



Bakalářská práce

Neuronové sítě pro zpracování přirozeného jazyka

Autor:
Jan Čedík

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

Závěrečná práce pro získání titulu Bakalář (Bc.)

Katedra kybernetiky

22. května 2023

ZÁPADOČESKÁ UNIVERZITA V PLZNI

Fakulta aplikovaných věd
Akademický rok: 2022/2023

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Jan ČEDÍK**
Osobní číslo: **A20B0365P**
Studijní program: **B0714A150005 Kybernetika a řídicí technika**
Specializace: **Umělá inteligence a automatizace**
Téma práce: **Neuronové sítě pro zpracování přirozeného jazyka**
Zadávací katedra: **Katedra kybernetiky**

Zásady pro vypracování

1. Nastudujte nejnovější metody v oblasti neuronových sítí pro zpracování přirozeného jazyka, zaměřte se na metody 'text-to-text' a metody klasifikace textu.
2. Seznamte se s principy použití T5 (Text-to-Text Transfer Transformer) a naučte se pracovat s katedrálním rozhraním pro dotrénování T5.
3. Nasbírejte vhodná data, sestavte automatickou chatovací platformu a vytvořte pro ni webové uživatelské rozhraní.
4. Vyberte vhodnou metodu filtrace nevhodných odpovědí a vyhodnoťte její přínos.
5. Vyhodnoťte možnosti aplikace použitých metod na Raspberry Pi.

Rozsah bakalářské práce: **30-40 stránek A4**
Rozsah grafických prací:
Forma zpracování bakalářské práce: **tištěná**

Seznam doporučené literatury:

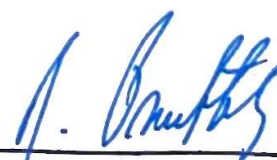
1. Bishop, C. (1995). Neural networks for pattern recognition. Oxford University Press, USA.
2. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. CoRR, abs/1810.04805. <http://arxiv.org/abs/1810.04805>
3. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... Liu, P. J. (2019). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. CoRR, abs/1910.10683. <http://arxiv.org/abs/1910.10683>

Vedoucí bakalářské práce: **Ing. Martin Bulín, M.Sc.**
Výzkumný program 1

Datum zadání bakalářské práce: **17. října 2022**
Termín odevzdání bakalářské práce: **22. května 2023**



Doc. Ing. Miloš Železný, Ph.D.
děkan



Prof. Ing. Josef Psutka, CSc.
vedoucí katedry

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 22. května 2023

.....

Podpis

„The best way to predict your future is to create it.“

Abraham Lincoln

ZÁPADOČESKÁ UNIVERZITA V PLZNI

Abstrakt

Fakulta aplikovaných věd

Katedra kybernetiky

Bakalář (Bc.)

Neuronové sítě pro zpracování přirozeného jazyka

Jan Čedík

Tato bakalářská práce se zabývá zpracováním přirozeného jazyka pomocí neuronových sítí. V současnosti dochází k rychlému rozvoji neuronových sítí v tomto oboru, přičemž společnost OpenAI je hlavním průkopníkem s jejími velkými modely založenými na architektuře Transformer. Teoretická část se zaměřuje právě na tuto architekturu, konkrétně na architektury T5, BERT a GPT. V praktické části provádíme experimenty jak v češtině, tak v angličtině, založené na architektuře T5. Shromáždili jsme data obsahující přirozené dialogy mezi jednotlivci. Pro dosažení co nejlepších předpovědí výstupu se snažíme o různé modifikace trénovacích dat, jako je značkování nebo augmentace. Hodnocení se provádí pomocí modelu sémantické návaznosti, který porovnává vstup a výstup modelu.

Klíčová slova

neuronové sítě, zpracování přirozeného jazyka, Transformer architektura, BERT, T5, GPT, augmentace, úprava dat, chatbot

UNIVERSITY OF WEST BOHEMIA

Abstract

Faculty of Applied Sciences

Department of Cybernetics

Bachelor (Bc.)

Neural networks for natural language processing

Jan Čedík

This bachelor's thesis deals with natural language processing using neural networks. Currently, there is a rapid development of neural networks in this field, with the company OpenAI being a major pioneer with its large models based on the Transformer architecture. The theoretical part focuses on this architecture, specifically the T5, BERT, and GPT architectures. In the practical part, we conduct experiments in both Czech and English languages based on the T5 architecture. We gathered data containing natural dialogues between individuals. Through various modifications of the training data, such as tagging or augmentation, we aim to achieve the best possible output predictions. Evaluation is performed using a semantic continuity model, comparing the input and output of the model.

Key words

neural networks, natural language processing, Transformer architecture, BERT, T5, GPT, augmentation, data modification, chatbot

Poděkování

Touto cestou bych rád vyjádřil svou vděčnost vedoucímu mé práce, panu Ing. Martinu Bulínovi, MSc. za jeho vedení, cenné rady a ochotu v průběhu celé práce.

Také bych rád poděkoval své rodině a přátelům za jejich neustálou podporu, povzbuzení a pochopení během celého procesu psaní práce.

Výpočetní prostředky byly poskytnuty z projektu e-INFRA CZ (ID: 90254), podporovaného Ministerstvem školství, mládeže a tělovýchovy České republiky.

Obsah

Abstrakt	v
1 Úvod	1
1.1 Současný stav	2
1.2 Cíle práce	2
1.3 Struktura práce	3
2 Neuronové sítě pro zpracování jazyka	4
2.1 Rekurentní neuronová síť	5
2.2 Transformer	6
Architektura	7
Enkodér	7
Dekodér	8
Positional encoding	9
Attention mechanismus	10
Self-attention	11
Multi-head self-attention	12
Nejpoužívanější Transformer architektury	13
2.2.1 T5	13
Architektura	14
C4 dataset	15
2.2.2 GPT	16
Architektura	16
Předtrénování bez učitele	17
Dotrénování s učitelem	18
Specifické úlohy	19
2.2.3 BERT	20
Architektura	20
Vstupní data	20
Předtrénování modelu	21
Dotrénování modelu	22
2.3 Automatický překlad	22
2.3.1 Algoritmus překladu	22
2.4 Augmentace dat	23
2.4.1 Augmentace pomocí synonym	23
2.4.2 Augmentace pomocí překladů	24
3 Data	25
3.1 Dialogové datasey	25
3.1.1 Commonsense-Dialogues Dataset (d01)	25

3.1.2	AnnoMI (d02)	26
3.1.3	DailyTalk (d03)	27
3.1.4	MELD (d04)	28
3.1.5	SGD (d05)	28
3.1.6	Harry Potter Movies Dataset (d06)	29
3.1.7	Simpsons (d07)	30
3.1.8	The Office (d08)	30
3.2	Zpracovaný dialogový korpus	31
4	Experimenty a výsledky	32
4.1	Architektura testovaného modelu	32
4.2	Rozdělení dat	32
4.3	Vyhodnocovací technika	33
4.4	Experimenty	34
4.4.1	Experiment E1	34
4.4.2	Experiment E2	35
	Experiment E2 a)	36
	Experiment E2 b)	37
4.4.3	Experiment E3	38
4.4.4	Překlad	39
	Francouzský překlad	40
	Španělský překlad	40
4.4.5	Synonyma	41
5	Diskuze	43
5.1	Shrnutí výsledků	43
5.2	Reálné využití	48
6	Závěr	49
6.1	Práce do budoucna	49
	Literatura	51
	A1 Zpracovaný dialogový korpus CS	54
	A2 Zpracovaný dialogový korpus EN	56

Seznam obrázků

2.1	Interpretace výpočtu výstupu neuronové sítě	5
2.2	Porovnání RNN a FFNN	5
2.3	Původní Transformer architektura	7
2.4	Reziduální propojení	8
2.5	Positional encoding	9
2.6	Positional encoding používaný v Transformer architektuře .	10
2.7	Uvažování spojitostí při překladu	11
2.8	Scaled dot product attention	12
2.9	Multi-head self-attention	13
2.10	Text-to-Text framework	14
2.11	Architektury T5	15
2.12	GPT architektura	17
2.13	Různé reprezentace vstupu pro GPT	19
2.14	Vstupní data pro model BERT	21
2.15	Algoritmus překladu	23
2.16	Augmentace pomocí synonym	24
2.17	Augmentace pomocí překladu	24
4.1	Model sémantické návaznosti	33
4.2	Vytvoření nových dialogů pomocí překladu	40
4.3	Vytvoření nových dialogů pomocí synonym	42
5.1	Porovnání výsledků experimentů E1, E2, E3	44
5.2	Porovnání výsledků českých modelů E1, E2, E3	45
5.3	Porovnání výsledků anglických modelů E1, E2, E3	46
5.4	Porovnání výsledků experimentů s augmentací	47
5.5	Webové rozhraní pro chatovací aplikaci	48
5.6	Schéma robota	48

Seznam tabulek

2.1	Velikosti předtrénovaných modelů T5	15
2.2	Porovnání velikostí GPT modelů	17
3.1	Vygenerovaný (.tsv) formát dialogových dat z datasetu Commonsense-dialogues	26
3.2	Vygenerovaný (.tsv) formát dialogových dat z datasetu An-noMI	26
3.3	Vygenerovaný (.tsv) formát dialogových dat z datasetu DailyTalk	27
3.4	Formát původních dat datasetu MELD	28
3.5	Vygenerovaný (.tsv) formát dialogových dat z datasetu MELD	28
3.6	Vygenerovaný (.tsv) formát dialogových dat z datasetu SGD	29
3.7	Formát původních dat datasetu Harry Potter	29
3.8	Vygenerovaný (.tsv) formát dialogových dat z datasetu Harry Potter	30
3.9	Formát původních dat datasetu Simpsons	30
3.10	Vygenerovaný (.tsv) formát dialogových dat z datasetu Simpsons	30
3.11	Formát původních dat datasetu The Office	31
3.12	Vygenerovaný (.tsv) formát dialogových dat z datasetu The Office	31
3.13	Zpracovaný dialogový korpus pro trénování konverzačního modelu	31
4.1	Rozdělení dat	32
4.2	Počet promluv v jednotlivých množinách datasetu - trénovací, testovací, validační	33
4.3	Připravená data pro E1 (bezkontextová varianta)	34
4.4	Vybrané ukázky vygenerovaného výstupu - anglický model E1 (bezkontextová varianta)	35
4.5	Vybrané ukázky vygenerovaného výstupu - český model E1 (bezkontextová varianta)	35
4.6	Vyhodnocení experimentu E1	35
4.7	Připravená data pro E2a	36
4.8	Vybrané ukázky vygenerovaného výstupu - anglický model E2a (kontextová varianta)	36
4.9	Vybrané ukázky vygenerovaného výstupu - český model E2a (kontextová varianta)	36
4.10	Vyhodnocení experimentu E2a	37
4.11	Připravená data pro E2b	37

4.12 Vybrané ukázky vygenerovaného výstupu - anglický model E2b (kontextová varianta)	37
4.13 Vybrané ukázky vygenerovaného výstupu - český model E2b (kontextová varianta)	38
4.14 Vyhodnocení experimentu E2b	38
4.15 Připravená data pro E3	38
4.16 Vybrané ukázky vygenerovaného výstupu - anglický model E3 (kontextová varianta s ohledem na řečníka)	39
4.17 Vybrané ukázky vygenerovaného výstupu - český model E3 (kontextová varianta s ohledem na řečníka)	39
4.18 Vyhodnocení experimentu E3	39
4.19 Vybrané ukázky vygenerovaného výstupu - anglický model FR překlad (kontextová varianta s ohledem na řečníka)	40
4.20 Vyhodnocení experimentu FR překlad	40
4.21 Vybrané ukázky vygenerovaného výstupu - anglický model ES překlad (kontextová varianta s ohledem na řečníka)	41
4.22 Vyhodnocení experimentu ES překlad	41
4.23 Vybrané ukázky vygenerovaného výstupu - anglický model synonyma (kontextová varianta s ohledem na řečníka)	42
4.24 Vyhodnocení experimentu Synonyma	42

Kapitola 1

Úvod

Za poslední roky došlo k velkému pokroku v oblasti zpracování přirozeného jazyka, kdy pomocí umělé inteligence dokážeme rozpoznávat, zpracovávat a generovat přirozenou řeč. Tyto možnosti umožňují otevření nových dveří v širokém spektru úloh a odvětví.

