# Optimal Partial Transport based Sentence Selection for Long-form Document Matching

**Weijie Yu**[1], **Liang Pang**[4], **Jun Xu**[2,3,*] , **Bing Su**[2,3], **Zhenhua Dong**[5] and **Ji-Rong Wen**[2,3]
[1]School of Information, Renmin University of China
[2]Gaoling School of Artificial Intelligence, Renmin University of China
[3]Beijing Key Laboratory of Big Data Management and Analysis Methods
[4]Institute of Computing Technology, Chinese Academy of Sciences
[5]Noah's Ark Lab, Huawei
{yuweijie, junxu, bingsu, jrwen}@ruc.edu.cn, pangliang@ict.ac.cn, dongzhenhua@huawei.com

## Abstract

One typical approach to long-form document matching is first conducting alignment between cross-document sentence pairs and then aggregating all of the sentence-level matching signals. However, this approach could be problematic because the alignment between documents is *partial* — despite two documents as a whole are well-matched, most of the sentences could still be dissimilar. Those dissimilar sentences lead to spurious sentence-level matching signals which may overwhelm the real ones, increasing the difficulties of learning the matching function. Therefore, accurately selecting the key sentences for document matching is becoming a urgent problem. To address this issue, we propose a novel matching approach that equips existing document matching models with an Optimal Partial Transport (OPT) based component, namely OPT-Match, which selects the sentences that play a major role in matching. Enjoying the partial transport properties of OPT, the selected key sentences can not only effectively enhance the matching accuracy, but also be explained as the rationales for the matching results. Extensive experiments on four publicly available datasets demonstrated that existing methods equipped with OPT-Match consistently outperformed the corresponding underlying methods. Evaluations also showed that the sentences selected by OPT-Match were consistent with human-provided rationales.

## 1 Introduction

Long-form document matching, which identifies the semantic relationship between a source document and a target document, has become a fundamental problem in both NLP and IR. Representative tasks include cite recommendation (Jiang et al., 2019) and plagiarism detection (Foltýnek et al., 2020) etc. For example, in cite recommendation, the document matching has been used to recommend existing papers to be cited in a new paper. In plagiarism detection, the document matching model has been used to determine whether a paper is plagiarized from another paper.

**Document 1:** ... *the attraction of medicinal and aromatic plants as worthy farm crops has grown due to the demand created by consumer interest in these plants for culinary, medicinal, and other anthropogenic applications* ... an understanding of future opportunities in the medicinal and aromatic plant industry who 2003 is necessary to enable us growers to envision and invest in medicinal and aromatic crops that will meet market demands.

**Document 2:** *medicinal and aromatic plants are nowadays becoming the main constituents for medicines, colorings, preservatives, and fragrances*... although it is of high priority, the effect of slowgrowth conservation on the chemical profile of prestored medicinal plants after conservation was not extensively investigated .this research aimed to analyze in vitro stored thymbra spicata l. var. ...

Figure 1: A pair of matched long-form documents from S2ORC dataset. Documents 1 focuses on the future opportunities in the medicinal and aromatic plant industry. Document 2 studies the vitro storage of spicata. Though most sentences are not similar, Document 1 cites Document 2 (matched) because they both take medicinal and aromatic plants as examples (the highlighted sentences).

Existing approaches to long-form document matching typically follow the paradigms developed for short-text (sentence) matching, i.e., they conduct matching based on all of the sentences in the documents. For example, Jiang et al. (2019); Pappagari et al. (2019); Zhou et al. (2020) map the documents into a latent semantic space from a hierarchical perspective (e.g., word, sentence, paragraph) and conduct matching in the semantic space. However, these methods ignore the fact that a long-form document usually contains multiple paragraphs and sentences, which convey complex and diverse semantics. Unlike short-text matching where almost every word-level matching signal matters, in long-form document matching, the alignment between a document pair is partial and a few matching signals

[*] Corresponding author

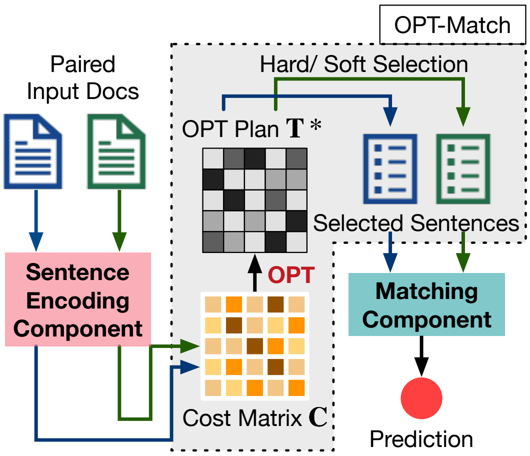

Figure 2: The architecture of OPT-Match. The sentence encoding component and the matching component are with existing methods (the underlying models).

between the key sentences can determine the matching result at the document level. Figure 1 illustrates a typical example: two matched paper abstracts (*Document 1* cites *Document 2*) from the Semantic Scholar Open Research Corpus (S2ORC) (Lo et al., 2020). These two abstracts are matched because of the highly matched signals between the highlighted key sentences. Other parts of the abstracts are not important for matching. This observation inspires us that it is unnecessary to force a matching model to aggregate all of the sentence pairs. More importantly, the introduced noise may overwhelm the matching signals between key sentences. Thus, identifying key sentences is becoming an essential step for document matching.

In this paper, we propose a novel Optimal Partial Transport (OPT) (Figalli, 2010) based sentence selection component for existing long-form document matching model, namely OPT-Match. As illustrated in Figure 2, OPT-Match models the document alignment as an OPT process by regarding two input documents (two sets of sentences) as two piles of earth. To achieve partial alignment, OPT-Match poses a limitation on how much the earth from one pile (*Document 1*) needs to be transported to the other pile (*Document 2* ). Based on the cost matrix whose elements are defined as inverse (or negative) similarities between sentences from different documents, OPT-Match generates an optimal transport plan (matrix) (Benamou et al., 2015) which indicates the alignment of sentences. Therefore, key sentences can be extracted based on the transport plan. To make OPT-Match easily incorporated into existing document matching models, we provide two strategies to achieve the sentence selection.

Compared to existing OT-based methods (Kusner et al., 2015; Chen et al., 2018, 2019; Zhang et al., 2020), OPT-Match offers the following advantages in modeling the long-form document alignment: 1) OPT-Match models the partial nature of document alignment explicitly and flexibly, through limiting how much the mass to be transported; 2) OPT-Match allows the source and the target domains are not necessarily with the same total mass, which fits well with the phenomenon that the lengths of two documents may vary greatly. The OT-based methods, however, cannot take this into consideration; 3) OPT-Match is a model-agnostic approach, it can be easily plugged into a wide range of document matching models.

