Self Study

Based on my Understanding of the Subject

Compilation of my study materials

Author: Dr. Md Arafat Hossain Khan

Last updated : October 19, 2022

Contents

Chapter 1

Fundamentals

Chapter 2

Hahn-Banach Theorems

2.1 Helly, Hahn-Banach Analytic Form

The theorem is named for the mathematicians Hans Hahn and Stefan Banach, who proved it independently in the late 1920s. The special case of the theorem for the space $C[a,b]$ of continuous functions on an interval was proved earlier (in 1912) by Eduard Helly, and a more general extension theorem, the M. Riesz extension theorem, from which the Hahn–Banach theorem can be derived, was proved in 1923 by Marcel Riesz.

Definition 1: Sublinear Function

Let X be a vector space over a field $\mathbb{K}$, where $\mathbb{K}$ is either the real numbers $\mathbb{R}$ or complex numbers $\mathbb{C}$. A real-valued function $p : X \to \mathbb{R}$ on X is called a sublinear function (or a sublinear functional if $\mathbb{K} = \mathbb{R}$), and also sometimes called a quasi-seminorm or a Banach functional, if it has these two properties:

1. **Positive homogeneity/Nonnegative homogeneity**: $p(rx) = rp(x)$ for all real $r \geq 0$ and all $x \in X$. This condition holds if and only if $p(rx) \leq rp(x)$ for all positive real $r > 0$ and all $x \in X$.
2. **Subadditivity/Triangle inequality**: $p(x+y) \leq p(x)+p(y)$ for all $x, y \in X$. This subadditivity condition requires p to be real-valued.

A real-valued function $f : M \to \mathbb{R}$ defined on $M \subseteq X$ is said to be dominated (above) by a function $p : X \to \mathbb{R}$ if $f(m) \leq p(m)$ for every $m \in M$. Hence the reason why the following version of the Hahn-Banach theorem is called the dominated extension theorem [1].

Theorem 2.1.1: Helly, Hahn-Banach Analytic Form

If $p : X \to \mathbb{R}$ is a sublinear function (such as a norm or seminorm for example) defined on a real vector space X then any linear functional defined on a vector subspace of X that is dominated above by p has at least one linear extension to all of X that is also dominated above by p. In other words, any linear functional defined on a vector subspace of X that is dominated above by a sublinear function $p : X \to \mathbb{R}$ has at least one linear extension to all of X that is also dominated above by p.

Explicitly, if $p : X \to \mathbb{R}$ is a sublinear function, which by definition means that it satisfies

$$p(x+y) \leq p(x) + p(y) \quad \text{and} \quad p(tx) = tp(x) \quad \forall\, x, y \in X \quad \text{and} \quad \forall t \in \mathbb{R}_{\geq 0},$$

and if $f : M \to \mathbb{R}$ is a linear functional defined on a vector subspace M of X such that

$$f(m) \le p(m) \quad \text{for all } m \in M$$

then there exists a linear functional $F : X \to \mathbb{R}$ such that

$$F(m) = f(m) \quad \text{for all } m \in M,$$

$$F(x) \le p(x) \qquad \text{for all } x \in X.$$

Moreover, if p is a seminorm then $|F(x)| \le p(x)$ necessarily holds for all $x \in X$.

Proof of theorem 2.1.1. The proof of depends on Zorn's lemma. ■

Chapter 3

Timed Feature Analysis

3.1 RNN

Let us suppose the input sequence is,

$$\mathbf{X} = \left[\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(t-1)}, \mathbf{x}^{(t)}, \mathbf{x}^{(t+1)}, \ldots, \mathbf{x}^{(T-1)}, \mathbf{x}^{(T)}\right].$$

where, $\mathbf{x}^{(t)} \in \mathbb{R}^n$, $\forall t \in \mathbb{N}_{\leq T}$. Note that these inputs can also be vectors from higher dimensional space, e.g. for image sequence, $\mathbf{x}^{(t)} \in \mathbb{R}^{n \times m}$, $\forall t \in \mathbb{N}_{\leq T}$ for an $n \times m$-pixled image. For current explanation we will consider $\mathbf{x}^{(t)}$ to be 1-dimensional $\forall t \in \mathbb{N}_{\leq T}$.