V současné době dochází k obrovskému rozmachu neuronových sítí pro zpracování přirozeného jazyka. Mezi nejmodernější neuronové sítě lze zařadit architekturu *Transformer* (Vaswani et al., 2017), která umožnila nové přístupy k těmto úlohám. Na základě této architektury bylo vytvořeno mnoho různých variant v závislosti na typu úlohy, jako je například *GPT* (Radford a Narasimhan, 2018) vyvinutý společností OpenAI, jenž vyniká schopností generovat přirozené texty. Společnost OpenAI, díky obrovským jazykovým modelům, dosahuje mnoha *state-of-the-art* výsledků. Zásadní úspěch slaví jejich nejnovější a zároveň největší model *GPT-4*.

I přes úspěchy architektury *GPT* je stále důležité provádět experimenty i pomocí jiných architektur založených na architektuře *Transformer*. V této bakalářské práci se především zaměříme na architektury *T5* (*Text-To-Text Transfer Transformer*) (Raffel et al., 2020), *BERT* (*Bidirectional Encoder Representations from Transformers*) (Devlin et al., 2019) a *GPT* (*Generative Pre-trained Transformer*) (Radford a Narasimhan, 2018). Samotné experimenty jsou následně prováděny na natrénovaných modelech založených na *T5*.

V teoretické části této práce se zaměříme na základní popis již zmíněných architektur. *Text-To-Text Transfer Transformer* se proslavil jako všestranný model, který lze využít pro rozsáhlou škálu úloh zpracování přirozeného jazyka, jež využívají převodu textu na text. *Generative Pre-trained Transformer*, jak už název napovídá, dokáže generovat smysluplné texty. *BERT* uspěl například v předpovídání chybějícího slova ve větě, pomocí daného kontextu, nebo klasifikování textů do různých kategorií.

V části s experimenty se zaměříme na úpravu trénovacích dat, jelikož je známo, že velmi důležitou roli v procesu trénování hraje kvalita trénovacích dat. Pokusíme se data upravit tak, aby výstup modelu byl závislý na kontextu předchozích promluv pomocí mnoha tagů doplněných do sekvencí. Cílem je zjistit, která úprava dosáhne nejlepších výsledků při odpovídání na různé vstupní texty.

1.1 Současný stav

V posledních letech dosahuje nejlepších výsledků společnost OpenAI se svými modely založenými na architektuře *GPT*. V roce 2022 oficiálně spustili veřejně dostupnou chatovací platformu *ChatGPT* založenou na modelu *GPT-3*. V roce 2023 představili zatím největší model *GPT-4*, který dokáže přijímat jak textové, tak obrazové vstupy (Marr, 2023).

Modely *GPT* vynikají v řešení mnoha úloh v závislosti na *promptu*, který se nalézá ve vstupním textu. Pro ukázkou jsme použili právě chatovací platformu *ChatGPT*. Jako vstupní text jsme využili následující větu:

Napiš 5 promptů, které dokážeš zpracovat.

Výstup vypadal následovně:

Jistě! Zde je pět různých promptů, se kterými mohu pracovat:

1. "Jaký je vliv globálního oteplování na životní prostředí?"
2. "Vysvětli mi princip fungování umělé inteligence."
3. "Co jsou největší výhody a nevýhody používání solární energie?"
4. "Jaké jsou hlavní příčiny a důsledky nedostatku pitné vody ve světě?"
5. "Porovnej a popiš rozdíly mezi demokracií a autoritářským režimem."

Současným největším problémem platformy *ChatGPT* je takzvaný *hallucination*. Tento problém odkazuje na generování výstupů, které se mohou jevit jako smysluplné, avšak jsou buď fakticky nesprávně nebo nesouvisí s daným kontextem. Další limitací je zpoplatněné použití modelu a především to, že veškerý text, který použijeme jako vstup do modelu, posíláme společnosti OpenAI a to může být problém zpravidla u citlivých dat.

1.2 Cíle práce

Cíle této práce jsou:

1. zpracovat rešerši nejnovějších přístupů v oblasti neuronových sítí pro zpracování přirozeného jazyka
2. získat vhodná data pro natrénování českých a anglických modelů pro automatickou chatovací platformu
3. sestavit robustní dialogový korpus z nasbíraných dat v českém a anglickém jazyce
4. experimentálně ověřit vliv různých modifikací trénovacích dat pro trénink *T5* v rámci konverzačního modelu a následně vyhodnotit vliv zahrnutí konverzační historie pro následnou predikci modelu

1.3 Struktura práce

Tato práce se skládá ze šesti kapitol.

Úvodní část práce (Kapitola 1) se zabývá současným stavem v oblasti zpracovávání přirozeného jazyka a nejmodernějšími technikami, jež jsou v tomto odvětví používány.

Kapitola 2 je věnována popisu neuronových sítí využívaných pro zpracování přirozeného jazyka. Nejprve je popsána obyčejná dopředná neuronová síť, na kterou navazuje popis rekurentní neuronové sítě. V Sekci 2.2 jsou popsány principy fungování architektury Transformer. Zbytek kapitoly je věnován popisu jednotlivých architektur postavených na architektuře Transformer a metodám, jež byly použity pro zpracování a úpravu dat.

Ve Kapitole 3 jsou popsány veškeré získané datasey obsahující dialogy spolu s potřebnými anotacemi.

Kapitola 4 obsahuje popis jednotlivých experimentů, které byly provedeny. Pro každý experiment bylo provedeno vyhodnocení, pomocí modelu návaznosti (4.3).

V předposlední části (Kapitola 5) se věnujeme diskuzi o výsledcích, kterých jsme dosáhli. Výsledky jsou následně porovnány.

Kapitola 6 je věnována závěru, rekapitulaci a námětům na budoucí práci.

Kapitola 2

Neuronové sítě pro zpracování jazyka

Neuronové sítě jsou inspirovány biologickým protějškem jímž je mozek a nervový systém. Mозek má charakteristické schopnosti jako je učení a přizpůsobování. Neuronové sítě staví na neurálních principech pro řešení dílčích úloh.

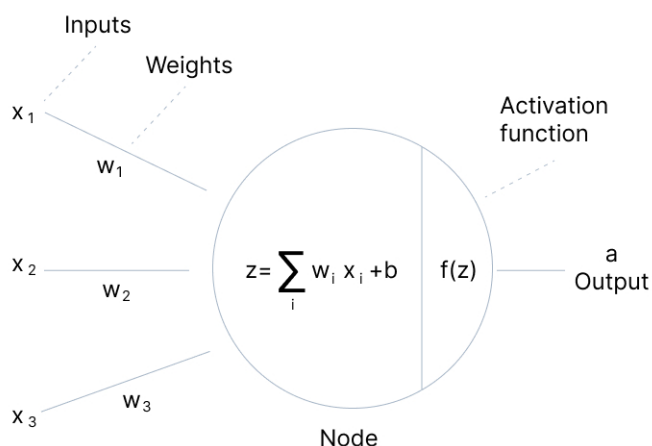
Základním stavebním kamenem neuronových sítí je neuron, který lze vnímat jako jednotku zpracovávající informace. V samotné neuronové síti jsou pak neurony propojeny synapsami, kde každá má svoji váhu. Díky těmto spojkám každý neuron v síti dostává váženou informaci od neuronů, na které je připojen. Výstup je tvořen váženým součtem výstupních signálů procházející aktivační funkcí, který lze interpretovat následujícími rovnicemi (Bulín, 2021):

$$z_k = b + \sum_{i=1}^n (x_i \cdot w_{k,i}) \quad (2.1)$$

, kde z_k je agregace všech vstupů, b je parametr, který je přičten k váženému součtu, n udává počet vstupních neuronů, x_n je hodnota signálu a $w_{k,i}$ je váha synapse. Na výslednou hodnotu z_k je následně použita vybraná aktivační funkce f :

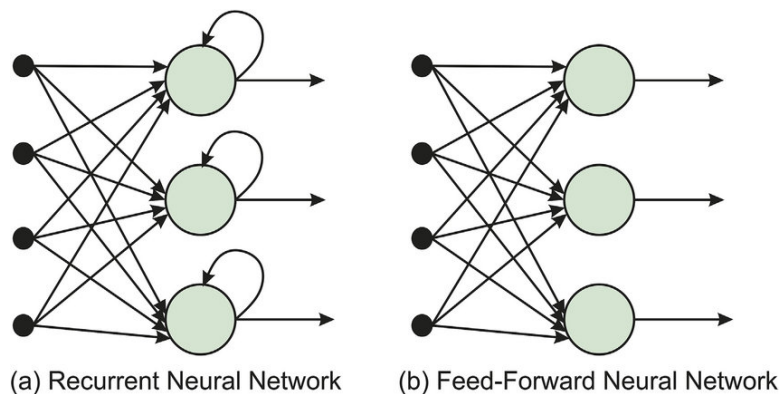
$$a_k = f(z_k) \quad (2.2)$$

Celý proces je shrnutý na Obrázku 2.1.



Obrázek 2.1: Interpretace výpočtu výstupu neuronové sítě (Baheti, 2021)

Neuronové sítě lze rozdělit do dvou hlavních skupin podle typu spojení mezi jednotlivými neurony. První skupinou jsou *neuronové sítě s dopřednou vazbou* a druhá skupina jsou *rekurentní neuronové sítě*, které jsou dále popsány v Sekci 2.1. Rozdíl mezi těmito typy sítí je následující - rekurentní sítě jsou rozšířené o smyčky, které umožňují výstupům z neuronů ve skryté vrstvě být opět vstupem do neuronů ve stejné vrstvě nebo i předchozí (Sazli, 2006).



Obrázek 2.2: (a) Rekurentní neuronová síť, (b) Dopředná neuronová síť (Eliasy a Przychodzen, 2020)

2.1 Rekurentní neuronová síť

Rekurentní neuronové sítě jsou obohaceny o koncept paměti, tím že si každý neuron v síti uchovává vnitřní stav, který předává informace z předchozího výpočtu do následujícího. Tato smyčka tvoří hlavní rozdíl mezi klasickými neuronovými sítěmi a rekurentními sítěmi. Smyčka umožňuje předávání informací v rámci vrstvy, to znamená, že vrstva používá svůj výstup jako vstup, na rozdíl od dopředných neuronových sítí,

v nichž se informace předávají pouze mezi vrstvami. Výhodou těchto sítí je možnost přijímat sekvenční data a rozpoznávat souvislost v datech díky jejich krátkodobé paměti. Nevýhodou pak může být *vanishing gradient problem*, jelikož hodnoty používané pro úpravu vah se v průběhu *backpropagation* algoritmu stávají velmi malými. Následné násobení vah gradientem, který se blíží nule, zabraňuje síti učit se nové váhy. Tento problém vede k tomu, že rekurentní neuronové sítě zapomínají, co viděly v delších sekvencích (Dancker, 2022).

Tyto sítě lze aplikovat na širokou škálu úloh avšak nejčastěji jsou využívány pro zpracovávání přirozeného jazyka a rozpoznávání řeči. Podrobnější popis rekurentních neuronových sítí lze nalézt v rigorózní práci Ing. Martina Bulína, MSc. On Using Multi-Agent Technologies to Build Neural Networks (Bulín, 2021).