To summarize, this paper makes the following main contributions: (1) We highlight the critical importance of the key sentence selection in long-form document matching, which has not yet been thoroughly studied in existing models. (2) We propose a wide applicable component called OPT-Match, which selects key sentences for document matching by conducting partial document alignments. (3) We conducted extensive empirical studies on four large-scale publicly available datasets. The experimental results demonstrated that OPT-Match improved existing document matching models and the sentences selected by OPT-Match were consistent with human rationales.

## 2 Proposed OPT-Match Method

### 2.1 Problem Statement

Suppose that we are given a set of labeled data tuples: $\mathcal{D} = \{(X_i, Y_i, z_i)\}$, where the elements in the $i$-th training instance $X_i \in \mathcal{X}$, $Y_i \in \mathcal{Y}$, and $z_i \in \mathcal{Z}$ respectively denote the source document, the target document, and the label which indicates the semantic relationship of $X_i$ and $Y_i$. Both the source and the target documents consist of a number of sentences, i.e., $X_i = [s_1^{X_i}, s_2^{X_i}, \ldots, s_M^{X_i}]$, $Y_i = [s_1^{Y_i}, s_2^{Y_i}, \ldots, s_N^{Y_i}]$, where the $M$ and $N$ denote the number of sentences in $X_i$ and $Y_i$ respectively. The learning objective of existing document matching models is $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$, which takes all sentences in the input documents as input and outputs a prediction of the relationship between them. As the key idea of this paper, we aim at learning $f'$ which selects the key sentences $S^X \subseteq X, S^Y \subseteq Y$ from input documents and conducts matching based on those selected sentences rather than all sentences.

## 2.2 The Principle of Our Method

To learn $f'$, we develop a novel sentence selection method from the viewpoint of optimal partial transport (OPT), which is an extension of optimal transport (OT). Originally, OT defines a distance between probability distributions. Given two probability distributions $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ which can be viewed as two piles of earth with equivalent mass, the optimal transport distance is defined as the minimum cost of turning one pile into the other, and the corresponding optimal transport plan provides a soft matching between two piles in a probabilistic way. Existing OT-based NLP studies (Kusner et al., 2015; Chen et al., 2018, 2019; Zhang et al., 2020) set the distributions $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ uniform, i.e., $\boldsymbol{\mu} = \frac{1}{M}\mathbf{1}_M$ and $\boldsymbol{\nu} = \frac{1}{N}\mathbf{1}_N$, where $\mathbf{1}_D$ represents the $D$-dimensional all-one vector, and accordingly, the optimal transport distance between them is:

$$\begin{aligned}\mathbf{T}^* &= \arg\min_{\mathbf{T}\in\Pi(\boldsymbol{\mu},\boldsymbol{\nu})} \mathbb{E}_{m,n\sim\mathbf{T}}[c(s_m^X, s_n^Y)] \\ &= \arg\min_{\mathbf{T}\in\Pi(\boldsymbol{\mu},\boldsymbol{\nu})} \sum_{m=1}^{M}\sum_{n=1}^{N} T_{mn}\cdot c(s_m^X, s_n^Y),\end{aligned} \tag{1}$$

where $\mathbf{T} \in \Pi(\boldsymbol{\mu},\boldsymbol{\nu}) = \{\mathbf{T} \in \mathbb{R}_+^{M\times N} | \mathbf{T}\mathbf{1}_N = \boldsymbol{\mu}, \mathbf{T}^\top\mathbf{1}_M = \boldsymbol{\nu}\}$ represents an arbitrary joint distribution of the sentences with marginals $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$. $\mathbf{C} = [c(s_m^X, s_n^Y)] \in \mathbb{R}^{M\times N}$ is a sentence-level cost matrix, whose element $c(s_m^X, s_n^Y)$ measures the discrepancy between the two sentences. As shown in Eq. (1), OT corresponds to the minimum expectation of the sentence-level discrepancy, and thus shares a similar spirit with existing methods, which aggregate all sentence-level matching signals.

However, OT suffers from the following issues in modeling long-form document matching: 1) OT requires that $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ have identical total mass. This setting is unsuitable for document matching because the number of sentences in documents may vary greatly and the lengthy document contains more semantics in general. 2) OT requires the source points must exactly map to the targets. However, in document matching, only some key sentences from the source document align to that from the target document, and thus there should be only a fraction of mass from the source should be transported to the target. 3) OT aggregates all sentence-level aligning signals which inevitably involves noises to the matching. In this work, we solve these issues by modeling the sentence-level alignment as an OPT process (issue 1,2) and incorporating it into existing document matching methods as a sentence selection module (issue 3).

## 2.3 OPT-based Sentence-level Alignment

To fix the issue 1 and 2, we need to break the constraint that $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ must have the same total mass and limit the transporting mass, which leads to an OPT problem:

$$\mathbf{T}^* = \arg\min_{\mathbf{T}\in\Pi_\leq(\boldsymbol{\mu},\boldsymbol{\nu}),\mathbf{1}_M^\top\mathbf{T}\mathbf{1}_N=\epsilon} \langle \mathbf{T}, \mathbf{C}\rangle. \tag{2}$$

where $\mathbf{T} \in \Pi_\leq(\boldsymbol{\mu},\boldsymbol{\nu}) = \{\mathbf{T}\mathbf{1}_N \leq \boldsymbol{\mu}, \mathbf{T}^\top\mathbf{1}_M \leq \boldsymbol{\nu}\}$, $<,>$ represents the Frobenius dot-product. For the issue 1, considering that in OPT, $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ are not necessarily with the same total mass (Benamou et al., 2015) and usually the longer documents contains more semantics, we set a unit mass on each sentence, i.e., $\boldsymbol{\mu} = \mathbf{1}_M, \boldsymbol{\nu} = \mathbf{1}_N$. For the issue 2, as shown in Eq. (2), we set $\epsilon$, indicating a proportion of total mass $\min(||\boldsymbol{\mu}||_1, ||\boldsymbol{\nu}||_1)$ to be transported, to control the degree of the document alignment. Intuitively, with the lower $\epsilon$, OPT-Match focuses more on strongly aligned sentence pairs, while filtering out more spurious alignment signals.

To measure the discrepancy between two cross-document sentences, we define the cost matrix $\mathbf{C}$ in Eq. (2) based on the similarity between sentence pairs derived from $(X, Y)$:

$$c(s_m^X, s_n^Y) = -\text{sim}(s_m^X, s_n^Y), \tag{3}$$

$\text{sim}(s_m^X, s_n^Y)$ is the similarity between $s_m^X$ and $s_n^Y$. Intuitively, we expect more similar pairs of sentences to be transported at a lower cost, and thus can be more strongly aligned. In Eq. (3), various methods can be adopted to measure $\text{sim}(s_m^X, s_n^Y)$, leading to different kinds of cost, for example, the cosine similarity between sentence embeddings, and the overlapping words ratio after removing stop-words of $(s_m^X, s_n^Y)$ (Mihalcea and Tarau, 2004). We respectively apply these two methods to soft selection strategy and hard selection strategy which we will introduce in Sec. 2.4 and Sec. 2.5.

