



FAKULTA APLIKOVANÝCH VĚD
ZÁPADOČESKÉ UNIVERZITY
V PLZNI

KATEDRA
KYBERNETIKY

BAKALÁŘSKÁ PRÁCE

Automatické ovládání žaluzií s využitím strojového učení

Autor:
Vojtěch Breník

Vedoucí práce:
Ing. Martin Bulín, M.Sc.

19. května 2022

Prohlášení

Předkládám tímto k posouzení a obhajobě bakalářskou práci zpracovanou na závěr studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni.

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím odborné literatury a pramenů, jejichž úplný seznam je její součástí.

V Plzni dne 23. května 2022

ZÁPADOČESKÁ UNIVERZITA

Fakulta aplikovaných věd

Katedra kybernetiky

Abstrakt

Bakalářská práce

Automatické ovládání žaluzií s využitím strojového učení

Vojtěch Breník

Your abstract goes here...

Poděkování

Your acknowledgements go here...

Obsah

Abstrakt	ii
1 Úvod	1
1.1 Umělé neuronové sítě (NS)	2
1.1.1 Umělý neuron	2
1.1.2 Aktivační funkce	3
Aktivační funkce Rectified Linear Unit (ReLU)	3
Aktivační funkce Swish	3
Aktivační funkce softplus	4
Sigmoida jako aktivační funkce	4
Heavisidova funkce jako aktivační funkce	4
Hyperbolický tangens jako aktivační funkce	4
Aktivační funkce softsign	5
1.1.3 Dopředná neuronová síť	5
1.1.4 Rekurentní neuronová síť, LSTM	5
1.1.5 Použití neuronových sítí na problémy v reálném čase	7
1.2 Internet věcí (IoT)	7
1.3 Venkovní žaluzie s pohonem	7
2 Definice úlohy	8
2.1 Vstupy systému	8
2.2 Výstupy systému	10
3 Hardwarové komponenty	11
3.1 Zařízení pro měření teploty vzduchu a intenzity osvětlení	11
3.1.1 Zapojení	12
3.1.2 Deska plošných spojů (DPS)	13
3.1.3 Modul ESP-12E s MCU a WiFi	14
3.1.4 Teplotní senzor DS18B20	14
3.1.5 Senzor intenzity osvětlení TSL2591	15
4 Sběr dat a komunikace komponent	16
4.1 Použití komunikačního protokolu MQTT	17
4.2 Komunikace s pohonem žaluzie	17
4.3 Využití služby OpenWeather	19
4.4 Databáze	19
4.5 Webový server pro komunikaci s GUI	20
5 Automatické ovládání žaluzií	22
5.1 Příprava dat pro strojové učení	22
5.2 Regresor založený na pravidlech (If-else)	23
5.3 Dopředná neuronová síť jako regresor (FFNN)	25

5.4	Rekurentní neuronová síť jako regresor (LSTM)	25
5.5	Přetrénování neuronových sítí (retraining)	25
6	Grafické uživatelské rozhraní	27
6.1	Stránka Data	27
6.2	Stránka Simulator	28
6.3	Stránka Control	29
6.4	Stránka Live	30
7	Vyhodnocení	32
7.1	Vyhodnocení sběru dat a spolehlivosti	32
7.2	Volba struktury NS a parametrů učení	33
7.3	Vliv jednotlivých příznaků na predikci	34
7.4	Porovnání regresorů	38
7.5	Vyhodnocení vlivu retrainingu	39
8	Diskuze	41
8.1	Možnosti reálného nasazení systému	42
9	Závěr	43
	Bibliografie	44

Seznam obrázků

1.1	Grafy aktivačních funkcí	3
1.2	Dopředná neuronová síť	5
1.3	LSTM jednotka	6
3.1	Skutečné umístění měřicího zařízení	12
3.2	Schéma zapojení měřicích zařízení	13
3.3	Návrh DPS měřicích zařízení	14
4.1	Schéma komunikace komponent	16
4.2	Schéma endpointů a přenášených struktur	21
6.1	Ovládací panel v záhlaví GUI	27
6.2	Stránka Data v GUI	28
6.3	Stránka Simulator v GUI	28
6.4	Vizualizace výstupu regresoru	29
6.5	Stránka Control v GUI	29
6.6	Stránka Live v GUI	30
7.1	Graf sesbíraných dat (příklad)	32
7.2	Graf druhu sesbíraných vzorků a výpadků	33
7.3	Důležitost příznaků FFNN	37
7.4	Důležitost příznaků LSTM	37
7.5	Porovnání predikce regresorů	38
7.6	Porovnání predikce FFNN (retraining)	40
7.7	Porovnání predikce LSTM (retraining)	40

