

## ELEKTR DVIGATELLARINING TEXNIK HOLATINI NAZORAT QILISH VA DIAGNOSTIKA QILISH UCHUN PRONIY MODELLARINI YARATISH.

*Namangan muhandislik-texnologiya instituti*

**Djurayev Sherzod Sobirjonovich**

**Onarqulov Maqsad Karimberdiyevich**

**Annotatsiya** Elektr dvigatellar texnologiyasining tez rivojlanishi ularning ishonchli nazorati va diagnostikasi uchun samarali tizimlarga bo'lgan talabni oshirdi. Dvigatellar texnik holatini diagnostika qilishda samarali usullardan biri – Proniy modellaridir. Ushbu modellar vaqt bo'yicha signal xatti-harakatlarini tahlil qilish orqali elektr dvigatellarning texnik holatini baholash imkonini beradi. Maqolada Proniy modelining matematik asoslari keltirilib, uni elektr dvigatellar signallarini tahlil qilishda qo'llashning afzalliklari ko'rsatildi. Bundan tashqari, Proniy modellarini takomillashtirish uchun takroriy neyron tarmoqlarini (TNT) qo'llash usullari ham o'rganildi.

**Kalit so'zlar.**Elektr dvigatellari, diagnostika, Proniy modeli, takroriy neyron tarmoqlar, elektromexanik tizimlar, signal tahlili, ishonchli nazorat, matematik modellashtirish, real vaqt kuzatuv, nosozlikni aniqlash, statik tahlil, dinamik tahlil, mexanik xatolar, elektromagnit jarayonlar.

**Kirish.** Elektr dvigatellarning texnik holatini nazorat qilish va diagnostika qilish ularning ishonchli ishlashini ta'minlashda muhim ahamiyatga ega. Dvigatelning noto'g'ri ishlashi energiya sarfini oshirishi, ishlab chiqarish samaradorligini pasaytirishi va uskunalarning muddatidan oldin ishdan chiqishiga olib kelishi mumkin. Shuning uchun dvigatelning texnik holatini real vaqt rejimida kuzatish va diagnostika qilish texnologiyalarini joriy etish zarur. Bu borada Proniy modellaridan foydalanish eng samarali yondashuvlardan biri bo'lib, u signal komponentlarini aniqlashga yordam beradi.

## Proniy modelining matematik asosi

Proniy usuli, birinchi navbatda, signalni bir nechta eksponensial funktsiyalar yig'indisi sifatida taqdim etishga asoslanadi. Ushbu yondashuv dvigatellar kabi dinamik tizimlar uchun o'z vaqtida o'zgaruvchan signal ma'lumotlarini aniq modellashtirishga imkon beradi.

### Matematik model:

$$y(t) = \sum_{i=1}^n A_i e^{\alpha_i t} \cos(\omega_i t + \varphi_i)$$

bu yerda:

- $A_i$  – amplitudalar,
- $\alpha_i$  – eksponent ko'rsatkichlari,
- $\omega_i$  – burchak chastotalar,
- $\varphi_i$  – fazalar,
- $t$  – vaqt o'zgaruvchisi,
- $n$  – komponentlar soni.
- Bu model orqali signalning murakkab komponentlarini aniqlash va ularning vaqt bo'yicha evolyutsiyasini kuzatish mumkin.

### Proniy modelining diagnostikadagi roli

Elektr dvigatellarning signalini Proniy modeli yordamida tahlil qilish dvigatelning ichki nosozliklari, mexanik yoki elektr muammolarini erta bosqichda aniqlashga yordam beradi. Masalan, signalning asosiy komponentlarini tahlil qilish orqali stator yoki rotorning mexanik xatolarini aniqlash mumkin. **Takroriy neyron tarmoqlarini Proniy modeliga integratsiya qilish**

Takroriy neyron tarmoqlari (TNT) vaqt bo'yicha ketma-ket signallarni tahlil qilish va prognoz qilishda samarali vositalardan biridir. TNT'lar o'tmish ma'lumotlarini tahlil qilib, kelajakdagi qiymatlarni aniq prognoz qila oladi, bu esa Proniy modelini yanada

rivojlantirish va elektr dvigatellarning texnik holatini kuzatishni optimallashtirish imkonini beradi.

