A. Proofs

A.1. Proof of Theorem 1

To prove Theorem 1, we show the theorems presented in (Korte & Vygen, 2006). The number of theorems in the bracket is the number of theorems in (Korte & Vygen, 2006).

Theorem 3 (Theorem 2.5)**.** *Let $\boldsymbol{V} = \{v_1, \cdots, v_N\}$ be a set of nodes and $G = (\boldsymbol{V}, \boldsymbol{E})$ be a directed graph. Then, the following statements are equivalent:*

- *G is a directed tree with the root v_1.*
- *For all $v \in \boldsymbol{V}$, $(v_1, v) \notin \boldsymbol{E}$, and for all $v \in \boldsymbol{V} \setminus \{v_1\}$, a unique $u \in V$ exists so that $(v, u) \in \boldsymbol{E}$, and G contain no circuit.*

Definition 1 (Definition 2.8)**.** Let $\boldsymbol{V} = \{v_1, \ldots, v_N\}$ be a set of nodes and $G = (\boldsymbol{V}, \boldsymbol{E})$ be a directed graph. A *topological order* of G is an order of the nodes so that for each edge $(v_i, v_j) \in \boldsymbol{E}$, we have $i < j$.

Theorem 4 (Proposition 2.9)**.** *A directed graph has a topological order if and only if it is acyclic.*

By replacing all edges $(v_i, v_j) \in \boldsymbol{E}$ with (v_j, v_i), we have the following.

Corollary 4.1. *Let $\boldsymbol{V} = \{v_1, \ldots, v_N\}$ be a set of nodes and $G = (\boldsymbol{V}, \boldsymbol{E})$ be a directed graph. If $i > j$ for all edges $(v_i, v_j) \in \boldsymbol{E}$, then G is acyclic.*

By using these theorems, we prove Theorem 1.

Proof. Because the adjacency matrix $\mathbf{D}_{\text{par}}$ satisfies condition (2) in Theorem 1, for all $v \in \boldsymbol{V}$, we have $(v_1, v) \notin \boldsymbol{E}$, and for all $v \in \boldsymbol{V} \setminus \{v_1\}$, there exists a unique $u \in V$ such that $(v, u) \in \boldsymbol{E}$. Because the adjacency matrix $\mathbf{D}_{\text{par}}$ satisfies conditions (1) in Theorem 1, we have $i > j$ for all edges $(v_i, v_j) \in \boldsymbol{E}$. Due to Corollary 4.1, G is acyclic.

Therefore, G is a directed tree with root v_1 by Theorem 3. □

A.2. Details of Eq. (2)

Because $\mathbf{D}_{\text{par}}$ is a nilpotent matrix, and $\mathbf{D}_{\text{par}}^N$ is a zero matrix,

$$\begin{aligned}(\mathbf{I} - \mathbf{D}_{\text{par}}) \sum_{k=0}^{\infty} \mathbf{D}_{\text{par}}^k &= (\mathbf{I} - \mathbf{D}_{\text{par}}) \sum_{k=0}^{N-1} \mathbf{D}_{\text{par}}^k \\ &= \mathbf{I} - \mathbf{D}_{\text{par}}^N \\ &= \mathbf{I}.\end{aligned}$$

Because $\mathbf{I} - \mathbf{D}_{\text{par}}$ is an upper triangular matrix and all diagonal elements are one, $\mathbf{I} - \mathbf{D}_{\text{par}}$ is a regular matrix. Therefore, the sum of the infinite geometric series converges to $(\mathbf{I} - \mathbf{D}_{\text{par}})^{-1}$.

A.3. Proof of Theorem 2

Proof. Assume that the tree metric is given, and let $\mathbf{D}_{\text{par}}$ be its adjacency matrix. The element in the i-th row and j-th column of the adjacency matrix to the power of k is the number of paths from v_j to v_i with k steps. $\mathbf{D}_{\text{par}}$ is the adjacency matrix of a tree, and the number of paths is at most 1. If there is a path from v_j to v_i with k steps, $[\mathbf{D}_{\text{par}}^k]_{i,j}$ is one; otherwise, it is zero. Then if there is a path from v_j to v_i, $[(\mathbf{I} - \mathbf{D}_{\text{par}})^{-1}]_{i,j}$ is one; otherwise, it is zero. The existence of a path from v_j to v_i means that v_j is contained in the subtree rooted at v_i. From the definition of $P_{\text{sub}}(v_j|v_i)$, if v_j is contained in the subtree rooted at v_i, $P_{\text{sub}}(v_j|v_i)$ is one; otherwise, it is zero. We now have

$$\mu(\Gamma(v)) = \sum_{u \in \Gamma(v)} \mu(u) = \sum_{u \in \boldsymbol{V}_{\text{leaf}}} P_{\text{sub}}(u|v)\mu(u).$$

