built by the Hugging Face team. This approach seemed adequate to us because in their paper, Talmor et al. use a model based on BERT (Devlin et al., 2019) to carry their experiments, with which we later compare our results. A great characteristic of BERT word embeddings is that they

provide the powerful contextual representations required for this averaging approach to work. Another big advantage of the Transformers library is that they provide pre-trained models optimized for different kind of NLP tasks including question answering. This model, called BertForQuestionAnswering, allows us to automatically extract contextual word embeddings for each token of both the context and the question without having to design a complex way to merge them.
We use the pre-trained BERT large model which has 340 million parameters. From this model we are able to extract a 1024-dimensional vector for each token. To get a single vector representation of the instance, we average these vectors.

5.3.2 Sentence embedding based on SBERT

We also consider a different approach where we embed whole datasets using sentence-level embeddings. Our intuition was that embedding sentences instead of words may capture different similarities between datasets because it provides a rich representation of each sentence by taking into account their surrounding context as well as their syntax. We base our approach on the one described by Alvarez-Melis and Fusi (2020) where they take advantage of pre-trained models to embed sentences in vector space. In their research they find that embedding every sentence of every dataset using a pre-trained model enables a meaningful measurement of the distances between different datasets.

For our purposes we use the sentence_transformer library to extract the sentence embeddings. This pre-trained model is based on SBERT (Reimers and Gurevych, 2019) which finetunes BERT in order to produce meaningful sentence embeddings that are suitable for unsupervised tasks like semantic similarity comparison or clustering. It takes as input a sentence and outputs a 768-dimensional vector. SBERT is trained on both SNLI (Bowman et al., 2015) and Multi genre NLI (Williams et al., 2018) datasets. The SNLI includes 570,000 sentence pairs and the algorithm has to predict whether the sentences are contradictory neutral or entailing. MultiNLI contains 430,000 sentence pairs which includes a large sample of different genres of spoken and written text (Reimers and Gurevych, 2019). In addition, it uses siamese and triplet network structures which makes the embedding process fast and efficient.

In many of our RC datasets, the number of sentences in the context is much larger than in the question. Hence, we believe that embedding both the context and the question together will implicitly put a high weight on the contexts of the datasets. Also, by stacking every context and question together we lose the control over the way we want to interpolate them together. For these reasons we split the questions and contexts of each dataset and embed them separately as we explain in more details in the experiment section of this paper.

6 Experiments

The two embedding approaches described in the previous section give us an intuitive way of representing datasets and allow us to compute the entropic-regularized Wassertsein distance between them. In this paper we perform the following experiment: for both embedding approaches separately, we embed every instances of a dataset (for every dataset) and then measure the distances between all datasets. We also evaluate the quality of our results by comparing them with Talmor and Berant (2019). As we will show in the results section of this paper, we indeed find that there exists a clear correlation between the Wasserstein distance of two datasets and the efficiency of generalization from one to the other.

6.1 Representing RC datasets as a matrix

For the word-level embedding method, we follow the standard implementation where the input is a sequence of maximum 512 tokens composed of the question and the context separated by special tokens : [CLS] $<$ question $>$ [SEP] $<$ context $>$ [SEP]. Some contexts were longer than 512 tokens and for these we simply took the first chunk of length 512. We do that for 1000 instances of each of the 10 datasets that we tokenize using the open source BERT uncased tokenizer. From this pre-trained model we had the choice of extracting the word representations from the output of each of the 24 layers of the BERT model. We empirically found that the output of the embedding layer (first layer) is the representation that best matches our needs. Intuitively this makes sense because every layer does some multi-headed attention computation on the word representation of the previous layer to create a new intermediate representation. The goal of these successive computations is

to allow the softmax classifier in the end to locate the answer span with precision. That means that as we go deeper in terms of layers, the word representations carry more information about the location of the answer span and probably less on the raw contextual word embedding that we are looking for. To get a single vector as a representation of the whole instance we average all 512 vectors. We thus represent a whole dataset as a $\mathbb{R}^{n \times d}$ matrix where n is the number of instances that we consider (1000 in our case) and d is the embedding dimension (1024 in our case).

