tf.train.sdca_optimizer

```
sdca_optimizer(
    sparse_example_indices,
    sparse_feature_indices,
    sparse_feature_values,
    dense_features,
    example_weights,
    example_labels,
    sparse_indices,
    sparse_weights,
    dense_weights,
    example_state_data,
    loss_type,
    l1,
    l2,
    num_loss_partitions,
    num_inner_iterations,
    adaptative=False,
    name=None
)
```

Defined in `tensorflow/python/ops/gen_sdca_ops.py`.

Distributed version of Stochastic Dual Coordinate Ascent (SDCA) optimizer for

linear models with L1 + L2 regularization. As global optimization objective is strongly-convex, the optimizer optimizes the dual objective at each step. The optimizer applies each update one example at a time. Examples are sampled uniformly, and the optimizer is learning rate free and enjoys linear convergence rate.

Proximal Stochastic Dual Coordinate Ascent.
Shai Shalev-Shwartz, Tong Zhang. 2012

$$LossObjective = \sum f_{i}(wx_{i}) + (l2/2) \square \ |w|^{2} + l1 \square \ |w|$$

Adding vs. Averaging in Distributed Primal-Dual Optimization.
Chenxin Ma, Virginia Smith, Martin Jaggi, Michael I. Jordan, Peter Richtarik, Martin Takac. 2015

Stochastic Dual Coordinate Ascent with Adaptive Probabilities.
Dominik Csiba, Zheng Qu, Peter Richtarik. 2015

Args:

- `sparse_example_indices`: A list of `Tensor` objects with type `int64`. a list of vectors which contain example indices.
- `sparse_feature_indices`: A list with the same length as `sparse_example_indices` of `Tensor` objects with type `int64`. a list of vectors which contain feature indices.
- `sparse_feature_values`: A list of `Tensor` objects with type `float32`. a list of vectors which contains feature value associated with each feature group.
- `dense_features`: A list of `Tensor` objects with type `float32`. a list of matrices which contains the dense feature values.
- `example_weights`: A `Tensor` of type `float32`. a vector which contains the weight associated with each example.

- `example_labels` : A `Tensor` of type `float32` . a vector which contains the label/target associated with each example.
- `sparse_indices` : A list with the same length as `sparse_example_indices` of `Tensor` objects with type `int64` . a list of vectors where each value is the indices which has corresponding weights in sparse_weights. This field maybe omitted for the dense approach.
- `sparse_weights` : A list with the same length as `sparse_example_indices` of `Tensor` objects with type `float32` . a list of vectors where each value is the weight associated with a sparse feature group.
- `dense_weights` : A list with the same length as `dense_features` of `Tensor` objects with type `float32` . a list of vectors where the values are the weights associated with a dense feature group.
- `example_state_data` : A `Tensor` of type `float32` . a list of vectors containing the example state data.
- `loss_type` : A `string` from: `"logistic_loss", "squared_loss", "hinge_loss", "smooth_hinge_loss"` . Type of the primal loss. Currently SdcaSolver supports logistic, squared and hinge losses.
- `l1` : A `float` . Symmetric l1 regularization strength.
- `l2` : A `float` . Symmetric l2 regularization strength.
- `num_loss_partitions` : An `int` that is `>= 1` . Number of partitions of the global loss function.
- `num_inner_iterations` : An `int` that is `>= 1` . Number of iterations per mini-batch.
- `adaptative` : An optional `bool` . Defaults to `False` . Whether to use Adapative SDCA for the inner loop.
- `name` : A name for the operation (optional).

Returns:

A tuple of `Tensor` objects (out_example_state_data, out_delta_sparse_weights, out_delta_dense_weights).

- `out_example_state_data` : A `Tensor` of type `float32` . a list of vectors containing the updated example state data.
- `out_delta_sparse_weights` : A list with the same length as `sparse_example_indices` of `Tensor` objects with type `float32` . a list of vectors where each value is the delta weights associated with a sparse feature group.
- `out_delta_dense_weights` : A list with the same length as `dense_features` of `Tensor` objects with type `float32` . a list of vectors where the values are the delta weights associated with a dense feature group.

Last updated November 2, 2017.

English

Terms | Privacy