To solve the OPT problem in Eq. (2), a number of algorithms have been proposed. As a representative method, Benamou et al. (2015) propose to add an entropic regularizer (Cuturi, 2013; Xie et al., 2018) and solve it with iterative Bregman projections (Bregman, 1967) and Dykstra algorithm (Dykstra, 1983). See also (Chizat et al., 2018; Zhou et al., 2020). For the fast approximation of OPT, an entropic regularizer $E(\mathbf{T})$ (Cuturi, 2013) is added and the the optimal transport plan is

$$\mathbf{T}^* = \mathop{\arg\min}_{\mathbf{T}\in\Pi_{\leq}(\boldsymbol{\mu},\boldsymbol{\nu}),\mathbf{1}_M^\top\mathbf{T}\mathbf{1}_N=\epsilon} \langle\mathbf{T}\cdot\mathbf{C}\rangle+\lambda E(\mathbf{T}), \quad (4)$$

where $\lambda$ is the trade-off coefficient. Thus, the optimal partial transport plan $\mathbf{T}^*$ can be iteratively calculated by Bregman-Dykstra iterations (Bregman, 1967; Dykstra, 1983; Benamou et al., 2015):

$$\begin{aligned}
\mathbf{T}_n^1 &= \mathrm{diag}\left(\min\left(\frac{\boldsymbol{\mu}}{\mathbf{T}_{n-1}\mathbf{1}},\mathbf{1}\right)\right)\mathbf{T}_{n-1},\\
\mathbf{T}_n^2 &= \mathbf{T}_n^1\cdot\mathrm{diag}\left(\min\left(\frac{\boldsymbol{\nu}}{\mathbf{T}_n^{1\top}\mathbf{1}},\mathbf{1}\right)\right),\\
\mathbf{T}_n &= \mathbf{T}_n^2\cdot\frac{\epsilon}{\mathbf{1}^\top\mathbf{T}_n^2\mathbf{1}},
\end{aligned} \quad (5)$$

where $\mathbf{T}_0 = \exp(-\mathbf{C}/\lambda)$. $\mathbf{T}^*$ indicates the amount of probability mass moved from one pile of earth to the other under the constraint that limited mass should be transported. In the sentence alignment scenario, $\mathbf{T}^*$ can be regarded as the degree of the alignment between the source sentences and the target sentences in which only those strongly aligned sentences are be highlighted.

### 2.4 Sentence Selection

To fix the issue 3, we need to select sentences $S^X, S^Y$ for matching according to $\mathbf{T}^*$. We provide two strategies to achieve that.

**Hard Selection.** We take an aggressive approach to filter out the noise in the document, that is, we select $k$ sentences from the source and the target document respectively with the highest alignment in the optimal transport plan and discarding the rest of the sentences, where $k$ is a hyper-parameter which stands for the desirable number of key sentences. Specifically, $\mathbf{T}^*$ is summed by rows, and the sentences in the source document corresponding to the top-$k$ indexes of $\mathbf{T}^*\mathbf{1}_n$ are selected, then placed to $S^X$. Similarly, for the target document, $S^Y$ is constructed based on the top-$k$ indexes of $\mathbf{1}_n^\top\mathbf{T}^*$. Although this strategy is non-differentiable, it effectively filters out noise.

**Soft Selection.** To make the selection differentiable, we provide an alternative. Given $\mathbf{T}^*$ as the sampling probabilities, the key sentences are sampled using the Gumbel softmax (Jang et al., 2016), which provides a differentiable sampling process:

$$\begin{aligned}
u_i &\sim U(0,1), g_i = -\log(-\log(u_i)),\\
w_i &= \frac{\exp\left((\log(prob_i)+g_i)/\tau\right)}{\sum_j \exp\left((\log(prob_i)+g_i)/\tau\right)},
\end{aligned} \quad (6)$$

where $U(0,1)$ represents the uniform distribution between $0$ and $1$, and $\tau$ is a temperature hyperparameter, $prob_i$ represents the probability of choosing each sentence as the selected sentence. For the source document, $prob_i$ is normalized $\mathbf{T}^*\mathbf{1}_N$, for the target document, $prob_i$ is normalized $\mathbf{1}_M^\top\mathbf{T}^*$. Therefore, we obtain the selection weight $w_i$ for each sentence, and the key sentence sets are $S^X = (w_1^X s_1^X, \cdots, w_M^X s_M^X)$ and $S^Y = (w_1^Y s_1^Y, \cdots, w_N^Y s_N^Y)$.

### 2.5 Combination with existing models

Till now, OPT-Match has extracted $S^X$ and $S^Y$, a paired subset of sentences from input documents. As a widely applicable module, OPT-match can be easily combined with various document matching models. In this work, we take two representative methods as the underlying models for OPT-Match.

For models which hierarchically encode document (Jiang et al., 2019; Pappagari et al., 2019; Zhou et al., 2020), they suppose that documents present a hierarchical structure including words, sentences, and paragraphs. OPT-Match can be plugged at the sentence level, because hierarchical models explicitly represent all sentences in a document. For example, once the sentence representations are obtained, one can construct the cost matrix $\mathbf{C}$ based on the cosine similarity between sentence embeddings (Eq.3), then use the soft selection strategy. OPT-Match also can be used before sentence encoding by adopting the overlapping words ratio cost and the hard selection strategy.

For BERT and its variants (Devlin et al., 2019; Dai et al., 2019; Beltagy et al., 2020), since they focus on token-level interaction and do not explicitly generate sentence representations, OPT-Match can be regarded as a pre-processing to combining with BERT and its variants. One can use the overlapping words ratio as the OPT cost and select sentences using the hard selection strategy.