Now, we do a linear transformation of $\mathbf{X}$ using $\mathbf{U}_{k \times n}$ as follows,

$$\mathbf{U}_{k\times n}\mathbf{X}_{n\times T} = \left[\mathbf{Ux}^{(1)}, \mathbf{Ux}^{(2)}, \ldots, \mathbf{Ux}^{(t-1)}, \mathbf{Ux}^{(t)}, \mathbf{Ux}^{(t+1)}, \ldots, \mathbf{Ux}^{(T-1)}, \mathbf{Ux}^{(T)}\right].$$

where, $\mathbf{Ux}^{(t)} \in \mathbb{R}^k$, $\forall t \in \mathbb{N}_{\leq T}$. Until this point we do not have any interaction between $\mathbf{x}^{(t-1)}$ and $\mathbf{x}^{(t)}$. We introduce a new matrix $\mathbf{W}_{k\times\ell}$ and initialize a vector $\mathbf{h}^{(0)} \in \mathbb{R}^\ell$. From here we create the following sequence of vectors,

$$\begin{aligned}
\mathbf{h}^{(1)} &= \tanh\left(\mathbf{Wh}^{(0)} + \mathbf{Ux}^{(1)}\right),\\
\mathbf{h}^{(2)} &= \tanh\left(\mathbf{Wh}^{(1)} + \mathbf{Ux}^{(2)}\right),\\
&\vdots\\
\mathbf{h}^{(t)} &= \tanh\left(\mathbf{Wh}^{(t-1)} + \mathbf{Ux}^{(t)}\right),\\
&\vdots\\
\mathbf{h}^{(T)} &= \tanh\left(\mathbf{Wh}^{(T-1)} + \mathbf{Ux}^{(T)}\right)
\end{aligned}$$

We may also want to introduce a bias vectors $\{\mathbf{b}^{(t)}\}_{t\in\mathbb{N}_{\leq T}} \subset \mathbb{R}^k$ and obtain the following,

$$\mathbf{h}^{(t)} = \tanh\left(\mathbf{Wh}^{(t-1)} + \mathbf{Ux}^{(t)} + \mathbf{b}^{(t)}\right), \ \forall t \in \mathbb{N}_{\leq T}.$$

Finally, we choose weight matrix $\mathbf{V}_{s\times k}$ and bias vectors $\{\mathbf{c}^{(t)}\}_{t\in\mathbb{N}_{\leq T}} \subset \mathbb{R}^s$ to produce the estimated output,

$$\hat{\mathbf{y}}^{(t)} = \text{softmax}\left(c^{(t)} + \mathbf{Vh}^{(t)}\right).$$

For simplicity, we may choose,

$$\mathbf{b} = \mathbf{b}^{(t)} \text{ and } \mathbf{c} = \mathbf{c}^{(t)}, \ \forall t \in \mathbb{N}_{\leq T}.$$

Drawback of RNN

Vanishing gradient problem. As a result memory does not sustain for long.

3.2 LSTM

Let us first define the variables,

- $\mathbf{x}^{(t)} \in \mathbb{R}^d$: input vector to the LSTM unit
- $\mathbf{f}^{(t)} \in (0,1)^h$: forget gate's activation vector
- $\mathbf{i}^{(t)} \in (0,1)^h$: input/update gate's activation vector
- $\mathbf{o}^{(t)} \in (0,1)^h$: output gate's activation vector
- $\mathbf{h}^{(t)} \in (-1,1)^h$: hidden state vector also known as output vector of the LSTM unit
- $\tilde{\mathbf{c}}^{(t)} \in (-1,1)^h$: cell input activation vector
- $\mathbf{c}^{(t)} \in \mathbb{R}^h$: cell state vector
- $\mathbf{W} \in \mathbb{R}^{h\times d}$, $\mathbf{U} \in \mathbb{R}^{h\times h}$ and $\mathbf{b} \in \mathbb{R}^h$: weight matrices and bias vector parameters which need to be learned during training

where the superscripts d and h refer to the number of input features and number of hidden units, respectively. Following are the activation functions

- σ_g: sigmoid function.
- σ_c: hyperbolic tangent function.
- σ_h: hyperbolic tangent function or identity function.