Therefore, if the tree metric is given and α approaches ∞, the soft tree-Wasserstein distance converges to the tree-Wasserstein distance that is,

$$\begin{aligned}W_{d_{\mathcal{T}}}^{\text{soft}}(\mu_i, \mu_j) &= \sum_{v \in \boldsymbol{V}} w_v \left| \sum_{x \in \boldsymbol{V}_{\text{leaf}}} P_{\text{sub}}(x|v) \left(\mu_i(x) - \mu_j(x)\right) \right|_{\alpha} \\ &= \sum_{v \in \boldsymbol{V}} w_v \left|\mu_i(\Gamma(v)) - \mu_j(\Gamma(v))\right|_{\alpha} \\ &\xrightarrow[\alpha \to \infty]{} W_{d_{\mathcal{T}}}(\mu_i, \mu_j)\end{aligned}$$

□

A.4. Additional Theoretical Analyses

In the formulation of the soft tree-Wasserstein distance, all nodes are contained in the subtree rooted at the root v_1. Furthermore, every node is contained in the subtree rooted at itself.

Theorem 5. *For all $u \in \boldsymbol{V}$, $P_{sub}(u|v_1) = 1$.*

Proof. We prove that the elements in the first row of $(\mathbf{I} - \mathbf{D}_{\text{par}})^{-1}$ are all one. Because $\mathbf{D}_{\text{par}}$ satisfies the conditions of Theorem 1, we have that

$$\begin{aligned}\mathbf{1}_N^\top \mathbf{D}_{\text{par}} &= (0, 1, \ldots, 1)\,, \\ \mathbf{1}_N^\top (\mathbf{I} - \mathbf{D}_{\text{par}}) &= (1, 0, \ldots, 0)\,.\end{aligned}$$

Since there exists the inverse matrix $(\mathbf{I} - \mathbf{D}_{\text{par}})^{-1}$, we multiply this inverse matrix with the above equation, yielding

$$\mathbf{1}_N^\top = (1, 0, \ldots, 0)\,(\mathbf{I} - \mathbf{D}_{\text{par}})^{-1}.$$

Therefore, the statement is true. □

Theorem 6. *For all $v \in \boldsymbol{V}$, $P_{sub}(v|v) = 1$.*

Proof. We prove that the diagonal elements of $(\mathbf{I}-\mathbf{D}_{\text{par}})^{-1}$ are all one. Because $\mathbf{I}-\mathbf{D}_{\text{par}}$ is an upper triangular matrix, $(\mathbf{I}-\mathbf{D}_{\text{par}})^{-1}$ is an upper triangular matrix. Because $\mathbf{I}-\mathbf{D}_{\text{par}}$ is an upper triangular matrix and all diagonal elements are one, all its eigenvalues are one. Then all eigenvalues of $(\mathbf{I}-\mathbf{D}_{\text{par}})^{-1}$ are one. Therefore, the diagonal elements of $(\mathbf{I}-\mathbf{D}_{\text{par}})^{-1}$ are all one. □

B. Additional Experimental Results

B.1. Additional Analyses of Batch Size

Figure 5 presents the time consumption of the tree-based methods when varying the batch size on AMAZON. Figure 6 illustrates the time consumption of the tree-based methods except for Flowtree. The results show that the time consumption of Quadtree increases linearly with the number of documents to be compared. However, the time consumption for the STW distance to compute a single batch is almost the same even if the batch size increases. As a result, if the batch size is sufficiently large, the STW distance is faster than that of Quadtree. Note that we implement the TSW distance by using the same formulation as the STW distance, which can be computed on a GPU. Figure 7, 8, 9, and 10 show the time consumption of all baseline methods and the STW distance when the batch size is varied from 500, 1000, 2500, and 5000. We omit datasets that contain only the number of training data below the batch size.

B.2. Additional Analyses of Depth Level

For the TSW and STW distances, we need to set the depth level of the tree as the hyperparameters. Figure 11 shows the time required to compare one document with 500 documents of the TSW and STW distances when varying the tree's depth level. The results show that, even if the depth level of the tree increases, the time consumption is almost the same.