2.2 Transformer

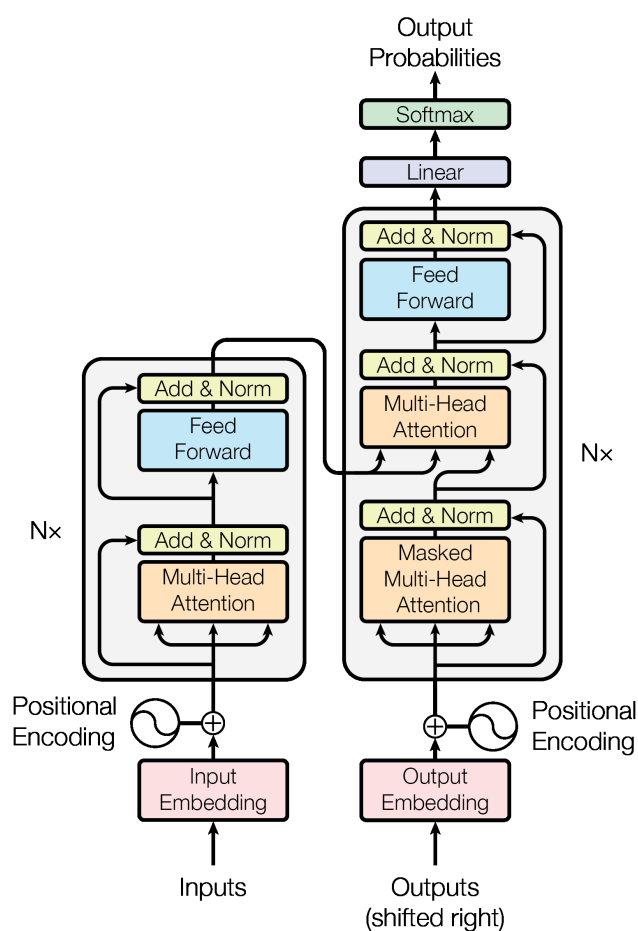
Neuronová síť typu Transformer byla poprvé zmíněna v odborném článku s názvem "Attention Is All You Need" v roce 2017 (Vaswani et al., 2017). Novinkou bylo využití enkodér-dekodér architektury založené na takzvaném *Attention mechanismu*. Další z podstatných rozdílů oproti jiným architektuám je možnost předávání vstupní sekvence paralelně, díky čemuž je možno efektivněji využít grafickou kartu a zvýšit tak rychlost trénování modelu. V následujících bodech jsou shrnuty problémy, se kterými se potýkají rekurentní neuronové sítě a jakým způsobem tyto problémy řeší Transformer (Sarkar, 2022).

- Rekurentní neuronové sítě se potýkají s problémem zohledňovat dlouhodobé závislosti v dlouhých textových dokumentech, což může ovlivňovat kvalitu výsledku. Architektura Transformer využívá již zmíněný Attention mechanismus, díky kterému lze vytvářet spojení mezi jakýmkoliv částmi sekvence a tak mají dlouhodobé závislosti stejnou prioritu jako krátkodobé závislosti.
- RNN sítě se potýkají s problémem gradient vanishing nebo gradient explosion. Transformer sítě se s tímto problémem potýkají minimálně až zanedbatelně, jelikož v Transformer sítích je trénována celá sekvence současně a na základě toho je přidáno pouze pár vrstev, což zajišťuje, že se problém gradient vanishing nebo explosion vyskytuje pouze zřídka.
- RNN sítě vyžadují delší kroky trénování pro dosažení lokálního nebo globálního minima. RNN síť lze vizualizovat jako velmi hlubokou rozvinutou síť, jejíž velikost závisí na délce sekvence, a tak vzniká velký počet parametrů, kde většina z nich je vzájemně propojena. Z toho vyplývá, že optimalizace vyžaduje dlouhou dobu trénování a velké množství kroků. V porovnání s Transformer sítí vyžadují více kroků k trénování.

- RNN sítě neumožňují paralelní výpočty, ačkoli grafické karty paralelní výpočty umožňují. Výpočet v RNN sítích probíhá sekvenčně, tudíž ho nelze paralelizovat.

Architektura

Stejně jako v dříve využívaných architekturách, které řešily *sequence-to-sequence* úlohy, i tato používá strukturu enkodér-dekodér. V tomto případě enkodér mapuje sekvenci vstupních symbolů $\mathbf{x} = (x_1, \dots, x_n)$ na sekvenci $\mathbf{z} = (z_1, \dots, z_n)$. Z dané sekvence \mathbf{z} dekodér vygeneruje výstupní sekvenci $\mathbf{y} = (y_1, \dots, y_n)$ symbolů. Model v každém kroku využívá již vygenerované symboly jako další vstupy při generování textu. Hlavním základem architektury Transformer je blok založený na *Attention mechanismu*, dále je architektura doplněna o plně propojené dopředné vrstvy (Vaswani et al., 2017).



Obrázek 2.3: Původní Transformer architektura (Vaswani et al., 2017)

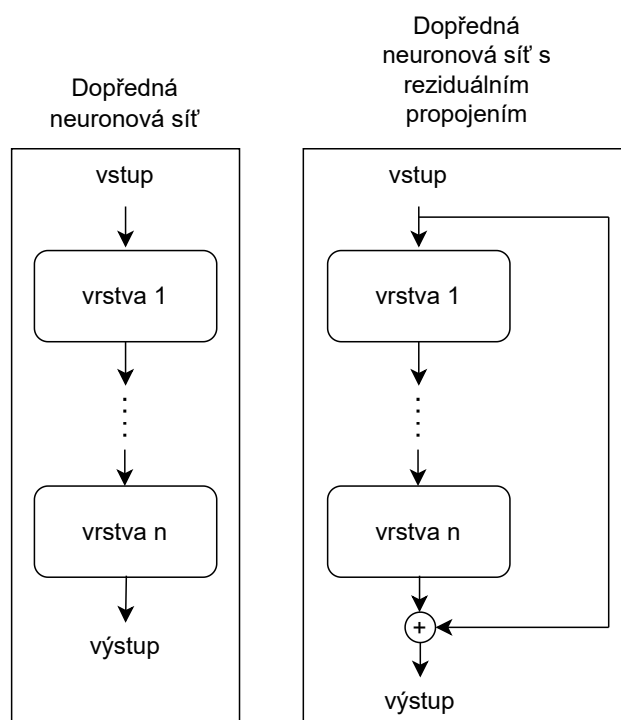
Enkodér

V obecném významu je enkodování převádění dat do určitého formátu. V úlohách automatického zpracování jazyka to znamená, že enkodér zajišťuje převedení textu do číselného vektoru.

Enkodér v původní Transformer architektuře (Obrázek 2.3), se skládá ze šesti identických vrstev. Každá z vrstev má ještě své vlastní dvě podvrstvy. První vrstva obsahuje *multi-head self-attention mechanismus* (blíže popsán v Sekci 2.2) a druhá vrstva je plně propojená dopředná neuronová síť. Kolem každé ze dvou dílčích vrstev jsou použita dvě reziduální spojení, která poskytují cestu pro data, což umožňuje dosažení poslední částí neuronové sítě přeskočením některých vrstev (Wong, 2021). Tato spojení jsou následována normalizační vrstvou. Výstup každé podvrstvy je normalizován

$$\text{LayerNorm}(x + \text{Sublayer}(x)) \quad (2.3)$$

, kde $\text{Sublayer}(x)$ je funkce implementovaná samotnou podvrstvou. Veškeré podvrstvy mají pevně danou velikost výstupu a to o dimenzi $d_{\text{modelu}} = 512$ (Vaswani et al., 2017).



Obrázek 2.4: Reziduální propojení

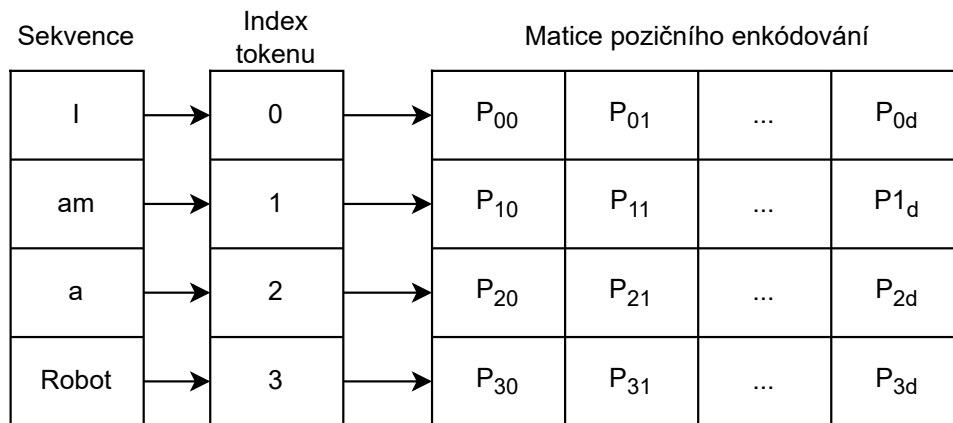
Dekodér

Dekódování obecně znamená zpětné převádění kódované zprávy do srozumitelného jazyka. V úlohách automatického zpracování jazyka to typicky znamená, že dochází k převodu číselného vektoru na text.

Stejně jako enkodér se i dekodér v původní architektuře (Vaswani et al., 2017) skládá z šesti identických vrstev. Navíc je k dvěma podvrstvám přidána třetí podvrstva, která provádí *multi-head attention* přes zásobník enkodéru. Stejně jako u enkodéru dochází k využití reziduálních spojení u každé podvrstvy následované normalizační vrstvou (Vaswani et al., 2017).

Positional encoding

Positional encoding popisuje umístění slova v sekvenci. V Transformer architektuře se využívá positional encoding schéma, které umožňuje jedinečnou reprezentaci umístění prvku v sekvenci pomocí vektorů. Výstupem tohoto encodingu je matice, kde jednotlivé řádky reprezentují enkódované tokeny vstupní sekvence spolu s informací o pozici. Tímto způsobem je nahrazena dříve používaná základní metoda indexování slov. Existuje několik důvodů, proč se nepoužívá pouze index pozice v sekvenci, pro reprezentaci samotné pozice. Například v dlouhých sekvencích mohou indexy nabývat velkých hodnot. Jedno z možných řešení je normalizace, kdy jsou indexy mapovány mezi nulu a jedničku, avšak nastává problém u sekvencí s proměnlivou délkou, jelikož by byly normalizovány odlišně (Saeed, 2023).



Matice pozičního enkódování pro sekvenci 'I am a robot'

Obrázek 2.5: Positional encoding (Saeed, 2023)

Uvažujeme positional encoding v architektuře Transformer. Předpokládáme, že máme sekvenci o délce L a potřebujeme určit pozici k – tého tokenu v sekvenci. Výpočet vypadá následovně:

$$P(k, 2i) = \sin\left(\frac{k}{n^{\frac{2i}{d}}}\right) \quad (2.4)$$

$$P(k, 2i + 1) = \cos\left(\frac{k}{n^{\frac{2i}{d}}}\right) \quad (2.5)$$

, kde k je pozice tokenu v sekvenci, d je velikost výstupního prostoru, n je skalár definovaný uživatelem (v původním článku (Vaswani et al., 2017) autoři použili hodnotu 10000). Hodnota i se používá pro mapování sloupce (Saeed, 2023).