We start with the similar type of formulations like RNN,

$$\begin{aligned}
\mathbf{f}^{(t)} &= \sigma_g(\mathbf{W}_f\mathbf{x}^{(t)} + \mathbf{U}_f\mathbf{h}^{(t-1)} + \mathbf{b}_f)\\
\mathbf{i}^{(t)} &= \sigma_g(\mathbf{W}_i\mathbf{x}^{(t)} + \mathbf{U}_i\mathbf{h}^{(t-1)} + \mathbf{b}_i)\\
\tilde{\mathbf{c}}^{(t)} &= \sigma_c(\mathbf{W}_c\mathbf{x}^{(t)} + \mathbf{U}_c\mathbf{h}^{(t-1)} + \mathbf{b}_c)\\
\mathbf{o}^{(t)} &= \sigma_g(\mathbf{W}_o\mathbf{x}^{(t)} + \mathbf{U}_o\mathbf{h}^{(t-1)} + \mathbf{b}_o)
\end{aligned}$$

Then we produce the cell state vector and hidden state vector as follows,

$$\begin{aligned}
\mathbf{c}^{(t)} &= \mathbf{f}^{(t)} \odot \mathbf{c}^{(t-1)} + \mathbf{i}^{(t)} \odot \tilde{\mathbf{c}}^{(t)}\\
\mathbf{h}^{(t)} &= \mathbf{o}^{(t)} \odot \sigma_h(\mathbf{c}^{(t)})
\end{aligned}$$

where the initial values are $\mathbf{c}^{(0)} = 0$ and $\mathbf{h}^{(0)} = 0$ and the operator $\odot$ denotes the Hadamard product (element-wise product). The superscript t indexes the time step.

3.3 Peephole terms in LSTM

We modify the LSTM equations as follows,

$$
\begin{aligned}
\mathbf{i}^{(t)} &= \sigma_g\left(\mathbf{W}_i\mathbf{x}^{(t)} + \mathbf{U}_i\mathbf{h}^{(t-1)} + \mathbf{b}_i + \mathbf{w}_{pi} \odot \mathbf{c}^{(t-1)}\right)\\
\mathbf{f}^{(t)} &= \sigma_g\left(\mathbf{W}_f\mathbf{x}^{(t)} + \mathbf{U}_f\mathbf{h}^{(t-1)} + \mathbf{b}_f + \mathbf{w}_{pf} \odot \mathbf{c}^{(t-1)}\right)\\
\tilde{\mathbf{c}}^{(t)} &= \sigma_c\left(\mathbf{W}_c\mathbf{x}^{(t)} + \mathbf{U}_c\mathbf{h}^{(t-1)} + \mathbf{b}_c\right)\\
\mathbf{c}^{(t)} &= \mathbf{f}^{(t)} \odot \mathbf{c}^{(t-1)} + \mathbf{i}^{(t)} \odot \tilde{\mathbf{c}}^{(t)}\\
\mathbf{o}^{(t)} &= \sigma_g\left(\mathbf{W}_o\mathbf{x}^{(t)} + \mathbf{U}_o\mathbf{h}^{(t-1)} + \mathbf{b}_o + \mathbf{w}_{po} \odot \mathbf{c}^{(t)}\right)\\
\mathbf{h}^{(t)} &= \mathbf{o}^{(t)} \odot \sigma_h(\mathbf{c}^{(t)})
\end{aligned}
$$

where $\mathbf{w}_{pi}$, $\mathbf{w}_{pf}$ and $\mathbf{w}_{po}$ are learnable parameters.

3.4 TimeLSTM

3.4.1 TLSTM1

We introduce time gate's activation vector $\mathcal{T}^{(t)}$. Note that instead of a similar matrix like $\mathbf{U}_i$, $\mathbf{U}_f$ or $\mathbf{U}_o$, we use vectors $\{\mathbf{u}^{(t)}\}_{t\in\mathbb{N}_{\leq T}}$. Also note that for inputs,

$$
\left[\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(t-1)}, \mathbf{x}^{(t)}, \mathbf{x}^{(t+1)}, \ldots, \mathbf{x}^{(T-1)}, \mathbf{x}^{(T)}\right].
$$

We have corresponding time vector,

$$
\left[\tau^{(1)}, \tau^{(2)}, \ldots, \tau^{(t-1)}, \tau^{(t)}, \tau^{(t+1)}, \ldots, \tau^{(T-1)}, \tau^{(T)}\right] \in \mathbb{R}^T.
$$