In the sentence embedding approach we embed the contexts and the questions separately. The questions are always single sentences so we only need to embed them using the sentence transformer library. The contexts first need to be tokenized into sentences before being embedded. This way, we extract 2000 sentences (chosen randomly) of both contexts and questions for each of the 10 datasets and map each one of them into a vector representation. We have thus split the initial datasets into a set of embedded questions represented by $\mathbb{R}^{n \times d}$ matrices where n is the number of questions (2000 in our case) and d is the embedding dimension (768 in our case). Similarly, we get a set of embedded contexts represented by matrices with the same dimensions.

6.2 Wassertstein distance for QA datasets

We compute the distance between every pair of datasets using the entropic-regularized Wasserstein distance. For the first approach this is immediate as each dataset is represented by a single matrix. For the second approach we compute the distance between the questions and the contexts separately. In order to get a single distance between each pair of datasets, a weighted average between both distances has to be considered. However, finding the optimal weighing is a complex question that involves a considerable number of parameters. In our research, we set the mean between these two distances as the final distance between two datasets. Further investigation is required in order to find a more efficient approach and this will be discussed in the last section of this paper.

In order to deal with the computational limitations, we attempted some dimensionality reduction techniques such as Principal Components Analysis (PCA) on our sentence embeddings such as that of Raunak (2019) and Mu et al. (2017), where the top principal components are removed from the embeddings. We hoped this would allow us to take larger samples, without removing too much information. To do this effectively required us to find the best combination of dimension size and number of principal components to remove from the embeddings. However, we did not find that we were able to uncover more meaningful results through this approach. In fact, we observed that increasing the number of samples does not significantly change the distance. For example, with the word-level embedding approach, we get almost the exact same distance relation using 100, 1000 or 4000 instances.

For each distance measurement, we set the epsilon parameter (entropy) to 1 and the maximum number of iterations to 100.

7 Results and Discussion

In this section we present our final results following both approaches. We also compare them with the results obtained in Talmor and Berant (2019) and try to interpret the relation between them. Overall, the results are quite conclusive : for more than half of the datasets, we can clearly see a correlation between our distance and the quality of generalization demonstrated by their research. More precisely we find that in both cases, for at least 5 datsatets out of 10, the most effective generalization is indeed between datasets that are the closest according to the Wasserstein distance.
However, there is a non-negligible portion of the datasets for which our results are not agreeing with theirs. In this section we also attempt to discuss the reasons why we think this might be the case.

7.1 1st approach : word-level embeddings

We arrange our distances in a table using the same format as Talmor and Berant used in their research in order to make it easier to visualize the correlation between their results and ours. For their results (Table 2) and for each dataset, we highlight in bold the highest score while for our results (Table 3 and 4), we highlight the smallest distance with another dataset.

In the uppermost table we can see that CQ, CWQ and ComQA generalize best from SearchQA which is indeed the dataset with which their distance is minimal. The same parallelism can be observed from TQA-W to Wikihop, TQA-

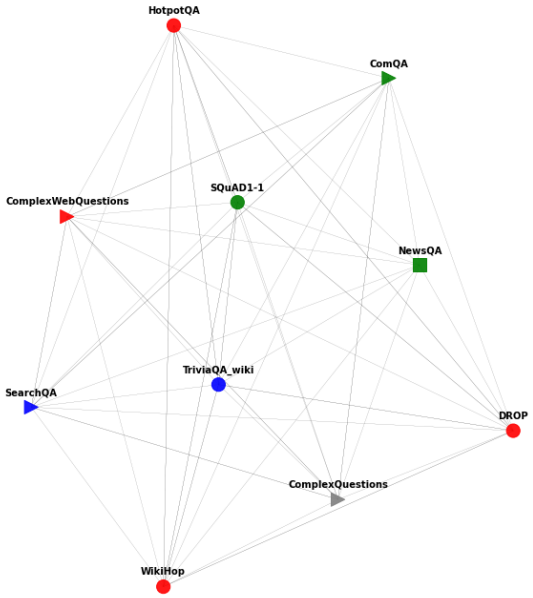

Figure 2: A 2-D visualization of the distances obtained with the sentece-level approach. We can observe that factoid datasets (SQuAD, NewsQA and ComQA) are clustered nearby as well as trivia datasets (TQA-W and SearchQA) which are also found close to one another.