B.3. Time Consumption on CPU

In this section, we show the time consumption of the STW distance on a CPU. We implement the STW distance with sparse matrix multiplications in SciPy. Table 4 shows the time consumption of the STW distance with sparse matrix multiplications on a CPU. Unfortunately, the results indicate that the STW distance with sparse matrix multiplications is slower than Quadtree. However, Quadtree is written in C++ and highly tuned. That is, if we implement the STW distance in the same way as Quadtree, the STW distance can be computed as fast as Quadtree on a CPU.

B.4. Analyses of Soft Tree-Wasserstein Distance

In the STW distance, we learn the probability of the tree's parent-child relationships by using the label information of documents, then we select the most probable parent node for each node. In this section, we show how this thresholding affects the accuracy. We refer to the STW distance with $\mathbf{D}_2$, which represents the probability of the parent-child relationship, and smooth approximation of the L1 norm as the soft-smooth-STW distance and the STW distance with smooth approximation of the L1 norm as the smooth-STW distance. We show the results in Table 5. By comparing the smooth-STW and soft-smooth-STW distances, the results show that this thresholding reduces the accuracy by about 1%.

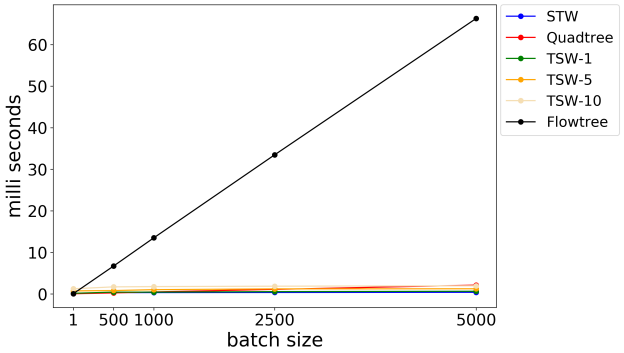

Figure 5. Average time consumption for all tree-based methods to compare one document with the number of batch size documents on AMAZON.

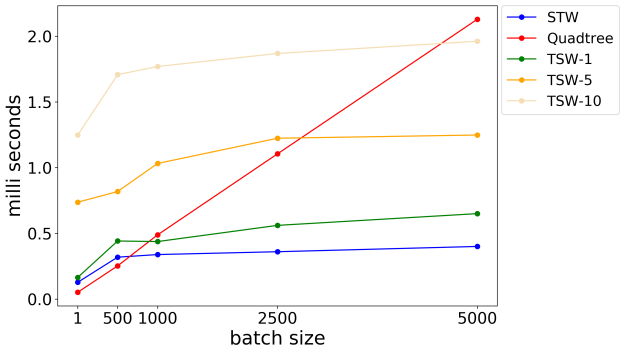

Figure 6. Average time consumption for comparing one document with the number of batch size documents on AMAZON.

B.5. Other Experimental Results

We show the loss value in the training in Figure 12.

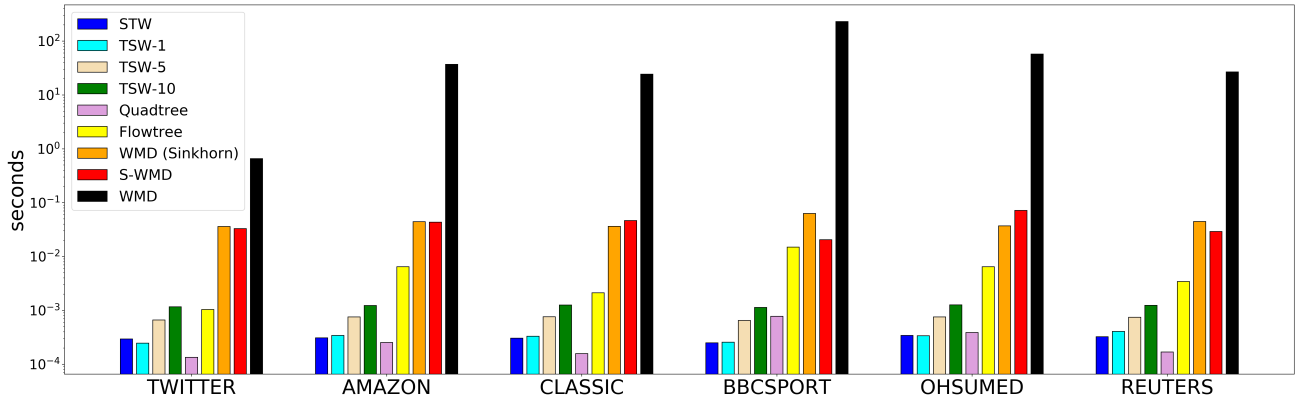

Figure 7. Average time consumption for comparing 500 documents with one document. For WMD (Sinkhorn), S-WMD, the STW distance, and the TSW distance, the batch size is set to 500.