Elektr dvigatelning signal komponentalari asosan uchta asosiy qismga bo'linadi, ular elektr dvigatelning ish jarayonini ifodalash uchun ishlatiladi. Har bir komponent dvigatel holatini baholash va mumkin bo'lgan nosozliklarni aniqlashda muhim rol o'ynaydi. Ushbu komponentalar:

### 1. **Asosiy chastota komponentasi:**

- Bu komponent elektr dvigatelining ishlashida muhim bo'lgan asosiy sinusoida signali hisoblanadi. Odatda, bu komponent dvigatelni quvvatlantiruvchi tarmoq chastotasi bilan bog'liq bo'ladi (masalan, 50 yoki 60 Gts).
- Asosiy chastota komponentasi dvigatelning mexanik harakatini va aylanishini ifodalaydi. Bu komponent dvigatelning normal ishlashini ko'rsatadi.

### 2. **Garmonika komponentalari:**

- Garmonika komponentalari asosiy chastotadan ko'paygan yoki kamaygan chastotalarda paydo bo'lgan sinusoida to'liqlaridir. Ular asosan  $2x$ ,  $3x$  va undan yuqori tartibdagi chastotalarda kuzatiladi.
- Garmonika komponentalari tahlili orqali elektr dvigatelda paydo bo'lgan nosozliklar, masalan, rotor yoki stator nosozliklari, izolyatsiya buzilishlari va elektromagnit maydon notekisliklarini aniqlash mumkin.
- Elektr dvigatellarida yuqori darajali garmonikalar ko'pincha nosozlik belgisi hisoblanadi.

### 3. **Amplituda va faza modulyatsiyasi:**

- Elektr dvigatel signalining vaqt bo'yicha amplituda yoki fazasi modulyatsiyalangan bo'lishi mumkin. Amplituda o'zgarishlari kuchlanish va tok signalidagi o'zgarishlarni ko'rsatishi mumkin, bu elektr yoki mexanik nosozliklar haqida ma'lumot beradi.
- Faza modulyatsiyasi esa dvigatel aylanishining anomaliyalarini, masalan, rotor nosozliklari yoki muvozanatsizlikni ko'rsatadi.

Takroriy neyron tarmoq (RNN) tenglamasini takomillashtirish bir necha usul orqali amalga oshirilishi mumkin. Standart RNN'lar ketma-ket ma'lumotlar bilan ishlashda sezilarli natijalar ko'rsatadi, ammo ba'zi cheklovlari mavjud, masalan, uzoq muddatli xotiralarni saqlashdagi qiyinchiliklar va ortga tarqatish (backpropagation through time) paytida gradientning so'nishi yoki portlashi kabi muammolar. Ushbu cheklovlarni yengish va ishlash samaradorligini oshirish uchun RNN'ning ba'zi ilg'or variantlarini ishlatish mumkin. Quyida RNN tenglamasini takomillashtirishning ba'zi usullari keltirilgan:

### 1. Uzoq qisqa muddatli xotira (LSTM)

LSTM (Long Short-Term Memory) oddiy RNN'lar bilan solishtirganda sezilarli yaxshilanishlarni ta'minlaydi, chunki u uzoq muddatli xotirani saqlash va gradient so'nishini oldini olish uchun mo'ljallangan. LSTM tarmog'i ma'lumotlarning har bir vaqt birligida qayerdan kelayotganini (yangi ma'lumotlarni saqlash yoki eski ma'lumotlarni unutish) nazorat qiluvchi darvozalar (gates) tizimiga ega.

LSTM tenglamalari:

LSTM hujayrasi quyidagi darvozalar yordamida ishlaydi:

• **Kirish darvozasi (input gate)** – yangi ma'lumotni xotiraga saqlashni nazorat qiladi:

$$i_t = \sigma(W_i \cdot x_t + U_i \cdot h_{t-1}) + b_i$$

**Chiqish darvozasi (output gate)** – xotira holatidan natijani chiqarish:

$$o_t = \sigma(W_o \cdot x_t + U_o \cdot h_{t-1}) + b_o$$

**Unutish darvozasi (forget gate)** – qanday ma'lumotlarni unutish kerakligini belgilaydi:

$$f_t = \sigma(W_f \cdot x_t + U_f \cdot h_{t-1}) + b_f$$

**Yangi hujayra holati (cell state)** – o'tgan hujayra holati va kirish signallari asosida yangilanadi:

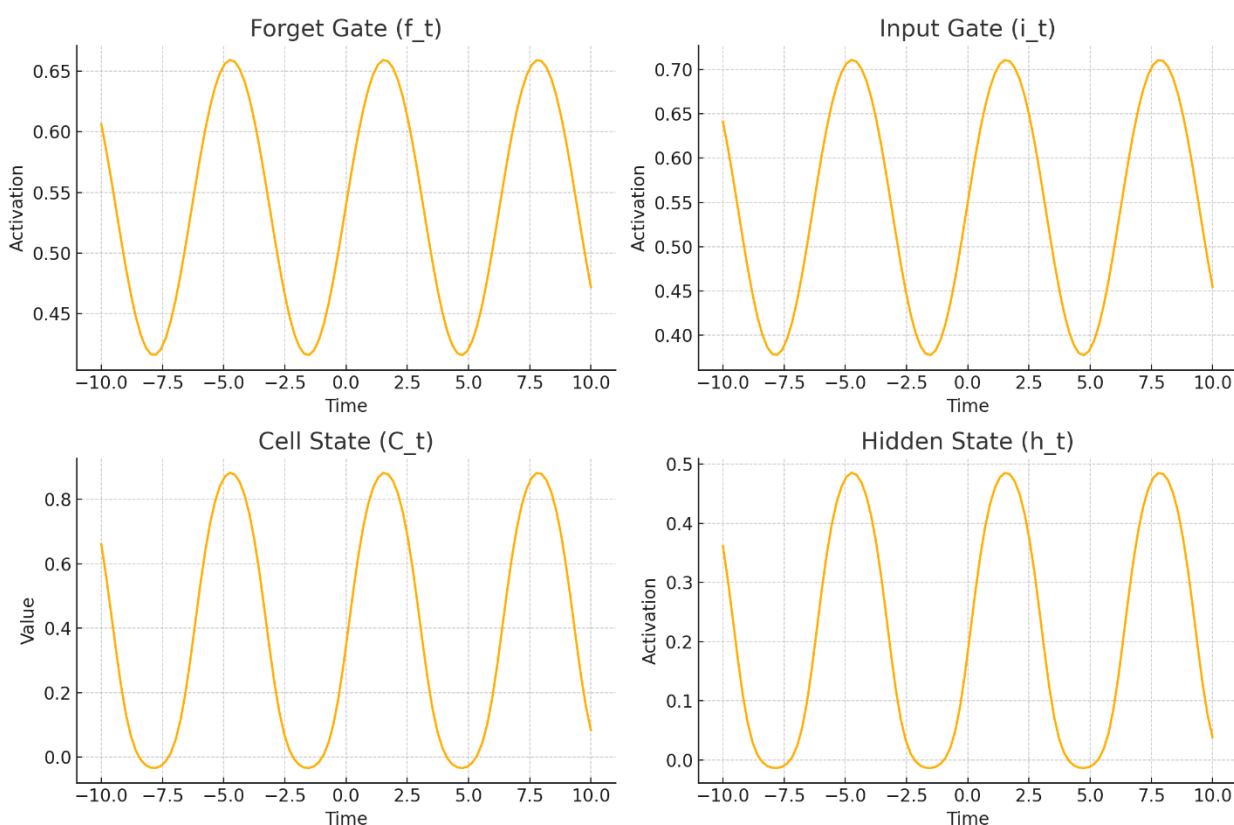
$$\check{c}_t = \tanh(W_c \cdot x_t + U_c \cdot h_{t-1}) + b_c$$

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \check{c}_t$$

### Yashirin holatni yangilash:

$$h_t = o_t \cdot \tanh(C_t)$$

LSTM yordamida uzoq muddatli va qisqa muddatli bog'lanishlar muammosini hal qilish mumkin. Xotiradagi muhim ma'lumotlarni unutmaslik va keraksiz ma'lumotlarni o'z vaqtida chiqarib tashlash darvozalar yordamida amalga oshiriladi.