W to SearchQA and from NewsQA to SQuAD. For the other datasets, even though we cannot rely on our distance to highlight the datasets which have the best generalization, we can still observe interesting trends such as the very poor generalization from SearchQA to Drop with SearchQA being indeed the furthest dataset from Drop.

7.2 2nd approach : sentence-level embeddings

The results obtained using the sentence transformer embedding are displayed in the same table after we rescaled them by a factor of $10^{-}2$..The scale of these measurements differs from the one of the first approach. This is probably explained by the different features emphasized by each method on which the Wasserstein distance makes its computations.

This time, we get the the correlation described earlier for 50% of the datasets while it is hard to conclude anything for the others. Still, this means that for half of the datasets, the Wasserstein distance gives an indication on the dataset from which it is best to generalize. Computing the distance using sentence-based embedding also highlights characteristics that we expected to see such as the closeness of SQuAD and NewsQA as each dataset contains questions on single documents.

7.3 Suboptimality of the results

It is difficult to make further observations from the obtained results without taking the risk of overinterpreting them. Indeed, although the correlation between the Wasserstein distance of 2 datasets and their ability to generalize between them evidently shows great potential, we must keep in mind the concessions we made to achieve these results.

First, it is important to consider the fact that in our research, we are investigating a distance between datasets. This task is significantly different from transfer learning or generalisation and hence, getting the exact same results as in Talmor and Berant (2019) was not something we expected. In fact, many factors and parameters can influence the results obtained by training a model including learning rate, batch size, number of epochs and many more. Those are hyper-parameters that we do not control and could influence generalization from one dataset to another. However, we wanted to show that for most of the QA datasets, choosing the closest datasets will in general lead to better generalization errors. In this study we focus only on the contexts and the questions and do not consider the targets. This could explain for example the discrepancy bewteen ours and Talmor and Berant (2019) results regarding the HotpotQA distances. Using the sentence-level embedding approach, the Wasserstein distance between HotpotQA and most of the datasets is the largest, but the accuracy on those datasets after training a model on HotpotQA is relatively high. Our intuition for this phenomenon is that the labels of HotpotQA are multi-hop while they are not for the majority of the rest . Including somehow this feature in the matrix representation of the datasets could probably mitigate this disparity.

Also, although we considered many different ways of representing these datasets as distributions, there exist an incredibly large number of methods to achieve that. As we mentioned in the methodology section, document embedding is still an open research and these results can considerably be improved with more complex document embedding techniques.

Other hindrances to better results are probably the computational sacrifices and choices we made along the way. For example, it was not possible to embed whole datasets so we had to comply with embedding a few thousand only. We also restricted ourselves to a single question by context. For the

	CQ	CWQ	COMQA	WIKIHOP	DROP	SQUAD	NEWSQA	SEARCHQA	TQA-W	HOTPOTQA
SQUAD	23.6	12.0	20.0	4.6	5.5	-	**31.8**	8.4	33.4	**11.8**
NEWSQA	24.1	12.4	18.9	7.1	4.4	**60.4**	-	10.1	28.4	8.0
SEARCHQA	**30.3**	**18.5**	**25.8**	12.4	2.8	23.3	12.7	-	**35.4**	5.2
TQA-W	**30.3**	16.5	23.6	**12.6**	5.1	35.5	19.4	**27.8**	-	8.7
HOTPOTQA	27.7	15.5	22.1	10.2	**9.1**	54.5	25.6	19.6	34.9	-

Table 2: MultiQA results - Exact match on the development set for all datasets in a zero-shot training setup (Talmor and Berant, 2019)

	CQ	CWQ	COMQA	WIKIHOP	DROP	SQUAD	NEWSQA	SEARCHQA	TQA	HOTPOTQA
SQUAD	22.63	14.25	15.40	12.73	12.21	-	12.63	15.85	13.14	14.35
NEWSQA	17.04	7.76	9.34	6.50	**9.89**	**12.63**	-	13.91	**6.37**	8.03
SEARCHQA	**14.9**	**6.53**	**7.51**	9.17	14.26	15.85	13.91	-	7.28	16.35
TQA	16.70	7.67	9.22	**5.86**	11.12	13.13	**6.38**	**7.28**	-	**7.37**
HOTPOTQA	15.02	7.40	8.53	6.81	12.59	14.35	8.03	16.35	7.37	-