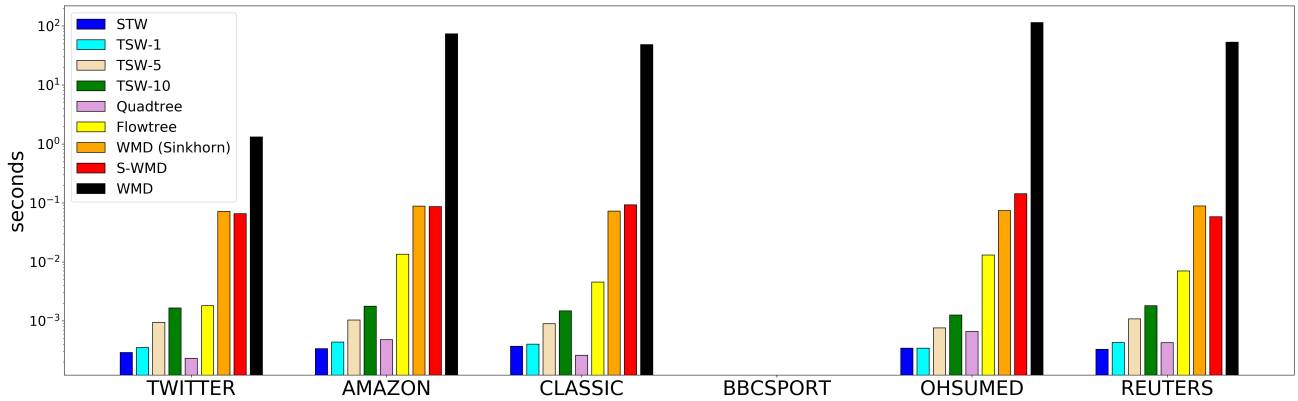

Figure 8. Average time consumption for comparing 1000 documents with one document. For the STW distance and the TSW distance, the batch size is set to 1000. For WMD (Sinkhorn) and S-WMD, the batch size is set to 500 due to the memory size limitations.

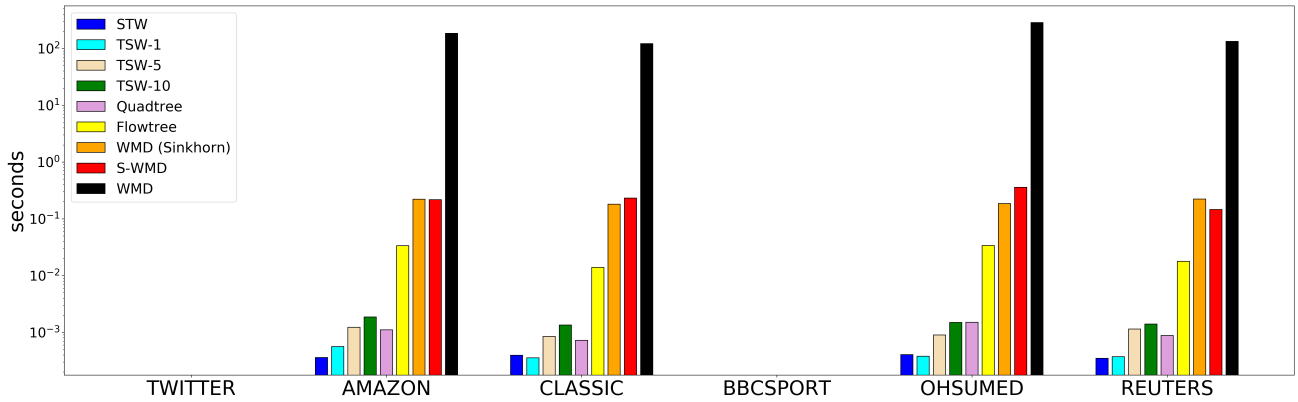

Figure 9. Average time consumption for comparing 2500 documents with one document. For the STW distance and the TSW distance, the batch size is set to 2500. For WMD (Sinkhorn) and S-WMD, the batch size is set to 500 due to the memory size limitations.