Seznam tabulek

2.1	Příznaky pro automatické ovládání žaluzie	10
4.1	Příznaky a jejich MQTT témata	17
5.1	Prohledávané parametry dopředné neuronové sítě	25
5.2	Prohledávané parametry rekurentní neuronové sítě	26
7.1	Výpadky	33
7.2	Výsledky prohledávání parametrů FFNN	35
7.3	Výsledky prohledávání parametrů LSTM	36
7.4	Porovnání regresorů	38
7.5	Retraining FFNN	39
7.6	Retraining LSTM	39

Seznam zkratek

API Application Programming Interface (rozhraní pro programování aplikací). 1, 17, 32, 41

DPS Deska plošných spojů. iv, vi, 11–13

FFNN Feedforward Neural Network (dopředná neuronová síť). 26

GUI Graphical User Interface (grafické uživatelské rozhraní). iv, 11, 18–20, 26, 27, 29, 30

IoT Internet of Things (internet věcí). iv, 7

LSTM Long Short-Term Memory (druh rekurentní neuronové sítě). 6, 7, 25, 26

MCU Microcontroller Unit (jednočipový počítač). iv, 12–14, 41

MSE Mean Squared Error (střední kvadratická odchylka). 10, 22, 25, 26, 33, 34, 38, 39, 42

NS Neuronová síť. iv, v, 2, 8, 20, 22, 23, 25, 26, 33, 34, 41–43

ReLU Aktivační funkce Rectified Linear Unit. iv, 3, 4

URL Uniform Resource Locator. 18

1 Úvod



Stínění oken v pozemních stavbách určených k bydlení se využívá k regulaci teploty a zajištění soukromí obyvatel. (Lubinová, 2013) Jednou z možností stínění jsou venkovní žaluzie osazené pohonem na dálkové ovládání. Jejich uživatel může měnit výšku vytažení žaluzie a sklon lamel a ovlivňovat tím množství záření, které skrz okno prochází. Pomocí vzdáleného ovládání pak lze žaluzie řídit automaticky.

Komplexní systém může generovat akční zásahy pro řízení žaluzií na základě různých vstupních informací (datum a čas, měření veličin pomocí senzorů) bez účasti uživatele za účelem zajištění jeho komfortu a úspory prostředků vynaložených na regulaci teploty v interiéru (vytápění a chlazení). Sběr informací na základě kterých se má rozhodovat o nastavení žaluzie i samotné rozhodování musí probíhat v reálném čase.



Tato práce se zabývá návrhem takového systému, návrhem měřicích zařízení použitelných pro měření veličin užitečných při rozhodování systému a možnostmi využití strojového učení k ovládání žaluzií v reálném čase. Zkoumá časový vývoj použitých algoritmů při opakovaném učení na postupně sbíraných datech s ohledem na měřící se požadavky uživatele a měnící se podmínky v průběhu roku. Její cíle jsou následující:

1. Určit vhodné vstupní veličiny (příznaky) a výstupní veličiny pro automatické rozhodování o stavu žaluzie.
2. Zkonstruovat měřicí zařízení pro měřitelné příznaky a odesílat jejich hodnoty pomocí MQTT.
3. Zajistit komunikaci s vybranými žaluziemi pomocí poskytovaného API² a MQTT¹.
4. Zajistit komunikaci se zdroji všech příznaků pomocí MQTT a data ukládat pro pozdější použití.
5. Navrhnout základní rozhodovací systém založený na pravidlech (baseline).
6. Navrhnout model neuronové sítě a porovnat jeho použitelnost s baseline systémem.



V práci se využívá skutečná venkovní žaluzie s hliníkovými lamelami a pohonem od firmy Somfy. Vzdálenou komunikaci s ním zajišťuje řídicí jednotka Tahoma připojená k internetu, která komunikuje se serverem výrobce, jejichž prostřednictvím je možné žaluzii ovládat a zjistit její aktuální stav. Žaluzie se nachází před oknem pokoje autora této práce v rodinném domě v obci

¹MQTT je komunikační protokol blíže popsáný v sekci 4.1

²Application Programming Interface (rozhraní pro programování aplikací)

vzdálené asi 12 km severovýchodně od Plzně. Uvnitř tohoto pokoje a na střeše daného domu také probíhají všechna měření.

1.1 Umělé neuronové sítě (NS)

Umělé neuronové sítě jsou stěžejní součástí navrhovaného systému, ve kterém se používají pro odhad správného nastavení žaluzie. Skládají se ze vzájemně propojených umělých neuronů obvykle uspořádaných do vrstev. Samotné neurony se považují za nejjednodušší neuronové sítě. Každý z neuronů má množinu parametřů, které se pomocí optimalizačních algoritmů nastavují tak, aby podle zvoleného kritéria výstup neuronové sítě co nejlépe odpovídal předloženým trénovacím datům.