Table 3: Word-level embeddings results - Wasserstein distances

	CQ	CWQ	COMQA	WIKIHOP	DROP	SQUAD	NEWSQA	SEARCHQA	TQA-W	HOTPOTQA
SQUAD	2.22	2.06	2.30	2.14	2.36	-	**1.87**	2.19	**1.88**	2.37
NEWSQA	2.27	2.09	2.35	2.26	2.38	**1.87**	-	2.27	2.01	2.51
SEARCHQA	2.31	2.10	2.36	2.36	2.49	2.19	2.27	-	2.05	2.37
TQA-W	**2.11**	**2.00**	**2.22**	**2.12**	**2.31**	1.88	2.01	**2.05**	-	**2.19**
HOTPOTQA	2.24	2.28	2.28	2.56	2.56	2.37	2.51	2.37	2.19	-

Table 4: Sentence-level embeddings results - Wasserstein distances (values scaled by 10^{-2})

word-level embedding method we sometimes had to crop a context to fit the 512 tokens restriction. A better way to do that would have been to sort all 512 chunks of a context according to strict rules and merge them using a similar method as Talmor and Berant did in their research.

Because after all this, we still get results that are much better than a randomized classification, we strongly believe in the potential of our research.

8 Conclusion and Future Work

We have shown that the Optimal Transport Wasserstein distance can be used to represent (dis)similarities between QA datasets. This distance strongly depends on the embedding approach and it is hence important to consider adequate embeddings when trying to represent datasets as a set of vectors. We have also shown that the Wasserstein distance can be used to choose suitable datasets for zero-shot transfer learning tasks. When evaluating a model on a reading comprehension task, and given a set of other QA datasets, it can in fact be interesting to pretrain a model on the QA dataset which is closest to the RC task of our interest in order to get better results.

There are many possible extensions to this work. In our research, we only consider the features of each RC dataset before computing the distance. An interesting direction of research would be to consider the labels (answers) of QA datasets when computing the distance similarly to Alvarez-Melis and Fusi (2020). We believe that developing an approach which takes into account the nature of the answer; including whether it is extractive or not, multi-hop or not; will enable better comparison between QA datasets. Another direction would be to investigate the importance of the different features (questions and contexts) when representing a dataset. In one approach, we decide to take the average Wasserstein distance between the contexts and the questions of two datasets in order to compute the final distance. However, further analysis is required in order to come up with a more efficient approach defining suitable parameters which will make the connection between questions and contexts more robust. Finally, when calculating the Wasserstein distance, we use an euclidean cost function. However, different costs functions or distances can be considered. Some suggestions consist of including the cosine similarity in the cost function or using the Gromow-Wasserstein distance if the distance has to be computed across incomparable spaces.

References

Alessandro Achille, Michael Lam, Rahul Tewari, Avinash Ravichandran, Subhransu Maji, Charless C. Fowlkes, Stefano Soatto, and Pietro Perona. 2019. Task2vec: Task embedding for meta-learning. In *The IEEE International Conference on Computer Vision (ICCV)*.

Ashutosh Adhikari, Achyudh Ram, Raphael Tang, and Jimmy Lin. 2019. Docbert: Bert for document classification. *ArXiv*, abs/1904.08398.

David Alvarez-Melis and Nicolò Fusi. 2020. Geometric dataset distances via optimal transport.

David Alvarez-Melis and Tommi Jaakkola. 2018. Gromov-Wasserstein alignment of word embedding spaces. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1881–1890, Brussels, Belgium. Association for Computational Linguistics.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 214–223, International Convention Centre, Sydney, Australia. PMLR.

K. Atasu, T. Parnell, C. Dünner, M. Sifalakis, H. Pozidis, V. Vasileiadis, M. Vlachos, C. Berrospi, and A. Labbi. 2017. Linear-complexity relaxed word mover's distance with gpu acceleration. In *2017 IEEE International Conference on Big Data (Big Data)*, pages 889–896.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Charlotte Bunne, David Alvarez-Melis, Andreas Krause, and Stefanie Jegelka. 2019. Learning generative models across incomparable spaces. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 851–861, Long Beach, California, USA. PMLR.

Minmin Chen, Zhixiang Eddie Xu, Kilian Q. Weinberger, and Fei Sha. 2012. Marginalized denoising autoencoders for domain adaptation. In *ICML*. icml.cc / Omnipress.