Sekvence	Index tokenu k	Matice pozičního enkódování, kde $d=4$, $n=100$			
		i=0	i=0	i=1	i=1
I	0	$P_{00}=\sin(0)$ = 0	$P_{01}=\cos(0)$ = 1	$P_{02}=\sin(0)$ = 0	$P_{03}=\cos(0)$ = 1
am	1	$P_{10}=\sin(1/1)$ = 0.84	$P_{11}=\cos(1/1)$ = 0.54	$P_{12}=\sin(1/10)$ = 0.10	$P_{13}=\cos(1/10)$ = 1.0
a	2	$P_{20}=\sin(2/1)$ = 0.91	$P_{21}=\cos(2/1)$ = -0.42	$P_{22}=\sin(2/10)$ = 0.20	$P_{23}=\cos(2/10)$ = 0.98
Robot	3	$P_{30}=\sin(3/1)$ = 0.14	$P_{31}=\cos(3/1)$ = -0.99	$P_{32}=\sin(3/10)$ = 0.30	$P_{33}=\cos(3/10)$ = 0.96

Matice pozičního enkódování pro sekvenci 'I am a robot'

Obrázek 2.6: Positional encoding používaný v Transformer architektuře (Saeed, 2023)

Attention mechanismus

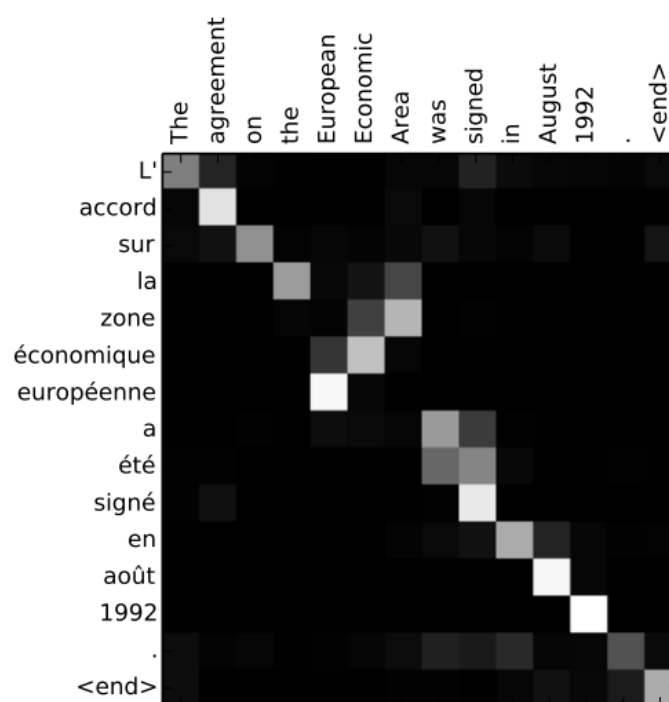
V článku (Bahdanau, Cho a Bengio, 2016), ve kterém byl Attention mechanismus představen v kontextu s překladem, použili autoři pro ukázkou následující věty:

- EN: The agreement on the European Economic Area was signed in August 1992.
- FR: L'accord sur la zone économique européenne a été signé en août 1992.

Cílem je přeložit první uvedenou větu do její ekvivalentní formy ve francouzském jazyce. Postup, kdy dochází k postupnému překládání každého slova, nebude fungovat z mnoha důvodů. Příkladem je jiné pořadí slov ve větě.

- EN: European Economic Area
- FR: la zone économique européenne

Tyto problémy řeší právě Attention mechanismus, který umožňuje textovému modelu zvažovat kontextové závislosti jednotlivých slov ve větě při rozhodování o tom, jak provést správný překlad (Markowitz, 2021). V již zmíněném článku (Bahdanau, Cho a Bengio, 2016) autoři používají pro vizualizaci Graf 2.7, kde osa x odpovídá slovům ze zdrojové věty a osa y vygenerovanému překladu. Jednotlivá pole zobrazují míru závislosti j – *tého* zdrojového slova a i – *tého* generovaného slova (světlejší barva značí větší závislost).



Obrázek 2.7: Uvažování spojitostí při překladu (Bahdanau, Cho a Bengio, 2016)

V článku (Bahdanau, Cho a Bengio, 2016) byl Attention mechanismus používán spolu s rekurentními neuronovými sítěmi. Inovací článku (Vaswani et al., 2017), ve kterém byla představena Transformer architektura, bylo oproštění od rekurentních neuronových sítí a právě proto se článek jmenuje "Attention is all you need".

Self-attention

Self-attention mechanismus je speciální typ základního Attention mechanismu. Zatímco klasický Attention spojuje vstupní a výstupní sekvenci, self-attention se zaměřuje pouze na jednu sekvenci. Tato vlastnost umožňuje modelu lépe porozumět vztahům mezi jednotlivými slovy v dané sekvenci, a tak je model schopný lépe pochopit kontextové závislosti. Díky tomuto mechanismu dochází ke zlepšení výsledků především v úlohách zpracování přirozeného jazyka (Dongen, 2022).

Jako příklad budeme uvažovat následující věty:

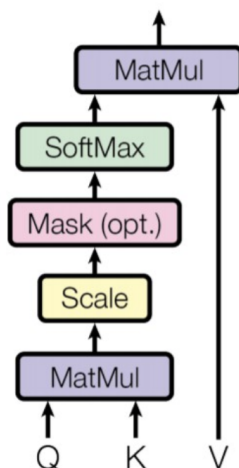
- Koruna je rozvětvená část dřevin.
- Česká koruna je oficiální měna.

Self-attention mechanismus umožní modelu rozpoznat rozdíl mezi "česká koruna" a "koruna stromu" v závislosti na kontextu.

V článku "Attention is all you need" (Vaswani et al., 2017) byl tento mechanismus autory prezentován jako "Scaled Dot-Product Attention",

kde jako vstup x uvažovali embedding vektor, který je následně přenášen váhovými maticemi W_Q , W_K , W_V , jenž jsou získány při trénovacím procesu. Výsledkem tohoto součinu jsou query matice Q , key matice K a value matice V (Dongen, 2022), (Vaswani et al., 2017). Attention funkce je definována následovně:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (2.6)$$



Obrázek 2.8: Scaled dot product attention (Vaswani et al., 2017)

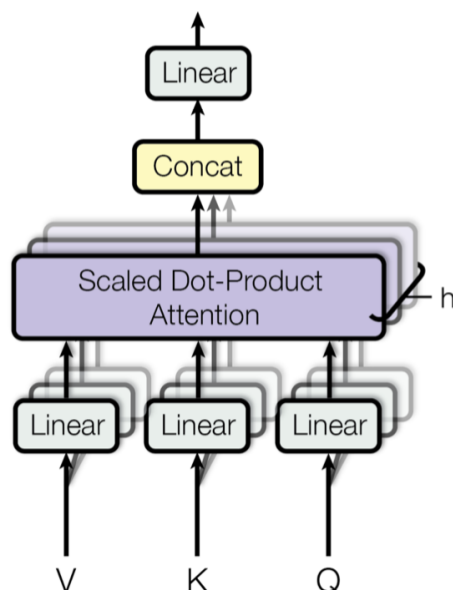
Multi-head self-attention

V Transformer architektuře je využíván takzvaný *multi-head self-attention* modul, který se skládá z tzv. *Attention heads*. Modul umožňuje počítat více Attention mechanismů paralelně. Opět uvažujeme query matici Q , key matici K a value matici V . V tomto případě je těchto matic n a každá z nich je následně samostatně vložena do jednotlivých *Attention heads*. Veškeré výpočty jsou na závěr spojeny a transformovány do finální hodnoty (Doshi, 2021)

$$MultiHead(Q, K, V) = Concat(head_1, \dots, head_h)W^O \quad (2.7)$$

$$\text{kde } head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \quad (2.8)$$

, kde matice W_i^Q , W_i^K , W_i^V , W^O obsahují naučitelné parametry. V článku (Vaswani et al., 2017) autoři uvažují osm paralelních Attention vrstev, kde každá z nich má sníženou dimenzi, a tak je celková výpočetní náročnost stejná jako u *single-head* Attention mechanismu s plnou dimenzí.



Obrázek 2.9: Multi-head self-attention (Vaswani et al., 2017)

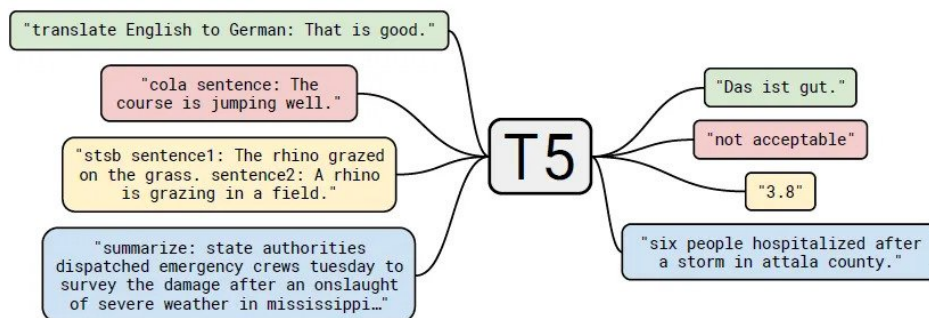
Nejpoužívanější Transformer architektury

V současné době jsou nejpoužívanější následující Transformer architektury:

- T5 (Sekce 2.2.1) - Tato architektura kombinuje úlohy porozumění přirozenému jazyku a generování přirozeného jazyka převedením na *text-to-text* úlohy. Model je předtrénován na datasetu C4, který byl vytvořen za účelem trénování T5.
- GPT (Sekce 2.2.2) - Architektura GPT je postavena na Transformer dekodéru. Předtrénování bylo provedeno pomocí predikce slova na základě předchozích. Model dosáhl skvělých výsledků v úlohách jako je například klasifikace.
- BERT (Sekce 2.2.3) - Tato architektura je založena na Transformer enkodéru. Předtrénování bylo provedeno pomocí úloh *Masked Language Modelling* a *Next Sentence Prediction*.

2.2.1 T5

Architektura T5 - Text-to-Text Transfer Transformer (Raffel et al., 2020) je verze původní architektury pro úlohy s textovým vstupem i výstupem. T5 je založený na transfer learningu, to znamená, že model je předtrénován na obrovských datech a následně je dotrénován na specifických datech pro řešenou úlohu. V článku, ve kterém byl Transformer T5 poprvé zmíněn, autoři představují dataset C4 (Colossal Clean Crawled Corpus), jenž použili pro natrénování samotného modelu. Díky tomuto datasetu dosáhli obstojných výsledků v mnoha úlohách jako například sumarizace textu, odpovídání na otázky, klasifikace textu (Raffel et al., 2020).



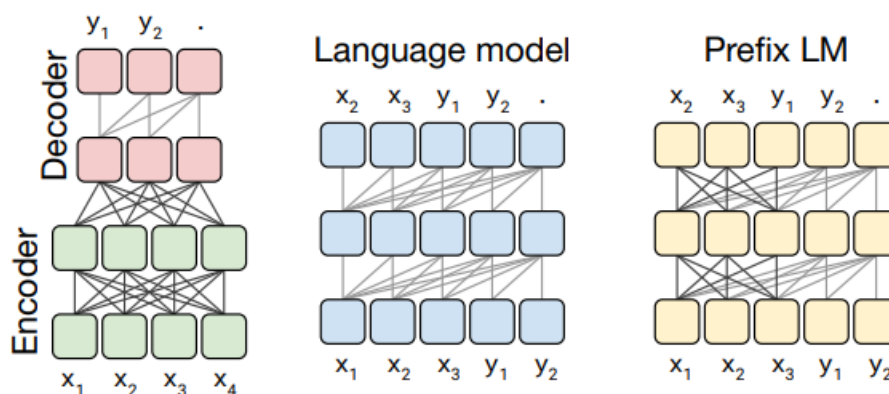
Obrázek 2.10: Text-to-Text framework (Raffel et al., 2020)

Architektura

Autoři původního článku (Raffel et al., 2020) se zaměřili na tři různé varianty architektur vhodných pro zpracovávání přirozeného jazyka.

- *Enkodér-Dekodér*
Jedná se o standardní enkodér-dekodér model využívající *fully visible* způsob v Attention mechanismu v enkodéru. Dekodér využívá kauzálního způsobu v Attention mechanismu, to znamená, že tokeny v sekvenci mohou přistupovat pouze k tokenům, které je předcházejí.
- *Jazykový model*
Jazykový model se skládá z jedné sady vrstev, kde dochází opět k využití kauzálního způsobu v Attention mechanismu.
- *Prefix jazykový model*
Prefix jazykový model je velmi podobný obyčejnému jazykovému modelu. Rozdíl je takový, že výstupy modelu se budou zaměřovat na konkrétní část vstupu, jenž obsahuje *prefix*, který může obsahovat specifické informace určující typ úlohy.

Schémata těchto tří modelů jsou znázorněna na Obrázku 2.11.



Obrázek 2.11: Jednotlivé čtverečky v diagramu reprezentují části sekvence a čáry, které je propojují, značí viditelnost v Attention mechanismu. Barvy čtverečků značí jednotlivé sady vrstev. Tmavé propojující čáry značí *fully-visible* způsob, zatím co světlé čáry značí kauzální způsob (Raffel et al., 2020).