### 2.6 Training

As aforementioned, OPT-Match is a sentence selection module before matching, and thus the learning objective of OPT-Match equipped models is identical to its underlying models. In the underlying models, the cross-entropy loss which measures the discrepancy between the model's predictions and ground-truth labels is widely used for training:

$$\mathcal{L} = \sum_i \ell(M(S^{X_i}, S^{Y_i}), z_i) = \sum_i z_i \mathrm{log} p_i + (1-z_i)\mathrm{log}(1-p_i) \quad (7)$$

**Algorithm 1** Training process of OPT-Mach based models.

```
Require: Training set $\mathcal{D} = \{(X_i, Y_i, z_i)\}_{i=1}^{N}$; mini-batch sizes $n_b$; learning rates $\eta$; Bregman-Dykstra iterations step $iter$; entropic regularizer coefficient $\lambda$; mass to be transported $\epsilon$, number of selected sentences $k$, Gumbel temperature $\tau$.
repeat
    ▷ OPT-Match component
    Sample mini-batch $\{(X_i, Y_i, z_i)\}_{i=1}^{n_b} \subseteq \mathcal{D}$
    Calculate the cost matrix $\mathbf{C}$ {Eq. (3)}
    $\mathbf{T} = \exp\left(-\mathbf{C}/\lambda\right)$
    for $t = 1$ to $iter$ do
        $\mathbf{T} \leftarrow \text{diag}\left(\min\left(\frac{\boldsymbol{\mu}}{\mathbf{T}\mathbf{1}}, \mathbf{1}\right)\right)\mathbf{T}$
        $\mathbf{T} \leftarrow \mathbf{T} \cdot \text{diag}\left(\min\left(\frac{\boldsymbol{\nu}}{\mathbf{T}^{\top}\mathbf{1}}, \mathbf{1}\right)\right)$
        $\mathbf{T} \leftarrow \mathbf{T} \cdot \frac{\epsilon}{\mathbf{1}^{\top}\mathbf{T}\mathbf{1}}$ {Eq. (5)}
    end for
    if Hard Selection then
        $S^X \leftarrow$ top $k$ indexes (sentences) of $\mathbf{T}\mathbf{1}_N$
        $S^Y \leftarrow$ top $k$ indexes (sentences) of $\mathbf{1}_M^{\top}\mathbf{T}$
    else if Soft Selection then
        $S^X \leftarrow (w_1^X s_1^X, \cdots, w_n^X s_n^X)$ {Eq. (6)}
        $S^Y \leftarrow (w_1^Y s_1^Y, \cdots, w_m^Y s_m^Y)$ {Eq. (6)}
    end if
    ▷ Matching component $M$
    $\mathcal{L} = \sum_{i=1}^{n_b} \ell(M(S^X, S^Y; \mathbf{\Theta}), z_i)$ {Eq. (7)}
    $\mathbf{\Theta} \leftarrow \mathbf{\Theta} - \eta \nabla_{\mathbf{\Theta}} \mathcal{L}$
until convergence
return $\mathbf{\Theta}$
```

where $z_i$ is the ground-truth label, $p_i = M(S^{X_i}, S^{Y_i})$ is the final matching prediction, and $M$ represents the matching component. We include the training procedure of OPT-Match equipped matching models in Alg. 1.

## 3 Experiments

### 3.1 Experimental Setup

**Datasets.** The experiments are conducted on four large-scale publicly available long-form document matching datasets [1]. Table 1 provides the dataset statistics. Data splits and preprocessing for all datasets follow (Zhou et al., 2020).

**Citation recommendations** is a task to predict whether a paper cites another. In the experiments, AAN (Radev et al., 2013), OC (Bhagavatula et al., 2018), and S2ORC (Lo et al., 2020) are exploited for this task. Note that following the practice of (Zhou et al., 2020), we only use the paper abstract of AAN. S2ORC contains human annotations to indicate which sentences in the source document cite the target document. These annotations can be used to assess the quality of selected sentences by OPT-Match.

Table 1: Statistics of datasets. '**#Word**' denotes the average number of words per document, and '**#Sent.**' denotes the average number of sentences per document.

| Dataset | Train | Dev | Test | #Word | #Sent. |
|---|---|---|---|---|---|
| PAN | 17,968 | 2,908 | 2,906 | 1,569.7 | 47.4 |
| S2ORC | 152,000 | 19,000 | 19,000 | 263.7 | 9.3 |
| AAN | 106,592 | 13,324 | 13,324 | 122.7 | 4.9 |
| OC | 240,000 | 30,000 | 30,000 | 190.4 | 7.0 |

**Plagiarism detection** is a task to detect whether a text span in the source document plagiarizes a text span in the target document. PAN (Potthast et al., 2013) is used for this task. PAN also contains human annotations to indicate which sentences in the source document plagiarizes the target document.

**Baselines.** For document matching, we take two representative kinds of methods including the hierarchical models and the variants of BERT as the underlying model and compare the performance of these models with and without OPT-Match.

The hierarchical models include GRU-HAN (Jiang et al., 2019) which uses stacked GRU and attention networks to encode documents following the order of words, sentences, paragraphs, and documents; BERT-HAN (Pappagari et al., 2019) which replaces sentence encoder of GRU-HAN with BERT; GRU-HAN-CDA and BERT-HAN-CDA (Zhou et al., 2020) which add cross-document attention to GRU-HAN and BERT-HAN respectively. Following (Zhou et al., 2020), the attention scores of sentences are used to indicate the model's selection of key sentences.

Variants of BERT includes BERT (Devlin et al., 2019), Transformer-XL (Dai et al., 2019), and Longformer (Beltagy et al., 2020). For BERT, we choose the *'bert-base-uncased'* and truncate the first 510 words of the document. For Transformer-XL and Longformer, we choose *"transfoxl-wt103'* and *'allenai/longformer-base-4096'* respectively.

To further verify the impact of partial alignment between documents, we consider two OT-based method: (Kusner et al., 2015) which uses OT as a similarity function between sentences and (Swanson et al., 2020) which conducts sparse OT between sentences by adding dummy nodes[2].

[1] https://github.com/XuhuiZhou/CDA

[2] We consider the relaxed 1-to-k assignment which is suit-

Table 2: Experimental results on S2ORC, PAN, AAN and OC test sets. $x$-OPT denotes OPT-Match equipped model $x$. $(soft)$ and $(hard)$ indicate the soft selection strategy and the hard selection strategy respectively. $^\dagger$ indicates the statistically significant difference between the model equipped with OPT-Match and the corresponding underlying model ($p$-value $< 0.05$).