1.1.1 Umělý neuron

Bishop ve své knize (1995) shrnuje vývoj od matematického modelu neuronu McCullocha a Pittse (1943) až po perceptron (Rosenblatt, 1958), který je v některých neuronových sítích využíván dodnes. V obou zmíněných případech se jedná o matematickou funkci, která zobrazuje vstupy x_1, x_2, \dots, x_N , kde N je počet vstupů neuronu, na výstup y .

McCullochův a Pittsův model neuronu pracuje v diskrétních úrovních vstupů a výstupů: $x_i, y \in \{0, 1\}$. Vstupy x_i jsou postupně sečteny. Přesáhne-li součet nastavený práh Θ , neuron je aktivován a jeho výstup $y = 1$. V opačném případě je $y = 0$. Výstup takového neuronu tedy můžeme vyjádřit jako

$$y = g\left(\sum_{i=1}^N x_i\right),$$

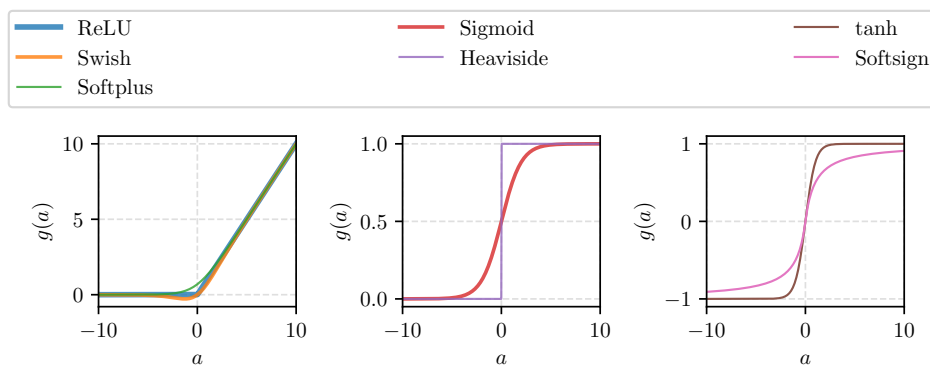
kde $g(a)$ je v tomto případě Heavisidova funkce (sekce 1.1.2). Funkce $g(a)$ se obecně nazývá *aktivační* a v jiných modelech neuronu se využívají různé i jinak definované funkce (Bishop, 1995; příklady jsou uvedené v sekci 1.1.2).

Perceptron (Rosenblatt, 1958) je v určitém slova smyslu zobecněním McCullochova a Pittsova modelu neuronu. Jako vstupy x_i uvažuje hodnoty z \mathbb{R} a kromě prahu Θ má navíc váhy vstupů $w_i \in \mathbb{R}$. Těmi jsou postupně vynásobeny příslušné vstupy x_i a výsledky jsou sečteny. Přesáhne-li tento součet nastavený práh Θ , perceptron je aktivován a jeho výstup $y = 1$. V opačném případě je $y = 0$. Označíme $\mathbf{x} = [x_0, x_1, x_2, \dots, x_N]^T$ a $\mathbf{w} = [w_0, w_1, w_2, \dots, w_N]^T$, kde $x_0 = 1$ je přidáný myšlený vstup, který nabývá vždy hodnoty 1, a $w_0 = \Theta$ je jeho odpovídající váha. Součet se tím zvýší o práh Θ a pro výstup y perceptronu pak platí:

$$y = g(\mathbf{w}^T \mathbf{x}),$$

kde $g(a)$ je Heavisidova funkce. Parametr w_0 se nazývá *bias* a reprezentuje práh Θ pro aktivaci perceptronu. (Bishop, 1995)

Rosenblatt také navrhl algoritmus učení perceptronu s učitelem pro klasifikaci, které postupnou úpravou parametrů zajistil u lineárně separabilních množin správnou klasifikaci trénovacích dat. (Rosenblatt, 1958)



OBRÁZEK 1.1: Grafy vybraných aktivačních funkcí

Pokud se jako funkce $g(a)$ místo Heavisidovy funkce použije jiná aktivační funkce (sekce 1.1.2), která je diferencovatelná vzhledem k vektoru vah a biasu \mathbf{w} , lze pro učení perceptronu použít algoritmus *gradient descent*. (Bishop, 1995)

1.1.2 Aktivační funkce

Aktivační funkce se v neuronových sítích využívají k vyhodnocení výstupu (aktivace) y na základě váženého součtu vstupů (včetně biasu). Podle svého průběhu mohou mít využití v různých úlohách a mají také vliv na průběh učení. (Nwankpa et al., 2018) Níže jsou uvedené některé aktivační funkce, jejich definice a vlastnosti. Jejich průběhy $g(a)$ pro $a \in \langle -10, 10 \rangle$ jsou v grafech na obrázku 1.1.