Yuqoridagi grafiklar LSTM tarmoqlarining asosiy komponentlarini ko'proq:

1. **Forget Gate ( $f_t$ )** : Xotiraning qaysi qismlarini unutar yoki boshqarish boshqaruvchi darvozaning aktivatsiya narxlari.
2. **Input Gate ( $i_t$ )** : Xotiraga qaysi ma'lumotlarni aniqlash aniqlovchi darvozaning aktivatsiya qiymati.

3. **Hujayra holati ( $C_t$ )** : Hujayra holati, bu LSTM xotirasi bo'lib, vaqt bo'yicha qanday o'zini ko'rsatadi.
4. **Hidden State ( $h_t$ )** : Yashirin holat, bu chiqish darvozasining aktivatsiyasi bilan birga vaqt bo'yicha qanday o'chirishni ifodalaydi.

Ush grafiklar LSTM'ining har bir qismi qanday ishlashini vizual ko'rinishda aks ettiradi.

1. **Ma'lumotlarni normallashtirish**: MinMaxScaler yordamida barcha ma'lumotlar 0 va 1 oralig'ida normallashtiriladi, bu esa tarmoqni tezroq o'qitishga yordam beradi.
2. **LSTM modeli**:
  - **Bir nechta LSTM qavatlari**: Modelga ikkita LSTM qatlami kiritilgan bo'lib, ularning har biri 50 ta birlikdan iborat. Bu qavatlar vaqt bo'yicha signalni qayta ishlashda yordam beradi.
  - **Dropout qavati**: Modelda overfittingdan saqlanish uchun 20% dropout qo'llangan.
  - **Chiqish qavati**: Dense qavati orqali bitta sonli natija chiqariladi (masalan, harorat).
3. **Modelni o'qitish**: fit funksiyasi yordamida modelni ma'lumotlar bo'yicha o'qitamiz.
4. **Bashorat qilish**: O'rgatilgan model yordamida ma'lumotlarni bashorat qilamiz va natijalarni real va bashorat qilingan qiymatlar sifatida grafiga solishtiramiz.

Shunday qilib, LSTM modelini yaratib, elektr dvigatel parametrlari asosida prognoz qilishni amalga oshirish mumkin.

**Gated Recurrent Unit (GRU)** — bu LSTM (Long Short-Term Memory) tarmog'iga o'xshash, lekin o'zining soddalashtirilgan va samaradorroq versiyasi hisoblanadi. GRU'da LSTM singari xotira va darvozalar mavjud, lekin ular bir necha jihatdan soddalashtirilgan. GRU vaqtli ketma-ketliklarni tahlil qilishda va uzoq

muddatli bog'lanishlarni o'rganishda samarali ishlaydi, lekin uni yanada takomillashtirish mumkin.

GRU'ni takomillashtirish uchun bir nechta usullarni qo'llash mumkin:

## 1. Qo'shimcha Darvoza Qo'shish

GRU ikkita asosiy darvozaga ega: **yangilash darvozasi** va **reset (qayta o'rnatish) darvozasi**. Ushbu darvozalar ketma-ketlikdagi o'zgarishlarni boshqaradi, lekin **darvozalarni ko'paytirish** orqali GRU'ni yanada kuchaytirish mumkin.

- **Update (yangilash) darvozasi:** Ushbu darvoza xotiraga qanday yangiliklar kiritilishini boshqaradi.
- **Reset (qayta o'rnatish) darvozasi:** Bu darvoza avvalgi yashirin holatning necha qismi hozirgi ma'lumot bilan bog'lanishi kerakligini belgilaydi.

Qo'shimcha ravishda:

- **Uzluksiz unutish darvozasi:** Ba'zi ketma-ket ma'lumotlar vaqt o'tishi bilan yangilanishi kerak bo'lsa-da, boshqa qismlariga tez-tez e'tibor berishning hojati yo'q. Ushbu darvoza GRU ga qaysi vaqt nuqtalarini unutish va qaysi ma'lumotlarni uzoq muddatli xotirada saqlashni nazorat qilish imkonini beradi.

## 2. Attention Mexanizmini Integratsiya Qilish

Attention mexanizmi GRU bilan birga ishlatilganda vaqtli ketma-ketliklarda muhim qismlarga e'tibor berish imkonini beradi. Bu, ayniqsa, uzoq muddatli bog'lanishlarni yaxshilashda va uzoq vaqt bo'yicha signallarda aniqroq xulosa chiqarishda foydali.