| | AAN | | OC | | S2ORC | | PAN | |
|---|---|---|---|---|---|---|---|---|
| **Models** | **Acc.** | **F1** | **Acc.** | **F1** | **Acc.** | **F1** | **Acc.** | **F1** |
| GRU-HAN | 68.01 | 67.23 | 84.46 | 82.26 | 82.36 | 83.28 | 75.70 | 75.88 |
| GRU-HAN-OPT $(soft)$ | 69.87$^\dagger$ | 69.30$^\dagger$ | **85.76**$^\dagger$ | **85.89**$^\dagger$ | 83.82$^\dagger$ | 84.29$^\dagger$ | 76.28$^\dagger$ | 76.63$^\dagger$ |
| GRU-HAN-OPT $(hard)$ | **71.02**$^\dagger$ | **70.91**$^\dagger$ | 85.49$^\dagger$ | 85.74$^\dagger$ | **85.14**$^\dagger$ | **85.65**$^\dagger$ | **76.76**$^\dagger$ | **77.03**$^\dagger$ |
| GRU-HAN-CDA | 74.51 | 74.81 | 88.71 | 88.96 | 88.91 | 89.92 | 77.04 | 78.23 |
| GRU-HAN-CDA-OPT $(soft)$ | 75.88$^\dagger$ | 76.06$^\dagger$ | **89.94**$^\dagger$ | **90.11**$^\dagger$ | 89.15 | 89.84 | 78.10$^\dagger$ | 78.39 |
| GRU-HAN-CDA-OPT $(hard)$ | **76.96**$^\dagger$ | **76.65**$^\dagger$ | 88.62 | 88.78 | **89.72**$^\dagger$ | **89.96**$^\dagger$ | **78.52**$^\dagger$ | **78.84**$^\dagger$ |
| BERT-HAN | 67.32 | 64.97 | 85.96 | 86.33 | 90.67 | 90.76 | 87.57 | 87.36 |
| BERT-HAN-OPT $(soft)$ | 68.72$^\dagger$ | 68.98$^\dagger$ | 87.30$^\dagger$ | 87.44$^\dagger$ | 90.75 | 90.87 | 87.74 | 87.25 |
| BERT-HAN-OPT $(hard)$ | **70.57**$^\dagger$ | **71.22**$^\dagger$ | **88.21**$^\dagger$ | **88.49**$^\dagger$ | **91.40**$^\dagger$ | **91.55**$^\dagger$ | **88.12**$^\dagger$ | **88.01**$^\dagger$ |
| BERT-HAN-CDA | 71.57 | 69.08 | 87.81 | 87.89 | 91.92 | 92.07 | 86.23 | 86.19 |
| BERT-HAN-CDA-OPT $(soft)$ | 73.85$^\dagger$ | 73.42$^\dagger$ | 89.07$^\dagger$ | 89.01$^\dagger$ | 92.52$^\dagger$ | 92.61$^\dagger$ | 87.13$^\dagger$ | 86.89$^\dagger$ |
| BERT-HAN-CDA-OPT $(hard)$ | **75.56**$^\dagger$ | **75.62**$^\dagger$ | **90.58**$^\dagger$ | **90.54**$^\dagger$ | **92.74**$^\dagger$ | **92.81**$^\dagger$ | **87.61**$^\dagger$ | **87.14**$^\dagger$ |
| BERT | 88.05 | 88.09 | 94.52 | 94.45 | 95.64 | 95.64 | 59.58 | 69.71 |
| BERT-OPT $(hard)$ | **89.31**$^\dagger$ | **89.35**$^\dagger$ | **95.06**$^\dagger$ | **94.97**$^\dagger$ | **96.85**$^\dagger$ | **96.82**$^\dagger$ | **89.09**$^\dagger$ | **88.61**$^\dagger$ |
| Transformer-XL | 83.18 | 82.92 | 91.19 | 91.26 | 92.50 | 92.39 | 58.25 | 69.07 |
| Transformer-XL-OPT $(hard)$ | **85.03**$^\dagger$ | **84.99**$^\dagger$ | **92.37**$^\dagger$ | **92.43**$^\dagger$ | **93.80**$^\dagger$ | **93.69**$^\dagger$ | **80.28**$^\dagger$ | **80.11**$^\dagger$ |
| Longformer | 88.01 | 88.29 | 93.46 | 93.51 | 96.02 | 96.07 | 56.82 | 69.78 |
| Longformer-OPT $(hard)$ | **88.92**$^\dagger$ | **89.07**$^\dagger$ | **94.88**$^\dagger$ | **94.87**$^\dagger$ | **96.61**$^\dagger$ | **96.56**$^\dagger$ | **82.68**$^\dagger$ | **82.21**$^\dagger$ |

**Evaluation Metrics.** We use Accuracy and F1 as the evaluation metrics since all the datasets have binary labels for document matching. Following (Zhou et al., 2020), we use MRR as the evaluation metric for sentence selection since the sentence selection is regarded as a ranking task.

**Implementation Details.** All hyper-parameters in OPT-Match[3] are tuned using grid search on the validation set. The tuning range of hyperparameters are as follows: the proportion of mass to be transported $\epsilon$ in Eq. (2) is tuned among $\{0.25, 0.50, 0.75\}$; coefficient $\lambda$ in Eq. (4) is tuned between $[0.5, 1.0]$; Gumbel temperature $\tau$ in Eq. (6) is tuned between $[0.5, 1.0]$. For OPT-Match equipped hierarchical models, the settings of optimizer, learning rate, batch size and hidden dimension are consistent with corresponding underlying models. For OPT-Match equipped BERT's variants, the fine-tuning optimizer is Adam (Kingma and Ba, 2014) with $\beta_1 = 0.9$, $\beta_2 = 0.999$, the learning rate is tuned between $[0.00001, 0.00005]$; the batch size is tuned between $[4, 8]$.

### 3.2 Matching performances

Table 2 reports the evaluation results of different models. All the methods are trained ten times and the averaged results are reported. Since BERT and its variants do not explicitly generate sentence representations, we only applied the word-overlap ratio cost and the hard selection versions of OPT-Match on these models. For a fair comparison, we follow (Swanson et al., 2020) and keep the number of selected sentences as 5 in the hard selection.

We summarize our observations from the results as follows: 1) in general, the models equipped with OPT-Match consistently outperform their corresponding underlying models across four datasets. The results verify the effectiveness of OPT-Match in terms of enhancing matching accuracy. 2) BERT and its variants such as Transformer-XL and Longformer achieve much worse performances on PAN, comparing to their good performances on other datasets. PAN has a large average document length ($>$ 1500 words) and a large average number of sentences per document ($>$ 40 sentences). The noise in the extremely long documents makes BERT and its variations perform poorly. OPT-Match equipped models, however, are not affected by the document length. It is because OPT-Match successfully filters out the noise in the document by selecting key sentences for matching. The results indicate that OPT-Match is more effective especially when the document length is extremely long; 3) Comparing the performances of OPT-Match in the soft selection version to that of in the hard selection version, the hard selection versions achieve better performances in most cases. We analyze the results, and the reason is that the soft selection is a weighting strategy, aiming at sampling sentences according to the optimal transport plan of OPT-Match. However, the soft selection cannot completely filter out the noise in documents, which often has negative impacts on the matching.

able for long-form document matching.