Aktivační funkce Rectified Linear Unit (ReLU)

Funkce ReLU je široce používána v aplikacích hlubokého učení a poskytuje výsledky na špičkové úrovni. Ve srovnání se sigmoidou a hyperbolickým tangensem dává v hlubokém učení lepší výsledky a zajišťuje větší míru zobecnění. (Nwankpa et al., 2018) Definována je takto:

$$g(a) = \max(0, a) = \begin{cases} a, & \text{jestliže } a \geq 0 \\ 0, & \text{jestliže } a < 0. \end{cases} \quad (\text{Nair a Hinton, 2010})$$

ReLU se obvykle používá ve skrytých vrstvách neuronových sítí a je doplněna jinou aktivační funkcí ve výstupní vrstvě. Velkou výhodou je malá výpočetní náročnost výpočtu funkční hodnoty, což má pozitivní vliv na rychlost trénování. (Nwankpa et al., 2018)

Aktivační funkce Swish

Funkce Swish v některých ohledech překonává ReLU. Průběhem se výrazněji liší jen v okolí 0, kde má ostré lokální minimum. Narozdíl od ReLU je hladká a nemonotónní. Nemonotónnost ji také odlišuje od většiny aktivačních funkcí. Stejně jako ReLU je omezená zdola a neomezená shora. Je definovaná jako:

$$g(a) = a \cdot \sigma(\beta a),$$

kde $\sigma(z) = \frac{1}{1+e^{-z}}$ je sigmoida a β je volitelný nebo trénovatelný parametr. (Ramachandran, Zoph a Le, 2017)

Ramachandran, Zoph a Le ukázali, že na naprosté většině zkoušených úloh je alespoň tak dobrá jako původní aktivační funkce (ve většině případů dokonce lepší). Její podobnost s ReLU umožňuje snadné nahrazení v již existujících modelech

Aktivační funkce softplus

Funkce softplus je hladkou aproximací ReLU s nenulovými gradienty. Je definována jako:

$$g(a) = \ln(1 + e^a). \quad (\text{Dugas et al., 2000})$$

Je hladká, ostře monotónní a zdola omezená. Při jejím použití modely konvergují rychleji než s využitím ReLU a sigmoidy.

Sigmoida jako aktivační funkce



Sigmoida je speciálním případem logistické funkce, jedná se o hladkou nelineární aktivační funkci, omezenou shora i zdola. Její definice je:

$$g(a) = \sigma(a) = \frac{1}{1 + e^{-a}}. \quad (\text{Nwankpa et al., 2018})$$

Využívá se ve výstupních vrstvách neuronových sítí, mimo jiné v případě výstupů založených na pravděpodobnosti. (Nwankpa et al., 2018)

Heavisidova funkce jako aktivační funkce

Heavisidova funkce se využívala v původních modelech neuronu pro svou podobnost s biologickou předlohou. (Bishop, 1995; McCulloch a Pitts, 1943) Je definována takto:

$$g(a) = \begin{cases} 0 & \text{jestliže } a < 0; \\ 1 & \text{jestliže } a \geq 0. \end{cases}$$

V současných aplikacích, které se trénují pomocí gradientních metod ji ale nelze použít, protože není diferencovatelná. Je limitním případem logistické funkce, je tedy omezená. Její obor hodnot je dvouprvková množina $\{0, 1\}$.

Hyperbolický tangens jako aktivační funkce

Hyperbolický tangens je hladká, monotónní, omezená funkce definovaná jako:

$$g(a) = \frac{e^a - e^{-a}}{e^a + e^{-a}}. \quad (\text{Nwankpa et al., 2018})$$

Její střední hodnota leží v 0. Využívá se v některých rekurentních neuronových sítích v oblastech jako zpracování přirozeného jazyka a rozpoznávání řeči. (Nwankpa et al., 2018)