- **Attention mexanizmi bilan takomillashtirish:** Har bir vaqt nuqtasiga diqqat qaratish orqali GRU modelini yanada kuchli qiladi. Bu usul uzoq ketma-ketliklarni, masalan, nutq tanib olish yoki tabiiy tilni qayta ishlash vazifalarida yanada yaxshi ishlaydi.

## Attention tenglamalari:

- **O'xshashlik bahosi:**

$$e_{t,t'} = Q_t K_{t'}$$

bu yerda  $Q_t$  – so'rov (query),  $K_{t'}$  – kalit (key) vektorlari.

- **Vaznlarni normallashtirish:**

$$a_{t,t'} = \frac{\exp(e_{t,t'})}{\sum_{t'} \exp(e_{t,t'})}$$

- **Yig'ilgan e'tiborli yashirin holat:**

$$C_t = \sum_{t'} a_{t,t'} V_{t'}$$

Attention mexanizmi bilan birlashtirilgan GRU ketma-ketlikdagi muhim vaqt nuqtalariga e'tibor qaratadi va o'zgaruvchilar orasidagi bog'lanishni yaxshilaydi.

Standart GRU faqat bitta yo'nalishda, ya'ni o'tmishdan kelajakka qarab ishlaydi. **Ikki yo'nalishli GRU** esa ketma-ketliklarni ikki yo'nalishda, o'tmishdan kelajakka va kelajakdan o'tmishga qarab tahlil qiladi. Bu, ayniqsa, kelajakdagi ma'lumotlar tahlilni yaxshilashi mumkin bo'lgan vaziyatlarda foydali bo'ladi.

- **Forward GRU:** O'tmishdan kelajakka qarab ma'lumotlarni o'rganadi.
- **Backward GRU:** Kelajakdan o'tmishga qarab ma'lumotlarni tahlil qiladi.

## Ikki yo'nalishli GRU tenglamasi:

- Oldinga yo'nalishdagi yashirin holat:

$$\vec{h}_t = f(W \cdot x_t + U \cdot \vec{h}_{t-1})$$

- Orqaga yo'nalishdagi yashirin holat:

$$\overleftarrow{h}_t = f(W \cdot x_t + U \cdot \overleftarrow{h}_{t-1})$$



Final yashirin holat oldinga va orqaga yo'nalishdagi yashirin holatlarning kombinatsiyasidan hosil bo'ladi:

$$h_t = [\overrightarrow{h}_t, \overleftarrow{h}_t]$$

Bu takomillashuv GRU ga vaqt ketma-ketligidagi har bir nuqtani aniqroq tahlil qilishga yordam beradi.

#### 4. Layer Normalization

GRU uchun **qatlam normalizatsiyasi** (Layer Normalization) qo'shilishi modelning konvergentsiyasini tezlashtiradi va o'qitishning barqarorligini oshiradi. Layer normalizatsiyasi har bir qatlamda ma'lumotlarning statistik jihatdan normalizatsiyasini ta'minlaydi, bu esa o'qitish davomida aktivatsiyalarning barqarorligini ta'minlaydi.

- **Layer Normalization formulasi**

$$h_t = \frac{h_t - \mu}{\sigma}$$

- bu yerda  $\mu$  o'rtacha qiymat,  $\sigma$  dispersiya.

Layer normalization gradientlarning kuchli o'zgarishini kamaytirib, o'qitishni yaxshilaydi.

#### 5. Adaptive Learning Rates (Moslashuvchan o'rganish tezligi)

GRU modelini o'qitishda **moslashuvchan o'rganish tezligi** ishlatilsa, modelning konvergentsiya tezligini oshirish va overfittingni kamaytirish mumkin. Optimallashtiruvchi algoritmlar, masalan, **Adam** yoki **RMSProp**, o'rganish tezligini moslashtiradi va GRU ning yanada tez va barqaror o'qitilishini ta'minlaydi.