[3] The source code of OPT-Match is available at `https://github.com/ruc-wjyu/OPT-Match`

Table 3: The impact of degree of the document alignment $\epsilon$ in OPT-Match. BHC-OPT denotes BERT-HAN-CDA-OPT model in the soft selection and BERT-OPT is in the hard selection version.

| | S2ORC | | PAN | |
|---|---|---|---|---|
| **Models** | **Acc.** | **F1** | **Acc.** | **F1** |
| BHC-OT(Kusner et al., 2015) | 91.46 | 91.50 | 85.90 | 85.98 |
| BHC-SOT (Swanson et al., 2020) | 91.91 | 91.96 | 86.55 | 86.32 |
| BHC-OPT($\epsilon = 0.75$) | 92.33 | 92.46 | 86.95 | 86.59 |
| BHC-OPT ($\epsilon = 0.50$) | 92.65 | 92.74 | 87.08 | 86.75 |
| BHC-OPT ($\epsilon = 0.25$) | **92.74** | **92.81** | **87.61** | **87.14** |
| BERT-OT (Kusner et al., 2015) | 95.27 | 95.31 | 66.28 | 65.47 |
| BERT-SOT (Swanson et al., 2020) | 96.05 | 96.13 | 87.86 | 87.95 |
| BERT-OPT ($\epsilon = 0.75$) | 96.79 | 96.75 | 88.02 | 87.44 |
| BERT-OPT ($\epsilon = 0.50$) | 96.77 | 96.76 | 88.67 | 88.24 |
| BERT-OPT ($\epsilon = 0.25$) | **96.85** | **96.82** | **89.09** | **88.61** |

### 3.3 Impact of Partial Alignment

To test the effects of the partial alignment between documents, we compare the performances of OPT-Match with different proportion of mass $\epsilon$ (degree of alignment) to two representative OT-based (full alignment) methods (Kusner et al., 2015; Swanson et al., 2020). We suppose that with the lower $\epsilon$, OPT-Match focuses more on strongly aligned sentence pairs, while filtering out more spurious alignment signals. Please note that (Kusner et al., 2015) conduct traditional OT between documents which denotes the full alignment and (Swanson et al., 2020) add dummy sentences and conducts full alignment between real sentences and dummy ones in order to achieve partial alignment within real sentences. As illustrated in Table 3, we find that models with partial alignment ($\epsilon < 1$) always achieve better performances than that of the full alignment on S2ORC and PAN datasets. The results verify that the alignment between documents is partial. Moreover, we find that OPT-Match tends to achieve better performance with smaller $\epsilon$. The results indicate that only a small fraction of strongly aligned sentences contributed to document matching, filtering out the noise sentences is helpful. In addition, although (Swanson et al., 2020) aims at partial alignment, the way of adding dummy nodes may not suit long-form documents matching because the document length varies greatly leading to many dummy nodes, and the alignment between dummy nodes and real sentences may dominate in the full alignment and overwhelm the real alignments between sentences.

### 3.4 Impact of the number of selected sentences.

We further conduct experiments to investigate how the number of selected sentences in OPT-Match affects the matching. Specifically, we configure BERT-OPT ($hard$) to select a different number of sentences (from 1 to the average number of sentences in a document) and then conduct matching using these sentences. Figure 3 illustrates matching accuracy curves w.r.t. the number of selected sentences on the datasets of S2ORC and OC. We find that BERT-OPT shows a competitive performance when only 1 or 2 sentences are selected. The results show the effectiveness of OPT-Match in terms of accurately selecting sentences key to document matching. Additionally, the accuracy curves first steadily increase and reach the peak when 4 or 5 sentences are selected for matching. After that, the curves drop with more selected sentences. The phenomenon is intuitive and can be explained as follows: when the number of selected sentences is too small, the model needs more information from the selected sentences to infer the document-level matching. However, the number of selected sentences is greatly less than the number of all of the sentences in the document. Therefore, after some thresholds, the additional selected sentences become noisy, which causes the drop of the matching accuracy. The results verify our assumption that not all the sentences in the document contribute to the long-form document matching.

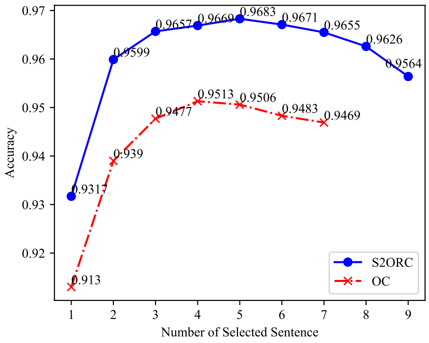

Figure 3: The impact of the number of selected sentence on OPT-Match. Results from BERT-OPT of the hard selection version on S2ORC and OC.

Table 4: Faithfulness evaluation. $(all)$, $(selected)$, and $(all \setminus selected)$ respectively denotes using all sentences, using sentences selected by OPT-Match, and using the sentences not selected by OPT-Match as the inputs.

| | S2ORC | | PAN | |
|---|---|---|---|---|
| **Models** | **Acc.** | **F1** | **Acc.** | **F1** |
| BERT-HAN $(all \setminus selected)$ | 73.62 | 74.33 | 56.76 | 53.52 |
| BERT-HAN $(all)$ | 90.67 | 90.76 | 77.04 | 78.23 |
| BERT-HAN $(selected)$ | **91.40** | **91.55** | **88.12** | **88.01** |
| BERT-HAN-CDA $(all \setminus selected)$ | 77.30 | 77.94 | 57.28 | 53.73 |
| BERT-HAN-CDA $(all)$ | 91.92 | 92.07 | 86.23 | 86.19 |
| BERT-HAN-CDA $(selected)$ | **92.74** | **92.81** | **87.61** | **87.14** |

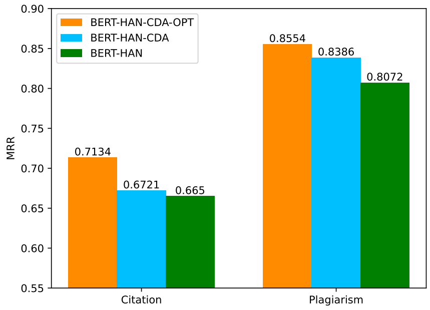

Figure 4: Plausibility comparison among BERT-HAN, BERT-HAN-CDA and BERT-HAN-CDA-OPT $(hard)$ on S2ORC (Citation) and PAN (Plagiarism).

### 3.5 Selected Sentences as Rationales

To further assess the quality of the key sentence selected by OPT-Match, we regard the selected sentences as the rationales for the document-level matching prediction. Following (Strout et al., 2019; DeYoung et al., 2020), we adopt plausibility and faithfulness as the evaluation metrics. Plausibility measures how well the rationales provided by models align with human annotations, and faithfulness measures the degree to which the rationales influence the corresponding predictions.