Aktivační funkce softsign

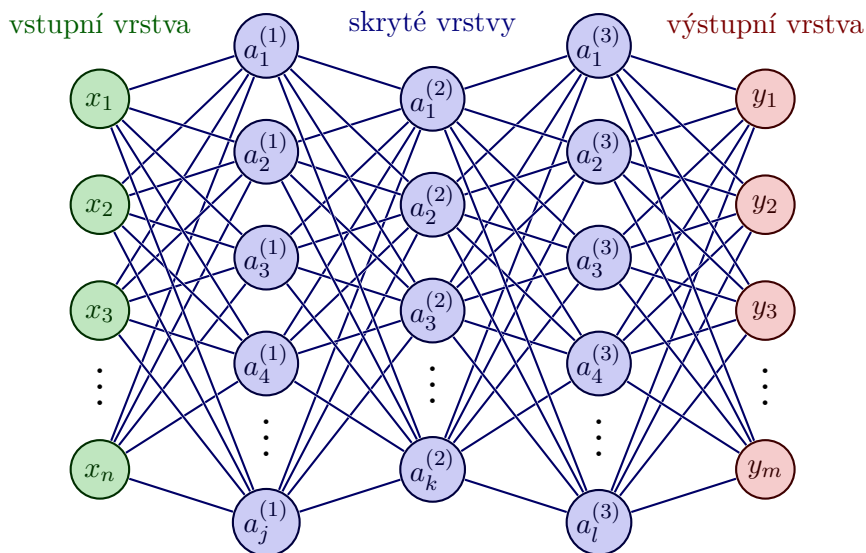
Softsign je další z aktivačních funkcí. Je podobná jako hyperbolický tangens, ale konverguje pomaleji. Definuje se jako:

$$g(a) = \frac{a}{|a| + 1}$$

Používána je zejména v regresních problémech.

1.1.3 Dopředná neuronová síť

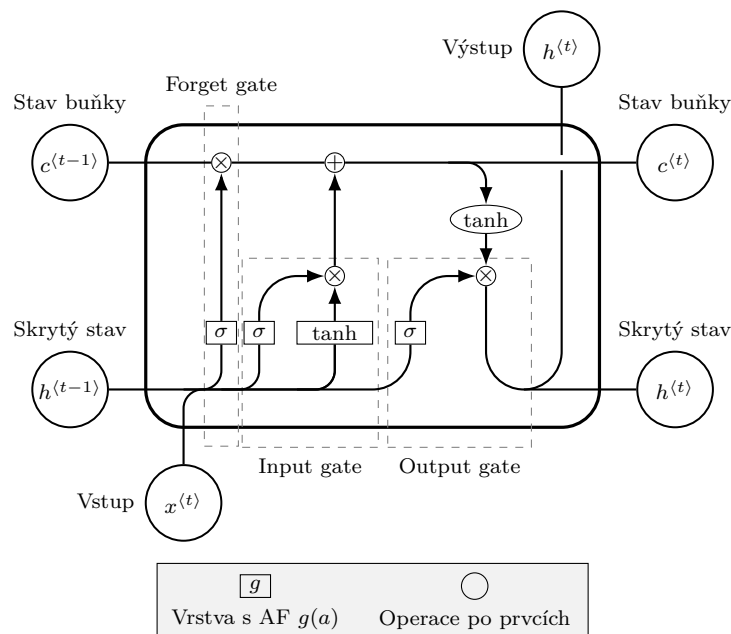
Dopředná neuronová síť, někdy také nazývaná vícevrstvý perceptron (přestože se používají jiné aktivační funkce než u původního perceptronu), je zřetězení umělých neuronů tak, že aktivace některých neuronů mohou být vstupem jiných. Aby síť byla dopředná, musí být splněna podmínka, že nesmí obsahovat žádnou zpětnovazební smyčku, tedy že výstup neuronu nesmí být vstupem některého z předchozích neuronů, na jejichž aktivaci tento výstup závisí. Přestože to není podmínkou, neurony se obvykle uspořádávají do vrstev tak, že aktivace neuronů v dané vrstvě slouží jako vstupy neuronů ve vrstvě další. Výstup pak lze vyhodnocovat postupně po vrstvách. Poslední z vrstev se nazývá výstupní, ostatní vrstvy neuronů jsou tzv. skryté. Vstupy, přestože se nejedná o neurony, jsou označovány jako vstupní vrstva, ale obvykle se nezapočítávají do počtu vrstev sítě. Jako počet vrstev se označuje součet počtu skrytých vrstev a 1 (výstupní vrstva). Počet neuronů v jednotlivých vrstvách je libovolný a záleží na návrhářích sítě. (Bishop, 1995) Příklad 4 vrstvé sítě je na obrázku 1.2. Obvykle se propojují všechny neurony z dané vrstvy s každým z neuronů ve vrstvě následující, obecně ale mohou některá tato spojení chybět.



OBRÁZEK 1.2: Příklad dopředné neuronové sítě o 3 skrytých vrstvách

1.1.4 Rekurentní neuronová síť, LSTM

Rekurentní neuronová síť se od dopředné liší tím, že obsahuje smyčky, kterými se přenáší informace zpět do předcházejících vrstev. Takto přenášeným



OBRÁZEK 1.3: Schéma Long Short-Term Memory jednotky

informacím se také říká skryté stavy. Výstup rekurentní sítě se vyhodnocuje v několika diskretních časových okamžicích pro časovou posloupnost jednotlivých vstupů.