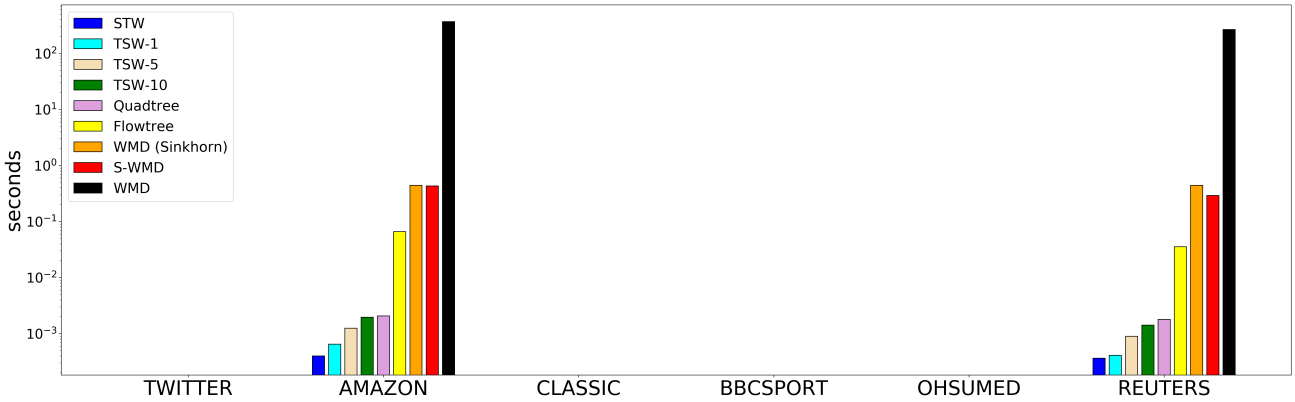

Figure 10. Average time consumption for comparing 5000 documents with one document. For the STW distance and the TSW distance, the batch size is set to 5000. For WMD (Sinkhorn) and S-WMD, the batch size is set to 500 due to the memory size limitations.