Following the setting of (Zhou et al., 2020), we first compare the sentences selected by OPT-Match $(hard)$ with human annotations in S2ORC and PAN. As shown in Figure 4, the sentences selected by OPT-Match are more consistent with human annotations compared with that of the sentence-level attention scores in BERT-HAN and BERT-CDA.

In terms of the faithfulness, we test the matching performance of BERT-HAN and BERT-HAN-CDA under three conditions: respectively using all sentences (denoted as $(all)$), using sentences selected by OPT-Match $(hard)$ (denoted as $(selected)$), and using sentences except those selected by OPT-Match $(hard)$ (denoted as $all \backslash selected$) as the model's input. From the results reported in Table 4, we find that the sentences selected by OPT-Match play a critical role in document matching, i.e., if the sentences selected by OPT-Match are removed from a model's input, the matching accuracy of the model drops dramatically. In addition, if all sentences are used as a model's input (note that the input still contains the sentences selected by OPT-Match), the predictive accuracy still drops to some extent because of the noise from other sentences. From the results, we conclude that OPT-Match is capable of accurately selecting the key sentences for document matching and filtering the noise.

Table 5: Training time (s/batch) on a single Nvidia Tesla V100 16GB. Batch size = 256 for both models.

| **Models** | S2ORC | PAN |
|---|---|---|
| BERT-HAN-CDA | 0.0992 | 0.2711 |
| BERT-HAN-CDA-OPT $(soft)$ | 0.1004 | 0.2917 |
| BERT-HAN-CDA-OPT $(hard)$ | 0.0934 | 0.1490 |

### 3.6 Time Complexity Analysis

Existing long-form document matching methods usually apply attention mechanism at the word level, which have a time complexity of $\mathcal{O}(N_{all}^2)$, where $N_{all}$ denotes the number of tokens in the input document. For OPT-Match, the computing overhead mainly comes from the calculation of the optimal transport plan (line 6-9 in Algorithm 1). With the help of the entropic regularizer (Cuturi, 2013) and Bregman-Dykstra iterations (Bregman, 1967), the calculation of the optimal transport plan have a time complexity of $\mathcal{O}(N_s^2)$ (Benamou et al., 2015), where the $N_s$ denotes the number of the sentences in the input document. Since usually the number of sentences is far less than the number of tokens in a document, i.e., $N_s \ll N_{all}$, we suppose the computational cost of OPT-Match is acceptable. In addition, if the word overlapping cost is applied, the hard selection version of OPT-Match can be used as a data pre-processing, therefore, OPT-Match does not increase the training time of the model. Also note that the input of the matching model equipped with OPT-Match is the selected sentences rather than all of the sentences. Therefore, the hard selection version of OPT-Match with word overlapping cost can effectively reduce the training time of the matching model. For the soft selection version of OPT-Match, although OPT-Match brings additional computational cost, considering OPT-Match is applied at the sentence level, the additional computation cost is not significant.

We further compare the average training time between BERT-HAN-CDA and BERT-HAN-CDA-OPT. From the results shown in Table 5, we can see that the soft selection version of OPT-Match brings an acceptable additional computation cost, while the hard selection version of OPT-Match effectively reduce the training time of the matching model.

## 4 Related Work

### 4.1 Long-form Document Matching

In long-form document matching, there are two representative methods in the literature — the hierarchical models and the variants of BERT.

The hierarchical models suppose that a document represents a hierarchical structure of words, sentences, paragraphs, and documents. Inspired by this idea, researchers exploit neural network to encode document in a hierarchical way. The representative method is (Jiang et al., 2019). This method first separately models input document pair as semantic vectors using RNN and Hierarchical Attention Network (Yang et al., 2016; Zhou et al., 2020). Then, the matching score is calculated by feeding the concatenation of the document vector to an MLP. Pappagari et al. (2019) improves (Jiang et al., 2019) by replacing the RNN-based encoder with the transformer-based encoder. Zhou et al. (2020) focuses on the interaction between documents and proposed the hierarchical cross-document attention to improve the document representation.

Although BERT has been dominated in the field of short-text matching, the quadratic time complexity of intrinsic attention mechanism makes BERT difficult to be applied in the long-form document matching. To tackle this issue. Dai et al. (2019) proposed Transformer-XL which consists of a segment-level recurrence mechanism and a positional encoding scheme. Transformer-XL enables learning dependency beyond a fixed length without disrupting temporal coherence. Beltagy et al. (2020) proposed dilated sliding windows attention which gradually increases the receptive field as the model goes deeper.

Recently, Pang et al. (2021) proposed to first filter out sentence-level noise from documents by applying PageRank on the sentence similarity graph and then plug PageRank into transformer layers to filter out word-level noise. Although these studies achieved improvement in document matching performances, they ignore the alignment between sentences in a document pair would be partial which inevitably introduced noises to the final matching.

### 4.2 OT in NLP

In recent years, OT have been widely studied in NLP. Kusner et al. (2015) formulated the distance between two sentences as an optimal transport problem and proposed Word Mover's Distance (WMD) which measures the dissimilarity between two text documents as the minimum amount of distance that words of one document need to transport to the words of another document. Yokoi et al. (2020) pointed out that the angle of semantic vectors is a good proxy for word similarity and proposed Word Rotator's Distance on top of WMD. Xu et al. (2018) proposed a Wasserstein method with a distillation mechanism, yielding joint learning of word embeddings and topics. Inspired from Order-Preserving OT (Su and Hua, 2017, 2019; Su et al., 2019), Liu et al. (2018) proposed to factorize sentence hierarchically based on Abstract Meaning Representation and obtain the reordered sentence representations. Then the semantic distance between a pair of text snippets can be solved by a penalized OT. Yu et al. (2020, 2022b) proposed to use OT to bridge the gap between heterogeous text pairs for sentence matching in asymmetrical domains. Chen et al. (2019); Zhang et al. (2020) respectively applied OT and OPT to sequence-to-sequence learning tasks such as text generation. (Li et al., 2019; Yu et al., 2022a) proposed to learn the similarity between texts using inverse optimal transport.

## 5 Conclusion

In this paper, we highlight the critical role of conducting partial alignment in long-form document matching. A novel key sentence selection component based on optimal partial transport is proposed, called OPT-Match. OPT-Match automatically selects key sentences for document matching, addressing the issue that not every sentence in one document can correspond to a sentence in another document. Moreover, OPT-Match can be easily incorporated into existing document matching models. Comprehensive experiments on four public datasets show that OPT-Match consistently outperformed its underlying document matching models. The empirical analysis also verifies that the sentences selected by OPT-Match are not only consistent with human-provided rationales but also contributed to document matching.