Long Short-Term Memory (LSTM) je speciální architektura rekurentní neuronové sítě, která ve spojení s vhodným trénovacím algoritmem řeší problém mizejícího gradientu (*gradient vanishing problem*), který se vyskytoval při použití metod běžných v době vzniku LSTM. Hochreiter a Schmidhuber navrhli použití LSTM jednotky, ve které se pomocí 2 hradel (vstupní a výstupní) řídí tok informací mezi jednotkou a okolím a mezi časovými okamžiky. Díky tomu je možné sestavit síť, která umí v datech překlenout různé dlouhé časové intervaly a reagovat na vzorce patrné ve vstupech v průběhu času. Také je možné takovou síť efektivně trénovat. (Hochreiter a Schmidhuber, 1997)

LSTM jednotka má několik vstupů a výstupů (všechny jsou znázorněny na obrázku 1.3). Jednotlivé vstupy jsou:

1. běžný vstup $x^{(t)}$ v daném časovém okamžiku t ;
2. skrytý stav $h^{(t-1)}$ z předchozího časového okamžiku $t - 1$ a
3. stav buňky $c^{(t-1)}$ z předchozího časového okamžiku $t - 1$.

Výstupy jsou pak:

1. běžný výstup a zároveň skrytý stav $h^{(t)}$ v časovém okamžiku t a
2. stav buňky $c^{(t)}$ v daném časovém okamžiku t .

Klíčovou součástí LSTM jednotky je LSTM buňka, která slouží jako paměť a umožňuje selektivně uchovávat informace napříč časovými okamžiky. Jedná se o přímé propojení v rámci jedné jednotky s jednokrokovým zpožděním, její stav je označen c .

Později bylo navrženo přidání 3. hradla, tzv. *forget gate*, které řídí uchovávání informace v LSTM buňce, což mělo pozitivní vliv při aplikaci LSTM na souvislé posloupnosti. (Gers, Schmidhuber a Cummins, 2000) Struktura LSTM jednotky se všemi třemi hradly je na obrázku 1.3.

1.1.5 Použití neuronových sítí na problémy v reálném čase

1.2 Internet věcí (IoT)

1.3 Venkovní žaluzie s pohonem

2 Definice úlohy



Jedním z cílů práce je navržení systému pro automatické řízení žaluzií. V této kapitole je definována úloha automatického řízení, kterou tento systém bude řešit.

Jak již bylo zmíněno v sekci 1.3, žaluzie mohou být vytažené do různé výšky a jejich lamely mohou svírat různé úhly se svislou rovinou. Jejich stav tak lze obecně vyjádřit pomocí dvou proměnných, které je nutné odhadovat (tzv. *target*). V této práci jsou tyto proměnné označovány jako `position` a `tilt` a jejich bližší význam je uveden v sekci 2.2.

Za běžných okolností stav žaluzie určuje její uživatel, jehož chování má systém napodobovat. V sekci 2.1 jsou vyjádřeny předpoklady o rozhodování uživatele, ze kterých vzešlo celkem 15 proměnných, které budou systému sloužit jako vstup (tzv. příznaky).

Vstupy uspořádáme do příznakového vektoru $\mathbf{x} = [x_1, x_2, \dots, x_{15}]^T$, kde x_i je příznak uvedený v i -tém řádku tabulky 2.1. Podobně výstupy uspořádáme do výstupního vektoru $\mathbf{y} = [\text{position}, \text{tilt}]$.

Poté lze definovat úlohu tak, že na základě obecné p -prvkové posloupnosti příznakových vektorů $\left(\mathbf{x}^{(t)}\right)_{t=0}^{p-1}$ má být generován odhad $\hat{\mathbf{y}}$ výstupního vektoru tak, aby nastavení žaluzie dle výstupního vektoru odpovídalo preferencím uživatele. Jedná se tedy o úlohu regrese. Parametr p závisí na zvoleném řešení a umožňuje k odhadu využít historický kontext příznaků.

V kapitole 5 jsou navržena 3 různá řešení této úlohy. První z nich je ručně vytvořený systém založený na pravidlech, který slouží pro porovnání (baseline). Další 2 navržená řešení využívají NS jako regresory. S využitím příkladů p -prvkové posloupnosti příznakových vektorů $\left(\mathbf{x}^{(t)}\right)_{t=0}^{p-1}$ v časových okamžicích t a odpovídajícího výstupního vektoru $\mathbf{y}^{(t)}$ v čase $t = p - 1$ (trénovací data) se natrénuje neuronová síť, která tak řeší výše definovanou úlohu. Pro dopřednou NS je $p = 1$, pro rekurentní je obecně $p > 1$, v této práci je uvažováno $p = 64$. Sběr reálných trénovacích dat je popsán v kapitole 4.