Existuje hned několik velikostí předtrénovaných modelů T5.

Název	Počet parametrů
t5-small	60M
t5-base	220M
t5-large	770M
t5-3b	3000M
t5-11b	11000M

Tabulka 2.1: Velikosti předtrénovaných modelů T5

C4 dataset

C4 neboli Colossal Clean Crawled Corpus je dataset použitý pro předtrénování T5 pro anglický jazyk. Základem tohoto datasetu je Common Crawl¹, který obsahuje data shromažďovaná po dobu delší než dvanáct let procházením a ukládáním dat z webu. Dataset obsahuje neupravená data webových stránek, výtažek metadat a text. Z tohoto důvodu vznikl dataset C4, protože většina textu získaného z původního datasetu Common Crawl není přirozený jazyk. Majoritní část datasetu obsahuje text nepoužitelný pro trénování a nebo se jedná o šablony typu menu, chybové hlášky a nebo duplicitní text. Proto musela být vymyšlena strategie, která bude použita pro filtraci nevhodných dat pro trénování T5 (Raffel et al., 2020). V následujících bodech jsou popsány jednotlivé úpravy datasetu:

- Došlo k odstranění řádků, které nebyly řádně ukončeny. To znamená, že nebyl použit správný ukončovací interpunkční znak jako je tečka, vykřičník, otazník nebo koncová uvozovka.

¹<https://commoncrawl.org>

- Stránky, na kterých se nacházelo méně než pět vět byly smazány a zachovány byly pouze takové řádky, které obsahovaly přinejmenším tři slova.
- Stránky obsahující jakékoliv slovo ze seznamu sprostých, neslušných, obscénních slov byly také odstraněny.
- Na některých stránkách docházelo k výskytu samotného kódu a tak došlo k odstranění veškerých řádků obsahujících slovo Javascript a také byly smazány celé stránky, pokud obsahovaly složenou závorku, z důvodu jejich výskytu v mnoha programovacích jazycích.
- Pro odstranění duplicitních dat byly z datasetu odstraněny všechny tří-větné úseky, které se v datasetu vyskytly více než jednou. V tomto případě byl ponechán pouze jeden z nich.
- Z důvodu, že většina úloh, které tvůrci řešili, využívala anglický jazyk, byly vyfiltrovány veškeré stránky, které nebyly klasifikovány jako anglické.

2.2.2 GPT

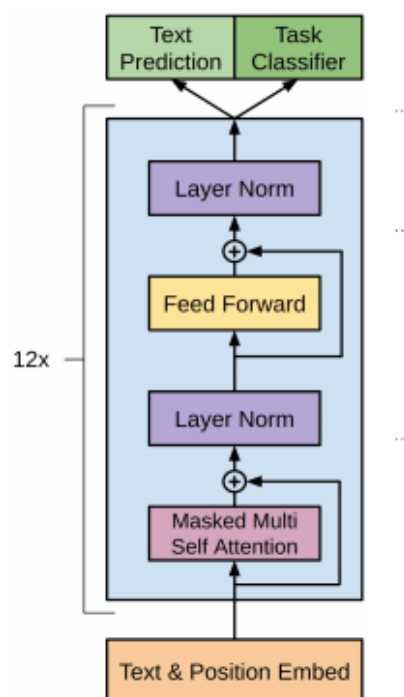
GPT neboli Generative Pre-Trained Transformer je architektura využívaná pro trénování jazykových úloh. GPT architektura je založena na struktuře Transformer. První zmínka o této metodě byla v článku "Improving Language Understanding by Generative Pre-Training" (Radford a Narasimhan, 2018). Koncept GPT je založen na dvou fázích. První fáze je předtrénování jazykového modelu na velkém množství dat z důvodu naučení počátečních parametrů. Následně dojde k dotrénování, kdy jsou parametry přizpůsobeny určité úloze pomocí správně zvolených dat.

Architektura

GPT používá architekturu Transformer dekodéru na rozdíl od architektury BERT (Sekce 2.2.3), která využívá pouze enkodér. Získává tak autoregresivní povahu díky dekodéru, protože k předpovědi dalšího tokenu používá pouze předchozí tokeny ze sekvence (Jagtap, 2020). Existuje mnoho velikostí GPT modelů. Pro demonstraci velikosti jsme vybrali tyto základní (ALI, 2023):

Název	Počet parametrů	Trénovací data	Max. délka sekvence
GPT-1	117M	Common Crawl, BookCorpus	1024
GPT-2	1500M	Common Crawl, BookCorpus, WebText	2048
GPT-3	175000M	Common Crawl, BookCorpus, Wikipedia, knihy, články, ...	4096
GPT-4	neznámo	neznámo	neznámo

Tabulka 2.2: Porovnání velikostí GPT modelů



Obrázek 2.12: Architekturu modelu GPT, která se skládá z dvanácti vrstev Transformer dekodérů se 117M parametry. Tato architektura byla použita autory v článku "Improving Language Understanding by Generative Pre-Training" (Radford a Narasimhan, 2018)

Předtrénování bez učitele

Unsupervised pre-training využívá algoritmy strojového učení k analýze a shlukování neanotovaných dat. Tyto algoritmy naleznou skryté souvislosti/vzory v datech bez nutnosti lidského zásahu.

Pro předtrénování je poskytnut dataset tokenů $U = u_1, \dots, u_n$. Cílem GPT modelu je pokusit se maximalizovat následující funkci:

$$L_1(U) = \sum_i \log P(u_i | u_{i-k}, \dots, u_{i-1}; \theta) \quad (2.9)$$

, kde k je velikost kontextového okna a P je podmíněná pravděpodobnost, která je modelována pomocí neuronové sítě s parametry θ . Tyto parametry jsou trénovány pomocí *gradient descent* algoritmu. Cílem je maximalizovat pravděpodobnost, že se token u_i objeví v daném kontextu vzhledem k tokenům u_{i-k}, \dots, u_{i-1} . Pomocí následujících rovnic dojde k výpočtu pravděpodobnosti a Attention:

$$h_0 = UW_e + W_p \quad (2.10)$$

$$h_l = \text{transformer_block}(h_{l-1} \forall i \in [1, n]) \quad (2.11)$$

$$P(u) = \text{softmax}(h_n W_e^T) \quad (2.12)$$

, kde $U = (u_{-k}, \dots, u_{-1})$ je kontextový vektor tokenů, n je počet vrstev, W_e je token embedding matice a W_p je position embedding matice (Radford a Narasimhan, 2018), (Jagtap, 2020).

Dotrénování s učitelem

Supervised fine-tuning je přístup strojového učení, jenž je založený na používání anotovaných datasetů.

V této části uvažujeme anotovaný dataset C , ve kterém je každá instance složena ze sekvence vstupních tokenů x^1, \dots, x^m spolu s třídou y a naším cílem je maximalizovat funkci:

$$L_2(C) = \sum_{(x,y)} \log P(y | x^1, \dots, x^m) \quad (2.13)$$

, to znamená, že se snažíme maximalizovat logaritmickou pravděpodobnost třídy y vzhledem k tokenům x_1, \dots, x_m .

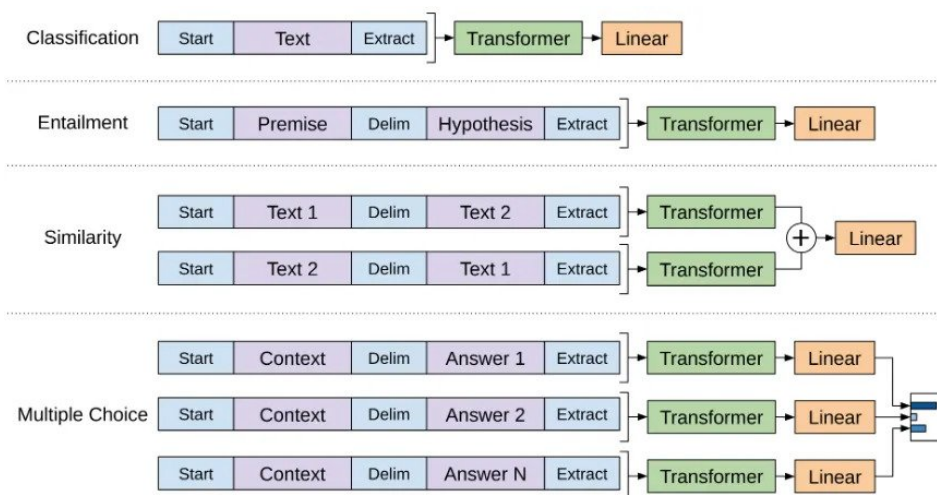
$$P(y | x^1, \dots, x^m) = \text{softmax}(h_m^l W_y) \quad (2.14)$$

, kde h_m^l je aktivační hodnota finálního bloku, která je získána díky předání vstupních tokenů do předtrénovaného modelu a W_y jsou parametry lineární výstupní vrstvy. Kombinací funkce $L_1(C)$, kterou jsme zmínili u předtrénování, a funkce $L_2(C)$ dostáváme funkci:

$$L_3(C) = L_2(C) + \lambda * L_1(C) \quad (2.15)$$

, kde $L_1(C)$ je Rovnice 2.9 použita pro dataset C a λ je váha, pomocí které lze funkci optimalizovat (Radford a Narasimhan, 2018), (Jagtap, 2020).

Pokud celé dotrénování shrneme, můžeme říct, že vše závisí na vstupu, který musí být upraven v závislosti na řešené úloze. Obrázek 2.13 zobrazuje možné transformace vstupu dle úlohy.



Obrázek 2.13: Úpravy vstupu pro dotrénování v závislosti na cílové úloze. (Radford a Narasimhan, 2018)

Specifické úlohy

- Textová implikace**

U úlohy tohoto typu dojde ke spojení textu a hypotézy a následnému rozdělení pomocí oddělovacího tokenu. Autoři článku zvolili token \$ (Radford a Narasimhan, 2018).
- Textová podobnost**

Pro úlohy, ve kterých se řeší textová podobnost, neexistuje pevně dané pořadí dvou porovnávaných vět. Aby se tato skutečnost zohlednila, vstupní sekvence je upravena tak, aby obsahovala obě dvě možná pořadí. Následně dojde ke zpracování obou možností nezávisle na sobě, z důvodu vzniku dvou reprezentací jedné sekvence, které jsou před vstupem do lineární vstupní vrstvy sečteny po prvcích (Radford a Narasimhan, 2018).
- Odpovídání na otázky a uvažování pomocí zdravého rozumu²**

Při řešení úloh tohoto typu je potřeba kontext, ve kterém se nachází možná odpověď na danou otázku. Dále je dobré mít k dispozici pár možných odpovědí z daného kontextu. Pokud máme všechny tyto prerekvizity, dojde ke spojení kontextu z , otázky q a množiny odpovědí a_k . Mezi jednotlivé možnosti je vložen rozdělovací token \$. Výsledek vypadá takto $[z; q; \$; a_k]$. Všechny takto vzniklé struktury jsou vloženy do Transformer dekodéru nezávisle na sobě. Následně jsou normalizovány pomocí Softmax vrstvy, kde vznikne možné rozložení výstupů přes možné odpovědi (Radford a Narasimhan, 2018).

²Common sense reasoning - schopnost uvažovat jako člověk na základě zkušeností, znalostí o světě a sociálních interakcí

2.2.3 BERT

Model typu BERT (Bidirectional Encoder Representations from Transformers) byl představen v článku "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" (Devlin et al., 2019). BERT je navržen k předtrénování hluboké obousměrné reprezentace z neanotovaného textu společným podmíněním jak levého, tak i pravého kontextu ve všech vrstvách. Výsledkem je předtrénovaný model, který lze dotrénovat pouze jednou výstupní vrstvou, díky čemuž přichází možnost natrénovat modely pro široké využití, aniž by bylo nutné provádět zásadní změny v architektuře. Například lze natrénovat modely pro zodpovídání otázek a jazykové odvozování.