## Acknowledgements

This work was funded by the National Key R&D Program of China (2019YFE0198200), the National Natural Science Foundation of China (61872338, 61832017, 62276248, 61906180, 61976206), Beijing Outstanding Young Scientist Program NO. BJJWZYJH012019100020098, Intelligent Social Governance Interdisciplinary Platform, Major Innovation & Planning Interdisciplinary Platform for the "Double-First Class" Initiative, Renmin University of China, and Public Policy and Decision-making Research Lab of Renmin University of China.

## References

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *CoRR*, abs/2004.05150.

Jean-David Benamou, Guillaume Carlier, Marco Cuturi, Luca Nenna, and Gabriel Peyré. 2015. Iterative bregman projections for regularized transportation problems. *SIAM J. Sci. Comput.*, 37(2).

Chandra Bhagavatula, Sergey Feldman, Russell Power, and Waleed Ammar. 2018. Content-based citation recommendation. 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 238–251, New Orleans, Louisiana. Association for Computational Linguistics.

Lev M Bregman. 1967. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR computational mathematics and mathematical physics*, 7(3):200–217.

Liqun Chen, Shuyang Dai, Chenyang Tao, Haichao Zhang, Zhe Gan, Dinghan Shen, Yizhe Zhang, Guoyin Wang, Ruiyi Zhang, and Lawrence Carin. 2018. Adversarial text generation via feature-mover's distance. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 4671–4682.

Liqun Chen, Yizhe Zhang, Ruiyi Zhang, Chenyang Tao, Zhe Gan, Haichao Zhang, Bai Li, Dinghan Shen, Changyou Chen, and Lawrence Carin. 2019. Improving sequence-to-sequence learning via optimal transport. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Lenaic Chizat, Gabriel Peyré, Bernhard Schmitzer, and François-Xavier Vialard. 2018. Scaling algorithms for unbalanced optimal transport problems. *Mathematics of Computation*, 87(314):2563–2609.

Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 2292–2300.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988, Florence, Italy. Association for Computational Linguistics.

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.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C Wallace. 2020. Eraser: A benchmark to evaluate rationalized nlp models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4443–4458.

Richard L Dykstra. 1983. An algorithm for restricted least squares regression. *Journal of the American Statistical Association*, 78(384):837–842.

Alessio Figalli. 2010. The optimal partial transport problem. *Archive for rational mechanics and analysis*, 195(2):533–560.

Tomás Foltýnek, Norman Meuschke, and Bela Gipp. 2020. Academic plagiarism detection: A systematic literature review. *ACM Comput. Surv.*, 52(6):112:1–112:42.

Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.

Jyun-Yu Jiang, Mingyang Zhang, Cheng Li, Michael Bendersky, Nadav Golbandi, and Marc Najork. 2019. Semantic text matching for long-form documents. In *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*, pages 795–806. ACM.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Matt J. Kusner, Yu Sun, Nicholas I. Kolkin, and Kilian Q. Weinberger. 2015. From word embeddings to document distances. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 957–966. JMLR.org.

Ruilin Li, Xiaojing Ye, Haomin Zhou, and Hongyuan Zha. 2019. Learning to match via inverse optimal transport. *Journal of machine learning research*, 20.

Bang Liu, Ting Zhang, Fred X. Han, Di Niu, Kunfeng Lai, and Yu Xu. 2018. Matching natural language sentences with hierarchical sentence factorization. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018*, pages 1237–1246. ACM.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. 2020. S2ORC: The semantic scholar open research corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4969–4983, Online. Association for Computational Linguistics.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411.

Liang Pang, Yanyan Lan, and Xueqi Cheng. 2021. Match-ignition: Plugging pagerank into transformer for long-form text matching.

Raghavendra Pappagari, Piotr Zelasko, Jesús Villalba, Yishay Carmiel, and Najim Dehak. 2019. Hierarchical transformers for long document classification. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 838–844. IEEE.

Martin Potthast, Matthias Hagen, Tim Gollub, Martin Tippmann, Johannes Kiesel, Paolo Rosso, Efstathios Stamatatos, and Benno Stein. 2013. Overview of the 5th international competition on plagiarism detection. In *Working Notes for CLEF 2013 Conference , Valencia, Spain, September 23-26, 2013*, volume 1179 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Dragomir R. Radev, Pradeep Muthukrishnan, Vahed Qazvinian, and Amjad Abu-Jbara. 2013. The ACL anthology network corpus. *Lang. Resour. Evaluation*, 47(4):919–944.

Julia Strout, Ye Zhang, and Raymond Mooney. 2019. Do human rationales improve machine explanations? In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 56–62, Florence, Italy. Association for Computational Linguistics.

Bing Su and Gang Hua. 2017. Order-preserving wasserstein distance for sequence matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Bing Su and Gang Hua. 2019. Order-preserving optimal transport for distances between sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(12):2961–2974.

Bing Su, Jiahuan Zhou, and Ying Wu. 2019. Order-preserving wasserstein discriminant analysis. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 9884–9893. IEEE.

Kyle Swanson, Lili Yu, and Tao Lei. 2020. Rationalizing text matching: Learning sparse alignments via optimal transport. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5609–5626, Online. Association for Computational Linguistics.

Yujia Xie, Xiangfeng Wang, Ruijia Wang, and Hongyuan Zha. 2018. A fast proximal point method for wasserstein distance. *CoRR*, abs/1802.04307.

Hongteng Xu, Wenlin Wang, Wei Liu, and Lawrence Carin. 2018. Distilled wasserstein learning for word embedding and topic modeling. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, San Diego, California. Association for Computational Linguistics.

Sho Yokoi, Ryo Takahashi, Reina Akama, Jun Suzuki, and Kentaro Inui. 2020. Word rotator's distance. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2944–2960, Online. Association for Computational Linguistics.

Weijie Yu, Zhongxiang Sun, Jun Xu, Zhenhua Dong, Xu Chen, Hongteng Xu, and Ji-Rong Wen. 2022a. Explainable legal case matching via inverse optimal transport-based rationale extraction. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 657–668.

Weijie Yu, Chen Xu, Jun Xu, Liang Pang, Xiaopeng Gao, Xiaozhao Wang, and Ji-Rong Wen. 2020. Wasserstein distance regularized sequence representation for text matching in asymmetrical domains. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2985–2994, Online. Association for Computational Linguistics.

Weijie Yu, Chen Xu, Jun Xu, Liang Pang, and Ji-Rong Wen. 2022b. Distribution distance regularized sequence representation for text matching in asymmetrical domains. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:721–733.

Ruiyi Zhang, Changyou Chen, Xinyuan Zhang, Ke Bai, and Lawrence Carin. 2020. Semantic matching for sequence-to-sequence learning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 212–222, Online. Association for Computational Linguistics.

Xuhui Zhou, Nikolaos Pappas, and Noah A. Smith. 2020. Multilevel text alignment with cross-document attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5012–5025, Online. Association for Computational Linguistics.