2.1 Vstupy systému

Při návrhu systému se vycházelo z předpokladu, že uživatel rozhoduje o ručním řízení žaluzií na základě odhadu, případně měření, některých fyzikálních veličin. Pro nahrazení rozhodování uživatele automatickým řízením je nutné stanovit množinu veličin, které se budou v pravidelných intervalech měřit a na základě jejich hodnot automaticky generovat vhodné řízení pomocí modelů získaných strojovým učením. Tato sekce se zabývá volbou jednotlivých prvků příznakového vektoru, tedy příznaků.

Uživatel může nastavením žaluzie sledovat různé cíle, hlavní 4 byly identifikovány takto:

1. Za šera a tmy mají být žaluzie zatažené pro zajištění soukromí uvnitř místnosti při použití umělého osvětlení a zabránění vniknutí světla od projíždějících vozidel v době spánku uživatele v místnosti.
2. Při slunečných dnech, kdy venkovní vzduch dosahuje teplot blízkých pokojové teplotě nebo je převyšuje, mají žaluzie bránit průniku přímého slunečního záření do místnosti.
3. Při teplotách nižších, kdy je nutné interiér vytápět, by žaluzie naopak měly umožnit maximální průchod záření oknem, aby se tak místnost vytápěla a nebylo nutné využívat zbytečně vysoké množství energie na běžné vytápění.
4. V případě silného větru by měly být žaluzie vytažené, aby se zabránilo jejich poškození.

Z bodu 1 vyplývá, že uživatel sleduje intenzitu osvětlení exteriéru, orientačně je možné ji určit podle času a data, ale silně závisí také na počasí (zejména oblačnosti) a protože jsou na trhu dostupné senzory, které komunikují pomocí standardních sběrnic, je vhodnější ji přímo měřit. Použití umělého osvětlení uvnitř místnosti jimi lze detekovat také. Vnitřní a vnější intenzita osvětlení tedy byly zvoleny jako příznaky.

Body 2 a 3 zmiňují vliv teploty venkovního vzduchu na manuální ovládání uživatelem. Dá se ale předpokládat, že pokud by teplota vzduchu uvnitř místnosti byla podle uživatele příliš nízká, nezastíňoval by okno a nechal místnost vytápět i slunečním zářením i přes vysokou vnější teplotu. Proto byly dalšími příznaky zvoleny teplota vzduchu uvnitř a teplota vzduchu venku. Kromě toho uživatel může sledovat předpověď počasí a na jejím základě vyhodnotit, že je vhodné stínit dříve než teplota v průběhu dne vzroste, protože by už nemusel mít možnost vyhrátý interiér ochladit venkovním vzduchem. Jako příznaky se tedy zvolily předpovězené teploty na hodinu, 2 a 3 dopředu, předpověď nejvyšší denní teploty a odhadovaný stav počasí ve formě číselného kódu. Okamžité teploty byly měřeny pomocí digitálního teploměru (sekce 3.1), předpovědi a odhad se získávaly z internetu pomocí OpenWeather API (sekce 4.3).

Z bodu 4 vyplývá vliv rychlosti větru na žaluzie, proto byla zvolena jako další příznak. Manuální vyhodnocení uživatelem probíhá obvykle subjektivně podle hluku způsobeného kmitáním lamel žaluzií v boční vodicí drážce. Na něj může kromě rychlosti mít vliv i směr větru, kvůli různímu se obtékání domu proudícím vzduchem. Obě veličiny byly zvoleny jako příznaky, jejich měření bylo vyhodnoceno jako příliš složité a nákladné a odhady jejich hodnot se tedy získávají stejně jako předpovědi teploty (sekce 4.3). Systém žaluzií od firmy Somfy také obsahuje vlastní anemometr, který žaluzie v případě silného větru zadá žaluziím příkaz k vytažení, uživatel ale může být opatrný a žaluzie vytahovat již při nižších rychlostech větru.

Lidé dále mohou mít pravidelné zvyky, které ovlivňují nastavení žaluzií. Z toho důvodu byly jako příznaky zvoleny některé časové údaje: počet uplynulých sekund v rámci dne, počet uplynulých dnů v rámci týdne a pořadí dne

v roce. Poslední z nich souvisí také s obvyklým počasím, které může ovlivňovat nastavení žaluzií. V případě nepřítomnosti uživatele v domácnosti nemusí nastavení žaluzií přesně odpovídat obvyklému záměru uživatele, proto je jedním z příznaků také jeho přítomnost v domácnosti. Ta se vyhodnocuje v systému domácí automatizace na základě periodicky přenášené polohy uživatelského telefonu.