Corinna Cortes and Mehryar Mohri. 2011. Domain adaptation in regression. In *Proceedings of the 22nd International Conference on Algorithmic Learning Theory*, ALT'11, page 308–323, Berlin, Heidelberg. Springer-Verlag.

Nicolas Courty, Remi Flamary, Alain Rakotomamonjy, and Devis Tuia. 2014. Optimal transport for domain adaptation. In *NIPS, Workshop on Optimal Transport and Machine Learning*, Montral, Canada.

Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 2292–2300. Curran Associates, Inc.

Timo I. Denk and Christian Reisswig. 2019. {BERT}grid: Contextualized embedding for 2d document representation and understanding. In *Workshop on Document Intelligence at NeurIPS 2019*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Yonatan Dukler, Wuchen Li, Alex Lin, and Guido Montufar. 2019. Wasserstein of Wasserstein loss for learning generative models. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1716–1725, Long Beach, California, USA. PMLR.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *ICML*, pages 513–520. Omnipress.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. 2017. Improved training of wasserstein gans. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5767–5777. Curran Associates, Inc.

Gao Huang, Chuan Guo, Matt J. Kusner, Yu Sun, Fei Sha, and Kilian Q. Weinberger. 2016. Supervised word mover's distance. In *NIPS*, pages 4862–4870.

L. V. Kantorovich. 1958. On the translocation of masses. *Journal of Mathematical Sciences*, 133(4):1381–1382.

Mikhail Khodak, Maria-Florina F Balcan, and Ameet S Talwalkar. 2019. Adaptive gradient-based meta-learning methods. In H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 5917–5928. Curran Associates, Inc.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From word embeddings to document distances. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 957–966, Lille, France. PMLR.

Quoc V. Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *ICML*, volume 32 of *JMLR Workshop and Conference Proceedings*, pages 1188–1196. JMLR.org.

Rui Leite and Pavel Brazdil. 2005. Predicting relative performance of classifiers from samples. pages 497–503.

Huidong Liu, Xianfeng GU, and Dimitris Samaras. 2018. A two-step computation of the exact GAN Wasserstein distance. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 3159–3168, Stockholmsmässan, Stockholm Sweden. PMLR.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. 2009. Domain adaptation: Learning bounds and algorithms. In *COLT*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

Jiaqi Mu, Suma Bhat, and Pramod Viswanath. 2017. All-but-the-top: Simple and effective postprocessing for word representations. *ArXiv*, abs/1702.01417.

Shay Palachy. 2019. Document embedding techniques.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Alec Radford. 2018. Improving language understanding by generative pre-training.

Vikas Raunak. 2019. Effective dimensionality reduction for word embeddings. In *RepL4NLP@ACL*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

S. E. Robertson and S. Walker. 1994. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *In Proceedings of SIGIR'94*, pages 232–241. Springer-Verlag.

Tim Salimans, Han Zhang, Alec Radford, and Dimitris Metaxas. 2018. Improving gans using optimal transport. Cite arxiv:1803.05573.

Gerard Salton and Chris Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Inf. Process. Manag.*, 24(5):513–523.

Michael Snow and Jan Van lent. 2018. The monge-kantorovich optimal transport distance for image comparison.

Alon Talmor and Jonathan Berant. 2019. MultiQA: An empirical investigation of generalization and transfer in reading comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4911–4921, Florence, Italy. Association for Computational Linguistics.

Nikolaj Tatti. 2007. Distances between data sets based on summary statistics. *J. Mach. Learn. Res.*, 8:131–154.

A. T. Tran, C. V. Nguyen, and T. Hassner. 2019. Transferability and hardness of supervised classification tasks. *arXiv:1908.08142*.

C. Villani. 2016. *Optimal Transport: Old and New*. Grundlehren der mathematischen Wissenschaften. Springer Berlin Heidelberg.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Mikhail Yurochkin, Sebastian Claici, Edward Chien, Farzaneh Mirzazadeh, and Justin M Solomon. 2019. Hierarchical optimal transport for document representation. In H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 1601–1611. Curran Associates, Inc.

Ruixuan Zhang, Zhuoyu Wei, Yu Shi, and Yining Chen. 2020. {BERT}-{al}: {BERT} for arbitrarily long document understanding.