We have the following TLSTM1 formulation,

$$
\begin{aligned}
\mathbf{i}^{(t)} &= \sigma_g\left(\mathbf{W}_i\mathbf{x}^{(t)} + \mathbf{U}_i\mathbf{h}^{(t-1)} + \mathbf{b}_i + \mathbf{w}_{pi} \odot \mathbf{c}^{(t-1)}\right)\\
\mathcal{T}^{(t)} &= \sigma_g\left(\mathbf{W}_{\mathcal{T}}\mathbf{x}^{(t)} + \sigma_g\left(\tau^{(t)}\mathbf{u}_{\mathcal{T}}\right) + \mathbf{b}_{\mathcal{T}} + \mathbf{w}_{p\mathcal{T}} \odot \mathbf{c}^{(t-1)}\right)\\
\mathbf{f}^{(t)} &= \sigma_g\left(\mathbf{W}_f\mathbf{x}^{(t)} + \mathbf{U}_f\mathbf{h}^{(t-1)} + \mathbf{b}_f + \mathbf{w}_{pf} \odot \mathbf{c}^{(t-1)}\right)\\
\tilde{\mathbf{c}}^{(t)} &= \sigma_c\left(\mathbf{W}_c\mathbf{x}^{(t)} + \mathbf{U}_c\mathbf{h}^{(t-1)} + \mathbf{b}_c\right)\\
\mathbf{c}^{(t)} &= \mathbf{f}^{(t)} \odot \mathbf{c}^{(t-1)} + \mathcal{T}^{(t)} \odot \mathbf{i}^{(t)} \odot \tilde{\mathbf{c}}^{(t)}\\
\mathbf{o}^{(t)} &= \sigma_g\left(\mathbf{W}_o\mathbf{x}^{(t)} + \tau^{(t)}\mathbf{v}_{\mathcal{T}} + \mathbf{U}_o\mathbf{h}^{(t-1)} + \mathbf{b}_o + \mathbf{w}_{po} \odot \mathbf{c}^{(t)}\right)\\
\mathbf{h}^{(t)} &= \mathbf{o}^{(t)} \odot \sigma_h(\mathbf{c}^{(t)})
\end{aligned}
$$

3.4.2 TLSTM3

TLSTM3 uses two time gates,

$$\begin{aligned}
\mathbf{i}^{(t)} &= \sigma_g\left(\mathbf{W}_i\mathbf{x}^{(t)} + \mathbf{U}_i\mathbf{h}^{(t-1)} + \mathbf{b}_i + \mathbf{w}_{pi} \odot \mathbf{c}^{(t-1)}\right)\\
\mathcal{T}_1^{(t)} &= \sigma_g\left(\mathbf{W}_{\mathcal{T}_1}\mathbf{x}^{(t)} + \sigma_g\left(\tau^{(t)}\mathbf{u}_{\mathcal{T}_1}\right) + \mathbf{b}_{\mathcal{T}_1} + \mathbf{w}_{p\mathcal{T}_1} \odot \mathbf{c}^{(t-1)}\right)\\
\mathcal{T}_2^{(t)} &= \sigma_g\left(\mathbf{W}_{\mathcal{T}_2}\mathbf{x}^{(t)} + \sigma_g\left(\tau^{(t)}\mathbf{u}_{\mathcal{T}_2}\right) + \mathbf{b}_{\mathcal{T}_2} + \mathbf{w}_{p\mathcal{T}_2} \odot \mathbf{c}^{(t-1)}\right)\\
\mathbf{f}^{(t)} &= \sigma_g\left(\mathbf{W}_f\mathbf{x}^{(t)} + \mathbf{U}_f\mathbf{h}^{(t-1)} + \mathbf{b}_f + \mathbf{w}_{pf} \odot \mathbf{c}^{(t-1)}\right)\\
\tilde{\mathbf{c}}^{(t)} &= \sigma_c\left(\mathbf{W}_c\mathbf{x}^{(t)} + \mathbf{U}_c\mathbf{h}^{(t-1)} + \mathbf{b}_c\right)\\
\bar{\mathbf{c}}^{(t)} &= \left(1 - \mathbf{i}^{(t)} \odot \mathcal{T}_1^{(t)}\right) \odot \mathbf{c}^{(t-1)} + \mathbf{i}^{(t)} \odot \mathcal{T}_1^{(t)} \odot \tilde{\mathbf{c}}^{(t)}\\
\mathbf{c}^{(t)} &= \left(1 - \mathbf{i}^{(t)}\right) \odot \mathbf{c}^{(t-1)} + \mathcal{T}_2^{(t)} \odot \mathbf{i}^{(t)} \odot \tilde{\mathbf{c}}^{(t)}\\
\mathbf{o}^{(t)} &= \sigma_g\left(\mathbf{W}_o\mathbf{x}^{(t)} + \tau^{(t)}\mathbf{v}_{\mathcal{T}} + \mathbf{U}_o\mathbf{h}^{(t-1)} + \mathbf{b}_o + \mathbf{w}_{po} \odot \mathbf{c}^{(t)}\right)\\
\mathbf{h}^{(t)} &= \mathbf{o}^{(t)} \odot \sigma_h(\bar{\mathbf{c}}^{(t)})
\end{aligned}$$