Přehled všech 15 příznaků, jejich označení, jednotky a očekávané nejnížší a nejvyšší hodnoty je uveden v tabulce 2.1

Příznak	Označení	Množina
Den v roce	year_day	$\langle 1, 365 \rangle$
Den v týdnu	week_day	$\langle 0, 6 \rangle$
Denní čas [s]	day_secs	$\langle 1, 86400 \rangle$
Uživatel je doma	home	$\{0, 1\}$
Vnitřní teplota [°C]	temp_in	$\langle 10, 35 \rangle$
Venkovní teplota [°C]	temp_out	$\langle -20, 40 \rangle$
Osvětlení uvnitř [lux]	lum_in	$\langle 0, 2000 \rangle$
Osvětlení venku [lux]	lum_out	$\langle 0, 60000 \rangle$
OWM - maximální denní teplota [°C]	owm_temp_max	$\langle -20, 40 \rangle$
OWM - předpověď teploty za 1h [°C]	owm_temp_1h	$\langle -20, 40 \rangle$
OWM - předpověď teploty za 2h [°C]	owm_temp_2h	$\langle -20, 40 \rangle$
OWM - předpověď teploty za 3h [°C]	owm_temp_3h	$\langle -20, 40 \rangle$
OWM - kód stavu počasí	owm_code	$\langle 200, 804 \rangle$
OWM - rychlost větru [$\frac{m}{s}$]	owm_wind_speed	$\langle 0, 50 \rangle$
OWM - směr větru [°]	owm_wind_heading	$\langle 0, 359 \rangle$

TABULKA 2.1: 15 příznaků použitých pro automatické ovládání žaluzií, jejich označení v systému a množiny očekávaných hodnot.

Po natrénování použitých algoritmů (kapitola 5) byla vyhodnocena důležitost jednotlivých příznaků pomocí algoritmu *permutation feature importance* (Breiman, 2001), při kterém se sleduje vliv permutace hodnot jednotlivých příznaků mezi vzorky v rámci datové sady na přesnost predikce. Důležitost každého příznaku pak odpovídá rozdílu MSE s promíchanými hodnotami tohoto příznaku a s původní datovou sadou (nezamíchanou). Pro každý z příznaků se MSE vyhodnotila 20x pro různé permutace a vypočetl se z nich aritmetický průměr. Důležitost byla vyjádřena jako rozdíl tohoto průměru a původní MSE.

2.2 Výstupy systému

Bibliografie

- [1] Warren S McCulloch a Walter Pitts. „A logical calculus of the ideas immanent in nervous activity“. In: *The bulletin of mathematical biophysics* 5.4 (1943), s. 115–133.
- [2] F. Rosenblatt. „The perceptron: A probabilistic model for information storage and organization in the brain.“ In: *Psychological Review* 65.6 (1958), s. 386–408. DOI: 10.1037/h0042519. URL: <https://doi.org/10.1037/h0042519>.
- [3] Christopher M. Bishop. *Neural networks for pattern recognition*. Oxford University Press, 1995.
- [4] Sepp Hochreiter a Jürgen Schmidhuber. „Long Short-term Memory“. In: *Neural computation* 9 (pros. 1997), s. 1735–80. DOI: 10.1162/neco.1997.9.8.1735.
- [5] Charles Dugas et al. „Incorporating Second-Order Functional Knowledge for Better Option Pricing“. In: *Advances in Neural Information Processing Systems*. Ed. T. Leen, T. Dietterich a V. Tresp. Sv. 13. MIT Press, 2000, s. 472–478. URL: <https://proceedings.neurips.cc/paper/2000/file/44968aece94f667e4095002d140b5896-Paper.pdf>.
- [6] Felix A. Gers, Jürgen Schmidhuber a Fred Cummins. „Learning to Forget: Continual Prediction with LSTM“. In: *Neural Computation* 12.10 (2000), s. 2451–2471. DOI: 10.1162/089976600300015015.
- [7] Leo Breiman. „Random Forests“. In: *Machine Learning* 45 (říj. 2001), s. 5–32. DOI: 10.1023/A:1010950718922.
- [8] Vinod Nair a Geoffrey E. Hinton. „Rectified Linear Units Improve Restricted Boltzmann Machines“. In: *ICML*. Ed. Johannes Fürnkranz a Thorsten Joachims. Omnipress, 2010, s. 807–814. URL: <https://icml.cc/Conferences/2010/papers/432.pdf>.
- [9] Štěpánka Lubinová. *Stínění oken: žaluzie, rolety, markýzy a slunolamy*. Grada Publishing, 2013. ISBN: 978-80-247-4579-4.
- [10] Prajit Ramachandran, Barret Zoph a Quoc V Le. „Searching for activation functions“. In: *arXiv preprint arXiv:1710.05941* (2017).
- [11] Chigozie Nwankpa et al. „Activation Functions: Comparison of trends in Practice and Research for Deep Learning“. In: *CoRR* abs/1811.03378 (2018). arXiv: 1811.03378. URL: <http://arxiv.org/abs/1811.03378>.