Architektura

Architektura BERT modelu je založena na originální implementaci vícevrstvého obousměrného Transformeru s tím rozdílem, že BERT využívá pouze enkodér. BERT lze popsat pomocí počtu Transformer bloků L , velikosti skryté vrstvy H a díky počtu *self-attention heads* A . V původním článku byly představeny výsledky na dvou velikostech modelu.

- $BERT_{BASE}(L = 12, H = 768, A = 12, \text{počet parametrů} = 110M)$
- $BERT_{LARGE}(L = 24, H = 1024, A = 16, \text{počet parametrů} = 340M)$

Velikost modelu $BERT_{BASE}$ byla zvolena tak, aby byla ekvivalentní s velikostí modelu GPT od OpenAI pro srovnávací účely. Rozdíl mezi nimi je takový, že BERT Transformer používá obousměrný *self-attention mechanismus*, zatímco GPT Transformer využívá omezený *self-attention*, kde každý token může přistupovat pouze ke kontextu na levé straně (Devlin et al., 2019).

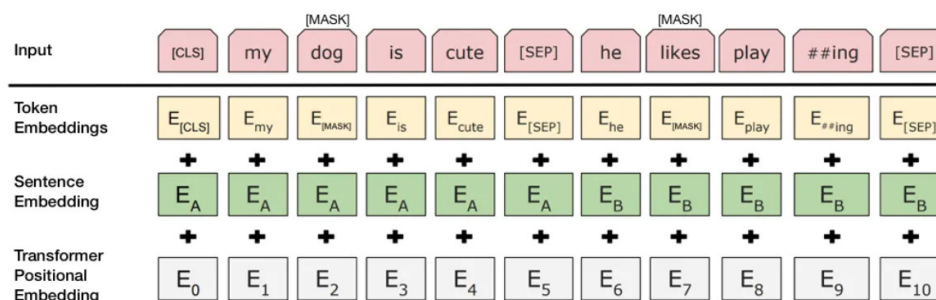
Vstupní data

Aby BERT dokázal řešit různé typy úloh, je nutné připravit vstupní data. V tomto modelu je využíván WordPiece algoritmus pro segmentaci slov se slovníkem o velikosti 30K tokenů. Wordpiece algoritmus je metoda pro tokenizaci textu na menší jednotky, které se nazývají subwordy (Y. Wu et al., 2016). Prvním tokenem každé sekvence je speciální klasifikační token $[CLS]$. Konečný skrytý stav odpovídající tomuto tokenu se použije jako souhrnná reprezentace sekvence pro klasifikaci. Dvojice vět jsou sjednoceny do jedné sekvence. Věty jsou následně v sekvenci rozlišovány dvěma způsoby. První způsobem je použití speciálního token $[SEP]$, který věty rozděljuje. Druhý způsob označí každý token ve větě štítkem, který nám říká, do jaké věty token patří. Na závěr jsou tokeny očíslovány indexy, které stanovují pozice tokenů v sekvenci (Devlin et al., 2019). Tento postup můžeme snadno shrnout do tří bodů (Horev, 2018):

- Dojde ke vložení $[CLS]$ tokenu na začátek sekvence a $[SEP]$ token je vložen mezi jednotlivé věty.

- Při mapování věty jsou použity štítky určující, do které věty patří jednotlivé tokeny.
- Každý token je označen číslem, jenž stanovuje pozici tokenu v sekvenci.

Na závěr dojde k sjednocení těchto tří zmíněných úprav. Výsledek je použit jako vstup pro BERT model.



Obrázek 2.14: Vstupní data pro model BERT (Devlin et al., 2019)

Předtrénování modelu

Model je předtrénován pomocí dvou neřízených úloh. Jednou z úloh je *Masked Language Modeling (MLM)*, to znamená, že dojde k nahrazení určitého procenta vstupních tokenů maskou *[MASK]* a následně dochází ke zpětné predikci těchto tokenů na základě kontextu ostatních tokenů ve větě. Tato metoda umožňuje získat předtrénovaný model, ale dochází k nesrovnalostem mezi předtrénováním a dotrénováním, protože *[MASK]* token není používán při dotrénování. Aby se tento problém zmírnil, ne vždy jsou tokeny, které mají být maskovány, přepsány tokenem *[MASK]*. Generátor trénovacích dat náhodně vybere 15% tokenů pro predikci. Z těchto tokenů je 80% nahrazeno tokenem *[MASK]*, 10% je nahrazeno jiným náhodným tokenem a u 10% nedojde ke změně.

Druhá úloha predikuje návaznost vět z důvodu pochopení vztahů mezi jednotlivými větami, protože mnoho úloh, jako například odpovídání na otázku, nebo generování textu, je postaveno právě na tomto kontextu. Pro předtrénování jsou data vybrána tak, že 50% dvojic na sebe navazuje a zbylých 50% je zvoleno tak, že druhá věta do dvojice je vybrána náhodně z veškerých dat. V původním článku je uvedeno, že finální model návaznosti dosahuje přesnosti 97-98%.

Data použitá pro předtrénování jsou získána z datasetu BooksCorpus, který obsahuje 800M slov. Dále je využita anglická wikipedia, ze které jsou vytaženy pouze textové odstavce, tzn. jsou vynechané tabulky, nadpisy a podobné struktury. Po této úpravě wikipedie poskytne okolo 2500M slov (Devlin et al., 2019).

Dotrénování modelu

Model může být použit na rozsáhlou škálu úloh založených na práci s jazykem. Většina úloh je řešena tak, že je přidána pouze jedna vrstva. Pro klasifikační úlohy je přidána klasifikační vrstva na výstup transformera. V úloze odpovídání na otázky model dostane text, ve kterém lze najít odpověď na danou otázku. Úkolem je vyznačit v sekvenci odpověď. Tento model může být natrénován pomocí naučení dvou dalších vektorů, které označí začátek a konec odpovědi. Při dotrénovacím procesu zůstává většina parametrů modelu stejných jako při předtrénování (Horev, 2018). Dotrénování je relativně nenáročné v porovnání s předtrénováním. Veškeré pokusy prováděné v původním článku (Devlin et al., 2019) trvaly maximálně jednu hodinu dotrénování na TPU nebo pár hodin na GPU.

2.3 Automatický překlad

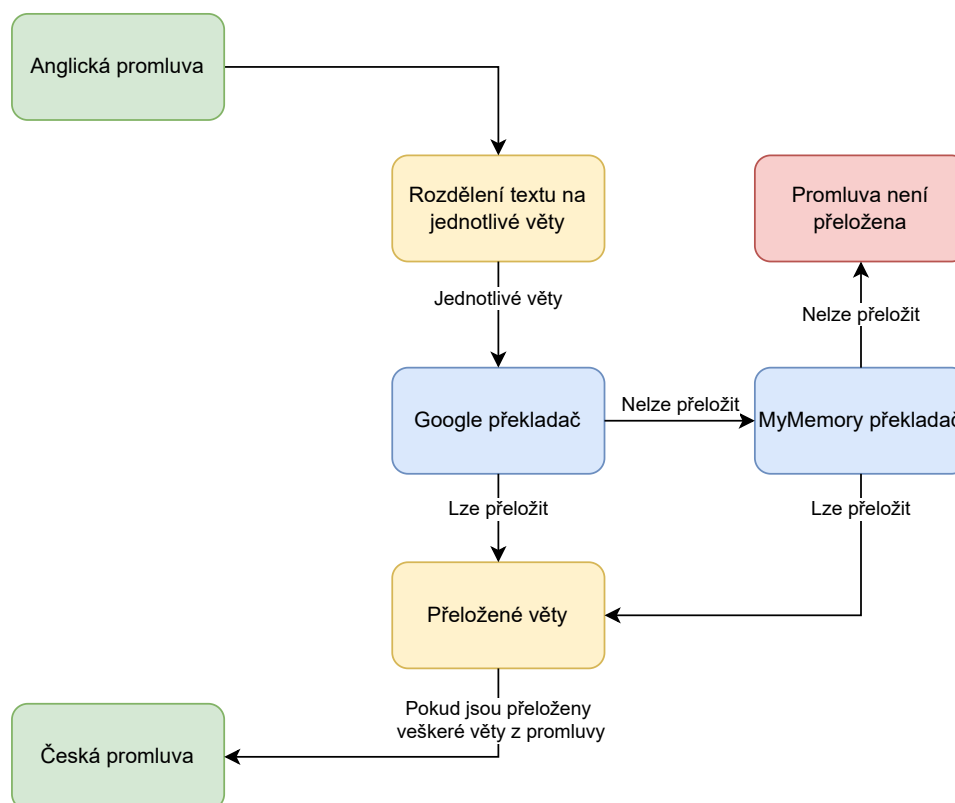
Pro překlad jsme použili Python knihovnu `deep-translator`³, kterou jsme vybrali z důvodu možnosti výběru z více frameworků od různých poskytovatelů. Veškeré frameworky jsme otestovali a pro naši práci jsme vybrali Google překladač a MyMemory překladač. Velkým plusem těchto překladačů je jejich bezplatné neomezené používání. Jediné omezení je, že lze překládat text o maximální délce 5K znaků.

2.3.1 Algoritmus překladu

Pro usnadnění překladu velkého množství dat byl vytvořen algoritmus, který na vstupu vyžaduje soubor s příponou `.tsv`, ve kterém jsou data uložena do sloupců. Před spuštěním musíme určit pořadové číslo sloupce obsahujícího text. Dále zvolíme jazyk vstupního textu a jazyk, do kterého chceme text přeložit. Posledním parametrem je uspořádání sloupců do výstupního souboru. Po spuštění algoritmu dojde k načtení souboru, ze kterého jsou data čtena po řádcích. Pomocí NLTK tokenizeru (Bird a Klein, 2009) dojde k rozdělení textu na jednotlivé věty z důvodu možnosti výskytu dlouhých promluv přesahujících 5000 znaků. Následně jsou věty jednotlivě překládány pomocí Google překladače. V případě, že při překladu dojde k chybě (nejčastěji nastane chyba `TranslationNotFound`, která říká, že text nelze přeložit), použije se překladač MyMemory. Pokud i pomocí tohoto překladače z nějakého důvodu není proveden překlad, promluva je přeskočena a dochází k překladu další promluvy. Jakmile je přeložena celá promluva, uloží se do python listu. Ve chvíli, kdy list přesahuje předem daný počet elementů, data jsou uložena do souboru s příponou `.tsv` a list se vyprázdní. Tato funkce je dobrá z důvodu průběžného ukládání, tudíž se v případě problému vyhneme ztrátě veškerých dosavadně přeložených dat a zároveň zabránujeme přetečení paměti

³<https://pypi.org/project/deep-translator/>

v zařízení, na kterém překlad probíhá. V následujícím diagramu je popsán algoritmus pro přehlednost (Obrázek 2.15).



Obrázek 2.15: Algoritmus překladu je znázorněn pro jednu promluvu. V našem případě dochází k opakování vyobrazeného algoritmu, dokud nejsou použity všechny promluvy ze vstupního souboru.

2.4 Augmentace dat

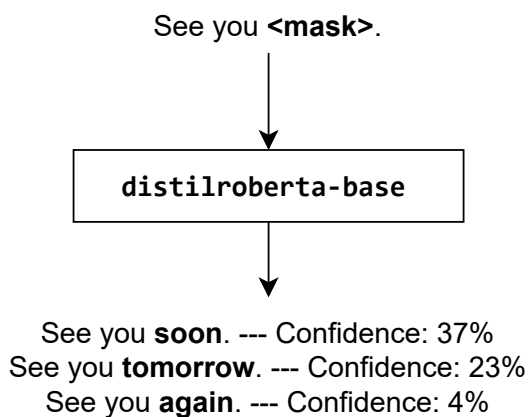
Augmentace je využívána k získání nových dat pomocí úprav již existujících dat. Existuje mnoho způsobů, jak k augmentaci přistupovat a kterou zvolit. Vždy záleží na dané úloze, kterou řešíme, a na datech, které máme k dispozici. Pro naši úlohu jsme vybrali dva způsoby augmentace, pomocí kterých byl rozšířen náš původní dataset (více v Kapitole 4).