Note that, we used $\mathcal{T}$ as the subscript of $\mathbf{v}$. We could have used $\mathbf{v}_{\mathcal{T}_1}$ or $\mathbf{v}_{\mathcal{T}_2}$. But to avoid confusion we used an independent symbol $\mathcal{T}$. Thus making $\mathbf{v}_{\mathcal{T}}$ invariant under actual time, time gates and the number of time gates.

3.5 TimeLSTM+Time2Vec

Recall the time vector,

$$\left[\tau^{(1)}, \tau^{(2)}, \ldots, \tau^{(t-1)}, \tau^{(t)}, \tau^{(t+1)}, \ldots, \tau^{(T-1)}, \tau^{(T)}\right] \in \mathbb{R}^T.$$

The operator **t2v** takes a scalar time as input and converts it to a vector of some dimension $\alpha + 1$ as follows,

$$\mathbf{t2v}\left(\tau^{(t)}\right) = \left[\omega_0\tau^{(t)} + \phi_0, \mathcal{F}\left(\omega_1\tau^{(t)} + \phi_1\right), \ldots, \mathcal{F}\left(\omega_\alpha\tau^{(t)} + \phi_\alpha\right), \right]$$

One choice of the activation function $\mathcal{F}$ is sine function. Experiemnts showed that periodic activation functions are specially useful to understand the repeated patterns.

We also introduce the matrix $\mathbf{U}_{\mathcal{T}}$ in place of the vector $\mathbf{u}_{\mathcal{T}}$ and $\mathbf{V}_{\mathcal{T}}$ in place of the vector $\mathbf{v}_{\mathcal{T}}$ with proper dimension.

3.5.1 TLSTM1+Time2Vec

$$\begin{aligned}
\mathbf{i}^{(t)} &= \sigma_g\left(\mathbf{W}_i\mathbf{x}^{(t)} + \mathbf{U}_i\mathbf{h}^{(t-1)} + \mathbf{b}_i + \mathbf{w}_{pi} \odot \mathbf{c}^{(t-1)}\right)\\
\mathcal{T}^{(t)} &= \sigma_g\left(\mathbf{W}_{\mathcal{T}}\mathbf{x}^{(t)} + \sigma_g\left(\mathbf{U}_{\mathcal{T}}\mathbf{t2v}\left(\tau^{(t)}\right)\right) + \mathbf{b}_{\mathcal{T}} + \mathbf{w}_{p\mathcal{T}} \odot \mathbf{c}^{(t-1)}\right)\\
\mathbf{f}^{(t)} &= \sigma_g\left(\mathbf{W}_f\mathbf{x}^{(t)} + \mathbf{U}_f\mathbf{h}^{(t-1)} + \mathbf{b}_f + \mathbf{w}_{pf} \odot \mathbf{c}^{(t-1)}\right)\\
\tilde{\mathbf{c}}^{(t)} &= \sigma_c\left(\mathbf{W}_c\mathbf{x}^{(t)} + \mathbf{U}_c\mathbf{h}^{(t-1)} + \mathbf{b}_c\right)\\
\mathbf{c}^{(t)} &= \mathbf{f}^{(t)} \odot \mathbf{c}^{(t-1)} + \mathcal{T}^{(t)} \odot \mathbf{i}^{(t)} \odot \tilde{\mathbf{c}}^{(t)}\\
\mathbf{o}^{(t)} &= \sigma_g\left(\mathbf{W}_o\mathbf{x}^{(t)} + \mathbf{V}_{\mathcal{T}}\mathbf{t2v}\left(\tau^{(t)}\right) + \mathbf{U}_o\mathbf{h}^{(t-1)} + \mathbf{b}_o + \mathbf{w}_{po} \odot \mathbf{c}^{(t)}\right)\\
\mathbf{h}^{(t)} &= \mathbf{o}^{(t)} \odot \sigma_h(\mathbf{c}^{(t)})
\end{aligned}$$