**Batch normalization** va **GRU** (Gated Recurrent Unit) ni birlashtirish neyron tarmoqlarni yanada barqaror va samarali o'qitish uchun foydali usul hisoblanadi. **Batch normalization** GRU tarmog'ida qatlamlar orasida ma'lumotlarning o'rtacha qiymati va

dispersiyasini o'zgartirish orqali tarmoqni yanada barqarorlashtiradi. Bu esa modelning o'qitish vaqtida konvergentsiyasini tezlashtiradi va overfittingni kamaytirishga yordam beradi.

## Batch Normalization nima?

Batch normalization qatlamlarning chiqishlarini normalizatsiya qilish orqali modelni o'qitish jarayonida qatlamlar orasida statistik o'zgarishlarni kamaytiradi. Bu har bir mini-batch uchun neyron tarmog'i qatlamlari tomonidan yaratilgan aktivatsiyalarni normalizatsiya qiladi va **o'rtacha qiymat** va **dispersiyani** saqlab turadi.

## GRU va Batch Normalizationni qo'llash

GRU modelida **Batch Normalizationni** o'zining asosiy qismlariga qo'shish mumkin, bu o'rganish jarayonida aktivatsiyalarni barqarorlashtiradi va har bir qatlamda o'zgaruvchanlikni kamaytiradi. Batch normalizationni quyidagi joylarda qo'llash mumkin:

- **Kirish ma'lumotlari:** GRU ga kiritilgan ma'lumotlarni normallashtirish.
- **Yashirin holatlar:** GRU'dagi har bir yashirin holatni normalizatsiya qilish.

## 2. Batch Normalizationning harakat tamoyillari

Batch normalization har bir qatlam uchun quyidagilarni bajaradi:

- Har bir mini-batch uchun o'rtacha qiymat va dispersiyani hisoblab chiqadi.
- Ma'lumotlarni ushbu qiymatlarga qarab normallashtiradi:

$$\tilde{x} = \frac{x - \mu_{batch}}{\sqrt{\sigma_{batch}^2 + \epsilon}}$$

bu yerda  $\mu_{batch}$ -batch o'rtacha qiymati,  $\sigma_{batch}^2$  mini-batch dispersiyasi, va  $\epsilon$  kichik qiymat bo'lib, nolga bo'linishni oldini oladi.

Shundan so'ng, normalizatsiya qilingan qiymatlar masshtablanadi va siljiydi:

$$y = \gamma \tilde{x} + \beta$$

bu yerda  $\gamma$  masshtablash parametri,  $\beta$  esa siljish parametri bo'lib, ular o'rganiladi.

### 3. GRUda Batch Normalization qanday yordam beradi?

- **Stabilizatsiya:** Batch normalization GRU'ning yashirin holatlarida kutilmagan o'zgarishlarni kamaytiradi. Bu modelning konvergentsiyasini yaxshilaydi, chunki aktivatsiyalarni bir xil diapazonda ushlab turadi.

- **Tezroq o'qitish:** Normalizatsiya tufayli GRU qatlamlari tezroq va samarali o'qitiladi. Odatda, o'qitish jarayonida o'rganish tezligi (learning rate) katta qiymatda bo'lishi mumkin, bu modelni tezroq o'qitishga yordam beradi.

- **Overfittingni kamaytirish:** Normalizatsiya qilinganda tarmoqning ma'lum bir aktivatsiyaga haddan tashqari moslashishi kamayadi, bu esa overfitting ehtimolini pasaytiradi.

### 4. Layer normalization bilan farqi

**Layer Normalization** ham aktivatsiyalarni normalizatsiya qiladi, lekin u ma'lumotlarni har bir qatlam bo'yicha, ya'ni qatlamning barcha neyronlari uchun umumiy o'rtacha va dispersiya bilan normalizatsiya qiladi, bu esa ketma-ketliklarda, ayniqsa, GRU va RNN lar uchun foydali bo'lishi mumkin.

**Batch normalization** neyron tarmoqlarni o'qitish jarayonida sezilarli ta'sir ko'rsatadigan samarali texnika bo'lib, tarmoqning ishlashiga turli jihatlardan ijobiy ta'sir qiladi. Quyida batch normalizationning modelga qanday ta'sir qilishini batafsil tushuntirib beraman:

#### 1. O'qitish barqarorligini yaxshilaydi

Batch normalization modelda har bir qatlamning chiqishlarini normalizatsiya qiladi, ya'ni har bir qatlamning o'rtacha qiymatini 0 ga va tarqalishini (dispersiyani) 1 ga yaqinlashtiradi. Bu tarmoqning barcha qatlamlarida aktivatsiyalarni muvozanatda ushlab turadi.