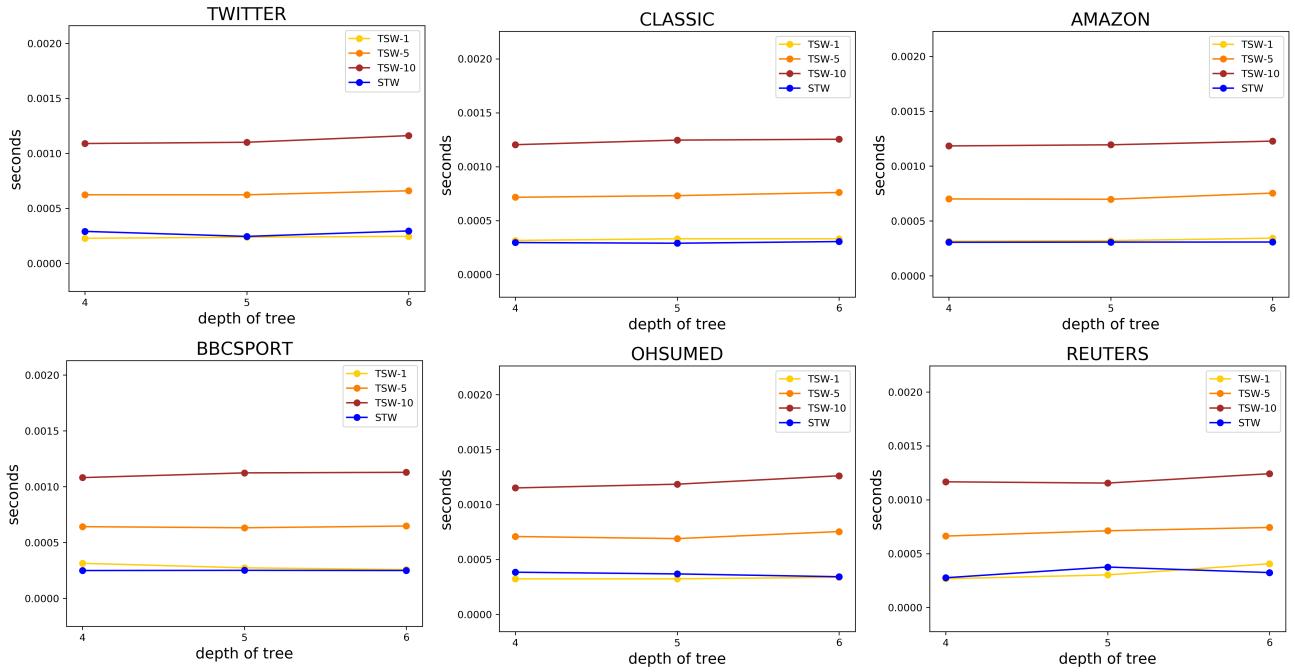

Figure 11. Average time consumption on all datasets for comparing one document with 500 documents when varying the depth level of the tree.

Table 4. Average time consumption to compare one document with 500 documents on a CPU [ms].

	TWITTER	AMAZON	CLASSIC	BBCSPORT	OHSUMED	REUTERS
Quadtree	0.13	0.25	0.16	0.77	0.39	0.17
STW (sparse)	1.78	4.79	3.77	7.65	6.42	4.49

Table 5. kNN test error rate.

	TWITTER	AMAZON	CLASSIC	BBCSPORT	OHSUMED	REUTERS
soft-smooth-STW	29.9 ± 1.3	8.4 ± 0.4	5.1 ± 0.2	4.5 ± 1.0	44.1	6.5
smooth-STW	30.0 ± 0.8	10.6 ± 0.4	9.6 ± 0.9	4.5 ± 0.9	45.6	6.5
STW	28.9 ± 0.7	10.1 ± 0.7	4.4 ± 0.7	3.4 ± 0.8	40.2	4.4

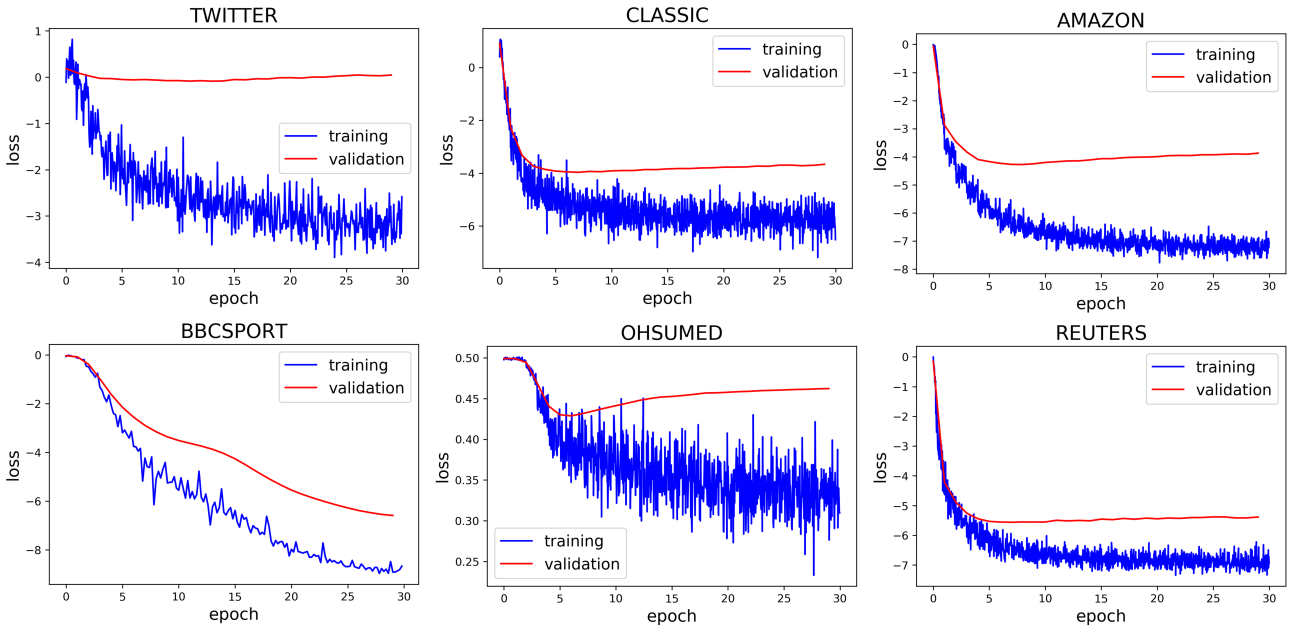

Figure 12. The loss value for all datasets.