3.5.2 TLSTM3+Time2Vec

$$\begin{aligned}
\mathbf{i}^{(t)} &= \sigma_g\left(\mathbf{W}_i\mathbf{x}^{(t)} + \mathbf{U}_i\mathbf{h}^{(t-1)} + \mathbf{b}_i + \mathbf{w}_{pi} \odot \mathbf{c}^{(t-1)}\right)\\
\mathcal{T}_1^{(t)} &= \sigma_g\left(\mathbf{W}_{\mathcal{T}_1}\mathbf{x}^{(t)} + \sigma_g\left(\mathbf{U}_{\mathcal{T}_1}\mathbf{t2v}\left(\tau^{(t)}\right)\right) + \mathbf{b}_{\mathcal{T}_1} + \mathbf{w}_{p\mathcal{T}_1} \odot \mathbf{c}^{(t-1)}\right)\\
\mathcal{T}_2^{(t)} &= \sigma_g\left(\mathbf{W}_{\mathcal{T}_2}\mathbf{x}^{(t)} + \sigma_g\left(\mathbf{U}_{\mathcal{T}_2}\mathbf{t2v}\left(\tau^{(t)}\right)\right) + \mathbf{b}_{\mathcal{T}_2} + \mathbf{w}_{p\mathcal{T}_2} \odot \mathbf{c}^{(t-1)}\right)\\
\mathbf{f}^{(t)} &= \sigma_g\left(\mathbf{W}_f\mathbf{x}^{(t)} + \mathbf{U}_f\mathbf{h}^{(t-1)} + \mathbf{b}_f + \mathbf{w}_{pf} \odot \mathbf{c}^{(t-1)}\right)\\
\tilde{\mathbf{c}}^{(t)} &= \sigma_c\left(\mathbf{W}_c\mathbf{x}^{(t)} + \mathbf{U}_c\mathbf{h}^{(t-1)} + \mathbf{b}_c\right)\\
\bar{\mathbf{c}}^{(t)} &= \left(1 - \mathbf{i}^{(t)} \odot \mathcal{T}_1^{(t)}\right) \odot \mathbf{c}^{(t-1)} + \mathbf{i}^{(t)} \odot \mathcal{T}_1^{(t)} \odot \tilde{\mathbf{c}}^{(t)}\\
\mathbf{c}^{(t)} &= \left(1 - \mathbf{i}^{(t)}\right) \odot \mathbf{c}^{(t-1)} + \mathcal{T}_2^{(t)} \odot \mathbf{i}^{(t)} \odot \tilde{\mathbf{c}}^{(t)}\\
\mathbf{o}^{(t)} &= \sigma_g\left(\mathbf{W}_o\mathbf{x}^{(t)} + \mathbf{V}_{\mathcal{T}}\mathbf{t2v}\left(\tau^{(t)}\right) + \mathbf{U}_o\mathbf{h}^{(t-1)} + \mathbf{b}_o + \mathbf{w}_{po} \odot \mathbf{c}^{(t)}\right)\\
\mathbf{h}^{(t)} &= \mathbf{o}^{(t)} \odot \sigma_h(\bar{\mathbf{c}}^{(t)})
\end{aligned}$$

Bibliography

[1] Wikipedia contributors. *Hahn–Banach theorem — Wikipedia, The Free Encyclopedia.* [Online; accessed 24-May-2022]. 2022. URL: https://en.wikipedia.org/w/index.php?title=Hahn%E2%80%93Banach_theorem&oldid=1089256573 (page - 2).