## Natijasi:

- **O'qitish jarayonida o'zgarishlar barqarorlashadi:** O'qitish jarayonida qatlamlar orasida ma'lumotlarning statistik o'zgarishlari (kiritish ma'lumotlarining kengayishi yoki qisqarishi) muammosi kamayadi, bu esa **gradientlarning barqarorligini** oshiradi.

- **Tarmoqning konvergeniyasi** (optimal yechimga yetib borish jarayoni) tezlashadi, chunki gradientlar noto'g'ri tarqalib ketmaydi.

## 2. Batch Normalizationning O'qitishdagi Ta'siri

### a) Tarmoqning konvergeniyasini tezlashtiradi

Batch normalization, o'qitish davomida aktivatsiyalarni o'rtacha qiymatga yaqinlashtirishi va ular orasidagi farqlarni muvozanatlashtirishi tufayli **gradient so'nishi** va **gradient portlashi** muammolarini kamaytiradi. Bu, ayniqsa, chuqur tarmoqlar (deep networks) uchun muhim, chunki chuqur tarmoqlarda gradientlar qatlamlar bo'yicha oldinga va orqaga tarqaladi va aktivatsiyalarning kengayishi yoki qisqarishi tufayli qiyinchiliklar paydo bo'lishi mumkin.

### **Gradient so'nishi (vanishing gradient) muammosi:**

Agar aktivatsiyalar juda kichik bo'lsa, gradientlar juda sekin o'zgaradi va tarmoq o'qitilishi sustlashadi. Batch normalization ushbu kichik qiymatlar orasidagi farqlarni izchil qiladi va gradientlarning barqarorligini oshiradi.

### **Gradient portlashi (exploding gradient) muammosi:**

Agar aktivatsiyalar juda katta qiymatlarga ega bo'lsa, gradientlar juda katta bo'lib ketadi va bu tarmoqning o'rganishini beqaror qiladi. Batch normalization bu muammoni ham hal qilishga yordam beradi, chunki aktivatsiyalar izchil qiymatlar atrofida joylashadi.

## b) Tezroq o'rganish imkonini beradi

Batch normalization yordamida tarmoqdagi o'zgarishlar izchil bo'lgani uchun, o'rganish jarayonida **katta o'rganish tezligi (learning rate)** bilan o'qitish imkoniyati paydo bo'ladi. O'rganish tezligini oshirish esa modelning tezroq konvergentsiyasiga olib keladi, ya'ni modelni ma'lumotlarga tezroq moslashtirish mumkin bo'ladi.

Matematik nuqtai nazardan, gradientning hisoblash jarayoni barqarorlashtirilgani uchun gradientlar juda katta yoki juda kichik qiymatlarga ega bo'lmaydi. Natijada, yuqori o'rganish tezligi bilan ham tarmoqni barqaror o'qitish mumkin bo'ladi.

## 3. Overfittingni kamaytirish

Batch normalization, o'qitish jarayonida **overfitting** muammosini kamaytirishga yordam beradi. Chunki aktivatsiyalar izchil qiymatlarga ega bo'lgani tufayli, tarmoq ma'lumotlar shovqiniga haddan tashqari moslashib qolmaydi. Shunday qilib, tarmoq **umumiylashuv qobiliyatini** yaxshiroq saqlab qoladi va yangi ma'lumotlarga nisbatan ham yaxshi natijalar beradi.

Matematik jihatdan, batch normalizationning yuqori darajadagi normallashtirish jarayoni ba'zi ma'lumotlar shovqiniga (noise) moslashib qolishdan saqlaydi va bu tarmoqning test ma'lumotlarida yaxshi ishlashiga olib keladi.

## 4. Batch Normalizationning Foydasi va Kamchiliklari

### Foydalar:

- **Tezroq o'qitish:** Aktivatsiyalarning normalizatsiyasi gradientlar o'zgarishini barqaror qiladi va yuqori o'rganish tezligi bilan modelni o'qitishga imkon beradi.

- **Gradient so'nishi va portlashining oldini olish:** Aktivatsiyalarni barqarorlashtirish orqali tarmoqni chuqur qatlamlarda ham samarali o'qitish imkonini beradi.