2.4.1 Augmentace pomocí synonym

Augmentace pomocí synonym je založena na Transformer modelu RoBERTa veřejně dostupného na stránce *Huggingface*⁴. Tento model je předtrénován na velkém množství neanotovaných dat pomocí metody masked language modeling (MLM) již zmíněné v Sekci 2.2.3. Samotný model je popsán v článku "RoBERTa: A Robustly Optimized BERT Pre-training Approach" (Liu et al., 2019). Vstupem pro predikci je text, ve kterém nahradíme jedno ze slov maskovacím tokenem na základě kterého je predikováno slovo na výstupu. Výstupem jsou pak slova, kterými

⁴<https://huggingface.co/distilroberta-base>

lze nahradit maskovací token. Spolu se slovy dostáváme také skóre na základě kterého můžeme usoudit, zda se slovo do dané věty hodí. Nevhodná slova můžeme poznat podle velkého rozdílu v hodnotách.

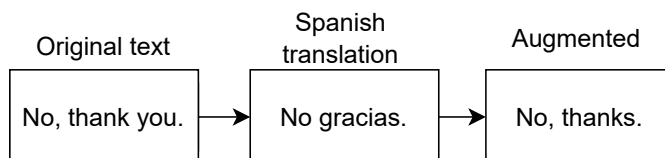


Obrázek 2.16: Augmentace pomocí synonym

V ukázce na Obrázku 2.16 bychom touto metodou augmentace vygenerovali dva nové vzorky "See you soon." a "See you tomorrow.". Třetí vzorek "See you again." by mohl být pomocí prahu vyfiltrován.

2.4.2 Augmentace pomocí překladů

Augmentace pomocí překladů je metoda, která se používá pro získání nových dat pomocí překládání jednotlivých sekvencí. Tento postup spočívá v přeložení vstupní sekvence do jiného jazyka a následném přeložení zpět do původního jazyka. Díky tomuto postupu je možno získat text s podobným významem, ale jinak formulovaný. Avšak je nutno počítat s problémy, které mohou nastat, jako například chyby v překladu nebo změna sémantického významu.



Obrázek 2.17: Augmentace pomocí překladu

Kapitola 3

Data

3.1 Dialogové datasety

V naší práci jsme využili celkem osm datasetů, které obsahují dialogy. Klíčovým prvkem pro výběr datasetů byly typy anotací, jež jsme mohli využít pro námi zvolené úlohy. Důležité byly informace o změně řečníka v dialogu a o místě přechodu z jednoho dialogu na druhý. Další poskytnuté anotace k jednotlivým promluvám, které jsme byli schopni využít jsou například údaje o emoci a sentimentu při promluvě. Pro zjednodušení práce bylo důležité sjednotit formát všech datasetů, proto jsme vyfiltrovali nepotřebné anotace a převedli jsme data do formátu *TSV*. Na závěr došlo k překladu veškerých promluv z anglického jazyka do českého jazyka.

3.1.1 Commonsense-Dialogues Dataset (d01)

Commonsense-Dialogues Dataset (Zhou et al., 2021) se skládá z 11 220 dialogů, které v součtu obsahují 26 424 promluv. Dataset je založený na sociálních kontextech, které byly získány ze souboru SocialIQA (Sap et al., 2019). Jedná se o měřítko rozumového uvažování založené na více možnostech výběru otázek a odpovědí. V závislosti na těchto kontextech byly napsány dialogy o čtyřech až šesti promluvách mezi dvěma osobami. V datasetu je u každého dialogu poznamenán kontext, na základě kterého byl psán a hlavní řečník.

```
1     "7": {
2         "context": "Kai climbed to the top of the mountain.
3                 It took a long time and a lot of effort.",
4         "speaker": "Kai",
5         "turns": [
6             "I am to the top of the mountain.",
7             "You finally did it.",
8             "It took hours to get here.",
9             "And a lot of effort, I bet.",
10            "Yeah with a lot of effort.",
11            "That's so good, happy for you."
12        ]
13    }
```

Zpracovaný formát dat pro integraci do finálního dialogového korpusu je ilustrován v Tabulce 3.1.

dial ID	en text	cs text	řečník
1	I got so mad, I couldn't contain it anymore.	Byl jsem tak naštvaný, že už jsem to nemohl udržet.	A
1	Did you huff off?	Nadechl ses?	B
1	I did, I flared up into anger.	Udělal jsem, vzplanul jsem vztekem.	A
...

Tabulka 3.1: Vygenerovaný (.tsv) formát dialogových dat z datasetu Commonsense-dialogues

3.1.2 AnnoMI (d02)

AnnoMI (Z. Wu et al., 2022) je dataset obsahující 133 dialogů mezi terapeutem a jeho klientem. Rozhovory byly natočeny a profesionály zpracovány do textové podoby. Ke každé promluvě je přiřazeno ID dialogu a samotné věty. Dále je ke každému dialogu poskytnuta adresa, na které se nachází video z terapie. Samotné promluvy jsou dále anotovány chováním terapeuta a klienta, probíraným tématem a časem, kdy byla promluva řečena v nahrávce.

Zpracovaný formát dat pro integraci do finálního dialogového korpusu je ilustrován v Tabulce 3.2.

dial ID	řečník	en text	chování terapeuta	chování klienta	cs text
4	therapist	It's been a while.	other	n/a	Už je to nějaký čas.
4	client	Yeah, it's been a while.	n/a	neutral	Jo, už je to nějaký čas.
4	therapist	This is something new.	reflection	n/a	To je něco nového.
...

Tabulka 3.2: Vygenerovaný (.tsv) formát dialogových dat z datasetu AnnoMI

Původní formát datasetu je téměř identický jako vygenerovaný (.tsv) soubor. Jediný rozdíl je ten, že jsme z datasetu odebrali informace o nahrávce terapie.

3.1.3 DailyTalk (d03)

Tento dataset byl představen v (Lee, Park a Kim, 2022) jako kvalitní řečový dataset založený na dialozích pro Text-to-Speech úlohy. Jako základ byl zvolen dataset DailyDialog (Li et al., 2017), ze kterého byly vybrány dialogy a následně zaznamenány. Pro výběr dialogů autoři zvolili dvě základní kritéria:

- Vybrané dialogy se skládají z více než pěti promluv, protože z krátkých dialogů nemusí být jasný kontext.
- Došlo k úpravě postav vystupujících v dialozích. Byly vybrány některé dialogy, ve kterých vystupovali dva řečníci stejného pohlaví. V těchto dialozích došlo ke změně jednoho řečníka na opačné pohlaví.

Celkem se v datasetu nachází 23772 promluv, které tvoří 2541 dialogů. Každá promluva je anotována emocí, druhem věty.

```

1     "0": {
2         "0": {
3             "index": "0-0",
4             "turn": 12,
5             "topic": 1,
6             "emotion": "no emotion",
7             "act": "question",
8             "speaker": 1,
9             "text": "What are you working on?",
10            "dialog_idx": 0,
11            "utterance_idx": 0
12        },
13        "1": {
14            ...
15        }
16    }

```

Zpracovaný formát dat pro integraci do finálního dialogového korpusu je ilustrován v Tabulce 3.3.

dial ID	druh věty	emoce	en text	cs text	řečník
7-6	commissive	no emotion	OK. I will be there.	OK. Budu tam.	A
7-7	commissive	happiness	Thank you for calling. Goodbye.	Děkuji za zavolání. Ahoj.	B
7-8	inform	happiness	Bye.	Sbohem.	A

Tabulka 3.3: Vygenerovaný (.tsv) formát dialogových dat z datasetu DailyTalk

3.1.4 MELD (d04)

Multimodal EmotionLines Dataset (Poria et al., 2019) obsahuje 1433 dialogů ze seriálu Friends, které se skládají z 13711 promluv. Veškeré promluvy jsou anotovány druhem emoce, sentimentu a jménem řečníka. Dále jsou k dispozici časové údaje umístění promluvy v nahrávce a informace o sérii a dílu, kde se promluva nachází. Tento dataset se liší od ostatních tím, že se v jednotlivých dialogích může vyskytovat více řečníků. Dataset je ve formátu .csv (comma-separated values).

en text	řečník	emoce	sentiment	série	epizoda	...
Hey!	Phoebe	joy	positive	6	4	...
Hey-Hey- Hey!	Joey	joy	positive	6	4	...
What are you doing?	Chandler	neutral	neutral	6	4	...
...

Tabulka 3.4: Formát původních dat datasetu MELD

Zpracovaný formát dat pro integraci do finálního dialogového korpusu je ilustrován v Tabulce 3.5.

en text	řečník	emoce	sentiment	dial ID	cs text
I know.	Rachel	neutral	neutral	25	Vím.
Ow!!	Rachel	anger	negative	25	Ou!!
Oh god!	Monica	surprise	negative	25	Pane bože!
...

Tabulka 3.5: Vygenerovaný (.tsv) formát dialogových dat z datasetu MELD

3.1.5 SGD (d05)

The Schema-Guided Dialogue (SGD) dataset obsahuje 22825 dialogů mezi člověkem a virtuálním asistentem. Celkem dostáváme 463256 promluv. Jednotlivé konverzace jsou o různých tématech jako je například cestování, počasí, kalendáři. Dataset obsahuje širokou škálu anotací, které se dají využít v mnoha odvětvích.

```

1   "88": {
2     "dialog_id": "4_00088",
3     "services": [
4       "0": "Buses_3"
5     ],
6     "turns": [
7       "2": {
8         "frames": [],
9         "speaker": "USER",
10        "utterance": "I am leaving San Francisco for Fresno."

```

```

11     },
12     "3": {
13         "frames": [],
14         "speaker": "SYSTEM",
15         "utterance": "When would you like to leave?"
16     }
17 ]
18 }

```

Zpracovaný formát dat pro integraci do finálního dialogového korpusu je ilustrován v Tabulce 3.6.

dial ID	en text	řečník	cs text
5	Will I reserve you the house?	SYSTEM	Zarezervuji vám dům?
5	Not now, thanks for your help	USER	Ted' ne, děkuji za pomoc.
5	You are welcome.	SYSTEM	Nemáš zač.
...

Tabulka 3.6: Vygenerovaný (.tsv) formát dialogových dat z datasetu SGD

3.1.6 Harry Potter Movies Dataset (d06)

Součástí tohoto datasetu¹ je několik souborů obsahující dialogy, postavy, místa, filmy, kouzla a kapitoly. Pro naši práci využijeme pouze soubor s dialogy a s informacemi o postavách, ze kterého jsme schopni pomocí id řečníka, určit jeho jméno. Dataset celkem obsahuje 7442 promluv v 234 dialozích.

dial ID	chap ID	place ID	char ID	dialogue
18	2	8	31	Wake up, cousin! We're going to the zoo!
19	2	8	32	Oh, here he comes, the birthday boy!
20	2	8	21	Happy birthday, son.
...

Tabulka 3.7: Formát původních dat datasetu Harry Potter

Zpracovaný formát dat pro integraci do finálního dialogového korpusu je ilustrován v Tabulce 3.8.

¹<https://www.kaggle.com/datasets/maricinnamon/harry-potter-movies-dataset>

řečník	en text	dial ID	cs text
Dudley Dursley	Wake up, cousin! We're going to the zoo!	2	Probud' se, bratranče! Jdeme do zoo!
Petunia Dursley	Oh, here he comes, the birthday boy!	2	Oh, přichází oslaveneček!
Vernon Dursley	Happy birthday, son.	2	Všechno nejlepší, synu.
...

Tabulka 3.8: Vygenerovaný (.tsv) formát dialogových dat z datasetu Harry Potter

3.1.7 Simpsons (d07)

Tento dataset² obsahuje 131841 promluv získaných ze seriálu Simpsons. Celkem dostáváme 17813 dialogů, ve kterých může vystupovat i více řečníků. V Tabulce 3.9 můžeme vidět, že anotací se zde vyskytuje minimum, ale pro naše účely jsou postačující, jelikož jsme podle mezer mezi větami schopni rozpoznat začátek nového dialogu..

řečník	en text
John	Queen of the Harpies!
Gloria	No, I'm not!
John	Here's your crown, your Majesty.
...	...