- **Umumiy lashuvni yaxshilash:** Overfittingni kamaytirib, modelning umumiy lashuv qobiliyatini oshiradi.

### **Kamchiliklar:**

- **Mini-batch hajmiga bog'liqlik:** Batch normalization samarali bo'lishi uchun mini-batch hajmi katta bo'lishi kerak. Kichik mini-batchlar bilan tarmoq barqaror bo'lmasligi mumkin.

- **Inference paytidagi qiyinchiliklar:** Test vaqtida (inference) batch normalizationni test qilish qiyinroq bo'lishi mumkin, chunki u butun ma'lumotlar to'plamiga emas, faqat o'rtacha qiymat va dispersiya o'lchovlariga asoslanadi.

### **Xulosa**

Elektr dvigatellarining texnik holatini nazorat qilish va diagnostika qilish zamonaviy sanoat ishlab chiqarish tizimlarida muhim ahamiyatga ega. Ushbu tadqiqotda Proniy modellaridan foydalanish orqali signal komponentlarini tahlil qilish va takroriy neyron tarmoqlarni integratsiya qilish natijasida diagnostika jarayonlarini optimallashtirish yo'llari ko'rib chiqildi. Proniy modellari elektr dvigatellarning murakkab signallarini matematik modellashtirishga imkon berib, nosozliklarni erta bosqichda aniqlashga yordam beradi. Takroriy neyron tarmoqlar esa vaqt bo'yicha ketma-ket ma'lumotlarni tahlil qilishni takomillashtirib, diagnostika jarayonining samaradorligini oshiradi. Ushbu yondashuvlar elektr dvigatellarning ishonchliligini ta'minlash, texnik xizmat ko'rsatishning iqtisodiy samaradorligini oshirish va nosozliklarni o'z vaqtida aniqlash uchun katta imkoniyatlar yaratadi. Tadqiqot natijalari elektr dvigatellar diagnostikasini yangi bosqichga olib chiqishda muhim qadam hisoblanadi.

### **Foydalanilgan adabiyotlar**

1. Bayindir, R., & Cetinceviz, Y. (2011). A water pumping control system with a programmable logic controller (PLC) and industrial wireless modules for industrial plants—An experimental study. *Energy Conversion and Management*, 52(1), 58–65.

2. Sayed, K. A., & Hashem, M. S. (2014). Fault diagnosis of induction motor drive using wavelet and neural network. *International Journal of Electrical Power & Energy Systems*, 61, 1–11.
3. Назаров, Р. Х., & Турсунов, С. К. (2018). Электродвигателей техническое обслуживание и диагностика. Ташкент: УЗНИТИ.
4. Гаврилов, А. И., & Литвинов, А. В. (2015). Применение Проней моделей в диагностике электродвигателей. *Электротехнический журнал*, 3(10), 56–62.
5. Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, 61, 85–117. <https://doi.org/10.1016/j.neunet.2014.09.003>
6. Прокофьев, С. И., & Ковалёв, Ю. В. (2012). Прогнозирование отказов электродвигателей с использованием нейронных сетей. *Автоматика и телемеханика*, 7(1), 12–18.
7. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.
8. Smith, G. B., & Brown, R. D. (2019). Real-time monitoring of electrical motors using advanced diagnostics. *Journal of Industrial Technology*, 35(2), 45–53.
9. Yusuf, A., & Ayodeji, S. (2020). Optimization of predictive maintenance using recurrent neural networks. *Applied Artificial Intelligence*, 34(8), 1–17.
10. Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder–decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.
11. Рузиматов, С., & Тухтасинов, Д. (2021). Выбор цифровых устройств для регулирования содержания влаги хлопка-сырца. *Central Asian Journal of Theoretical and Applied Science*, 2(9), 10-14.
12. Тухтасинов Д.Х., & Исманов М.А. (2018). СОВЕРШЕНСТВОВАНИЕ СИСТЕМЫ УПРАВЛЕНИЯ КОЛОННОЙ СИНТЕЗА АММИАКА НА ОСНОВЕ НЕЧЕТКОЙ ЛОГИКИ. *Экономика и социум*, (12 (55)), 1236-1239.