Tabulka 3.9: Formát původních dat datasetu Simpsons

Zpracovaný formát dat pro integraci do finálního dialogového korpusu je ilustrován v Tabulce 3.10.

řečník	en text	dial ID	cs text
John	Queen of the Harpies!	23	Královna Harpyjí!
Gloria	No, I'm not!	23	Ne nejsem!
John	Here's your crown, your Majesty.	23	Tady je vaše koruna, Vaše veličenstvo.
...

Tabulka 3.10: Vygenerovaný (.tsv) formát dialogových dat z datasetu Simpsons

3.1.8 The Office (d08)

Jedná se o dataset³ získaný z přepisu dialogů seriálu The Office do textové podoby. Dohromady soubor obsahuje 8157 dialogů o celkově 54626

²<https://www.kaggle.com/datasets/pierremegret/dialogue-lines-of-the-simpsons>

³<https://www.kaggle.com/datasets/nasirkhalid24/the-office-us-complete-dialoguetranscript>

promluvách. Jednotlivé promluvy jsou anotovány řečníkem a informacemi o tom, v jakém dílu seriálu se promluva odehrála.

série	epizoda	název	scéna	řečník	en text
1	1	Pilot	3	Pam	What?
1	1	Pilot	3	Michael	Any messages?
1	1	Pilot	3	Pam	Uh, yeah. Just a fax.
...

Tabulka 3.11: Formát původních dat datasetu The Office

Zpracovaný formát dat pro integraci do finálního dialogového korpusu je ilustrován v Tabulce 3.12.

dial ID	řečník	en text	cs text
3	Pam	What?	Co?
3	Michael	Any messages?	Nějaké zprávy?
3	Pam	Uh, yeah. Just a fax.	Ano. Jen fax.
...

Tabulka 3.12: Vygenerovaný (.tsv) formát dialogových dat z datasetu The Office

3.2 Zpracovaný dialogový korpus

Po úpravě veškerých datasetů jsme se rozhodli udělat jeden velký zpracovaný dialogový korpus. Jedná se o sjednocení veškerých souborů, kde jsou veškeré promluvy anotovány identifikačním číslem datasetu, identifikačním číslem dialogu a v každém dialogu jsou očíslované jednotlivé promluvy tak, aby šlo poznat jejich pořadí. Poslední uvedenou informací je změna řečníka. Celkově se tak dostáváme na 55589 dialogů a 727631 promluv. Struktura datasetu je uvedena v Tabulce 3.13.

ID dataset	ID dial	ID věty	řečník	en text	cs text
d01_000131	24	2	<A>	That must have been fun.	To musela být legrace.
d01_000132	24	3		It sure was.	To jistě byla.
d01_000133	24	4	<A>	What was the prank?	Co to bylo za žert?
...

Tabulka 3.13: Zpracovaný dialogový korpus pro trénování konverzačního modelu

Kapitola 4

Experimenty a výsledky

4.1 Architektura testovaného modelu

Pro náš výzkum jsme použili Transformer architekturu T5 blíže popsanou v Sekci 2.2.1. Experimenty, ve kterých nebyla použita augmentace dat, jsou prováděny jak v českém, tak i anglickém jazyce. V části, kde využíváme anglický jazyk, byl použit předtrénovaný model *t5-base*. U experimentů s českým jazykem jsme použili model, který byl předtrénován pouze na české části *Common Crawl*¹ datasetu.

4.2 Rozdělení dat

Veškeré experimenty jsou založeny na zpracovaném dialogovém korpusu, který jsme popisovali v Sekci 3.2. Cílem bylo pomocí úprav trénovacích dat natrénovat model, který bude fungovat jako chatovací bot založený na dialozích z běžného života. Jednotlivé úpravy byly zvoleny tak, aby se neuronová síť naučila kontextové závislosti mezi větami a byla schopna rozlišovat řečníky a jednotlivé promluvy.

Nejprve jsme celý dataset rozdělili na trénovací, testovací a validační část v následujícím poměru (Tabulka 4.1):

train	test	val
90%	5%	5%

Tabulka 4.1: Rozdělení dat

Jednotlivá procenta určují počet dialogů z jednotlivých datasetů, které jsou obsaženy ve zpracovaném dialogovém korpusu (Sekce 3.2). To znamená, že například z prvního datasetu (Sekce 3.1.1), jsme vzali 90% dialogů a uložili jsme je do souboru s trénovacími daty. Analogicky byly rozděleny dialogy i v ostatních datasetech. Počet promluv obsažených v množinách je závislý na počtu obrátů v samotných dialozích (Tabulka 4.2).

¹<https://huggingface.co/fav-kky/FERNET-C5>

train	test	val
654 952	36 887	35 785

Tabulka 4.2: Počet promluv v jednotlivých množinách datasetu - trénovací, testovací, validační

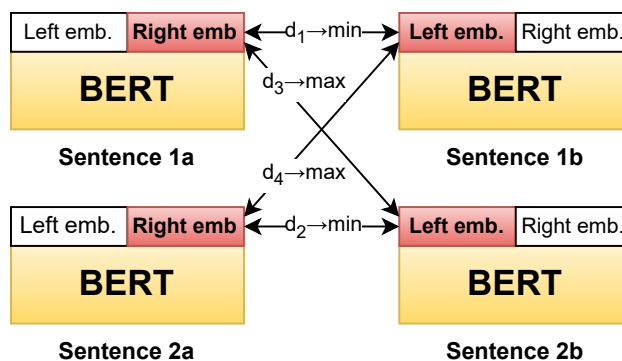
Důvod, proč jsme dataset nerozdělovali podle promluv, je takový, že jsme chtěli, aby byly dialogy v celku a nedošlo k jejich rozdělení. Příčinou, proč jsme nerozdělovali dataset jako celek, ale rozdělovali jsme jednotlivé původní datasety, je zohlednění různého procentuálního zastoupení jednotlivých datasetů ve zpracovaném dialogovém korpusu. Například dataset MELD (Sekce 3.1.4) obsahuje 1433 dialogů, zatímco dataset SGD (Sekce 3.1.5) obsahuje 22825 dialogů. Chtěli jsme, aby naše tři množiny dat obsahovaly adekvátní poměr dat z dílčích datasetů.

Trénovací množina je jednotná pro všechny experimenty z důvodu přesnějšího porovnání finálních výsledků. Jediným rozdílem je finální příprava dat pro trénování T5 modelu, která se liší u každého experimentu.

4.3 Vyhodnocovací technika

K vyhodnocení veškerých experimentů jsme použili model sémantické návaznosti představený v článku (Švec et al., 2023), který je založen na architektuře BERT (Sekce 2.2.3). Cílem tohoto modelu je určit, zda jedna věta navazuje sémanticky na druhou.

Pro trénování modelu byla použita sada textových dat, ze které byly náhodně vybírány dvě věty. Tyto věty byly následně rozděleny na čtyři části ($1a$, $1b$, $2a$, $2b$), kde část b navazuje na část a . Samotná síť je pak trénována tak, aby minimalizovala vzdálenosti mezi páry (d_1 , $d_2 \rightarrow \min$), jenž na sebe sémanticky navazují, a aby maximalizovala vzdálenosti mezi páry (d_3 , $d_4 \rightarrow \max$), které na sebe sémanticky nenavazují. Interpretace těchto vzdáleností je zobrazena na Obrázku 4.1. Vzdálenosti jsou založené na kosinové podobnosti. Jako aktivační funkce byla použita *Sigmoid* funkce, která normalizovala výsledek mezi nulu a jedničku.



Obrázek 4.1: Model sémantické návaznosti (Švec et al., 2023)

V našem případě testování provádíme tak, že použijeme jednotná testovací data, pro která predikujeme odpovědi. Vždy dojde k predikci tří možných odpovědí, ze kterých je jedna náhodně vybrána. Následně pomocí modelu sémantické návaznosti porovnáme testovací sekvence s odpovídajícími výstupy. Z veškerých výsledků v daném experimentu vypočteme střední hodnotu a směrodatnou odchylku. Díky těmto hodnotám jsme schopni porovnat efektivitu jednotlivých úprav v datech.

Výsledné číselné hodnoty anglických modelů nelze porovnávat s hodnotami českých modelů z důvodu rozdílných dat použitých pro natrénování modelu sémantické návaznosti.

4.4 Experimenty

Jak již bylo zmíněno (Sekce 4.1), experimenty 1-3 byly provedeny jak pro český, tak i pro anglický jazyk. Experimenty, ve kterých byla použita augmentace dat, byly provedeny pouze pro anglický jazyk. Celkem jsme provedli šest experimentů shrnutých v následujících bodech.

- E1: bezkontextová varianta
- E2: kontext bez tagu
- E3: kontext s rozdělovacím tagem
- E4: kontext s rozdělovacím tagem, který identifikuje řečníka (A:člověk, B:bot)
- Překlad: data byla augmentovaná pomocí překladu a následně upravena stejně jako v experimentu E4
- Synonyma: data byla augmentovaná pomocí synonym a následně upravena stejně jako v experimentu E4

4.4.1 Experiment E1

V tomto experimentu jsme v připravených datech nijak neuvažovali kontextové spojitosti v rámci dialogu a ani samotné řečníky. Z tohoto důvodu byly pro trénování využity pouze samotné dialogy, kde vždy použijeme promluvu jako vstup a následující promluvu v rámci dialogu jako výstup (Tabulka 4.3).

vstup	výstup
I made a new friend today.	Where did you meet them?
Where did you meet them?	We were in the same section at the book store.
We were in the same section at the book store.	Oh, so you must have the same interests.
...	...

Tabulka 4.3: Připravená data pro E1 (bezkontextová varianta)

Pomocí takto upravených dat byl natrénován model, který jsme následně použili pro predikci odpovědi při použití testovací množiny. Na závěr byl použit model návaznosti (Sekce 4.3), díky kterému jsme získali skóre (Tabulka 4.4), které nám říká, jestli vygenerovaný výstup navazuje na daný dialog.

vstup	vygenerovaný výstup	skóre
I made a new friend today.	Is there anything else I can do?	0.3859
I made a new friend today. Where did you meet them?	I met them in San Fran.	0.0132
...

Tabulka 4.4: Vybrané ukázky vygenerovaného výstupu - anglický model E1 (bezkontextová varianta)

Analogicky jsme připravili data pro český model, kde jsme dosáhli následujících výsledků (Tabulka 4.5).

vstup	vygenerovaný výstup	skóre
Dnes jsem si našla nového přítele.	Děkuji mnohokrát.	0.0986
Dnes jsem si našla nového přítele. Kde jsi je potkal?	Našel jsem 1 přítele, který by se vám mohl líbit. Je to v San Franciscu.	0.3724
...

Tabulka 4.5: Vybrané ukázky vygenerovaného výstupu - český model E1 (bezkontextová varianta)

Výsledek tohoto experimentu pro oba jazyky je uveden v následující tabulce (Tabulka 4.6).

Jazyk	Střední hodnota	Směrodatná odchylka
en	0.3238	0.5811
cs	0.1413	0.0975

Tabulka 4.6: Vyhodnocení experimentu E1

4.4.2 Experiment E2

Ve druhém experimentu jsme již uvažovali kontextové spojitosti mezi promluvami v rámci dialogu. Proto jsme upravili trénovací data tak, že na vstupu jsou vždy všechny promluvy v rámci dialogu, které předcházely odpovědi, jenž je na výstupu. Jednotlivé promluvy oddělujeme vybraným

tokenem. Na základě této přípravy dat byly provedeny dva experimenty, které se liší pouze využitým oddělovacím tokenem.

Experiment E2 a)

V tomto experimentu jsme věty oddělovali pouze mezerou. Data tak vypadají následovně (Tabulka 4.7).

vstup	výstup
I made a new friend today.	Where did you meet them?
I made a new friend today. Where did you meet them?	We were in the same section at the book store.
I made a new friend today. Where did you meet them? We were in the same section at the book store.	Oh, so you must have the same interests.
...	...

Tabulka 4.7: Připravená data pro E2a

S využitím takto upravených dat, jsme natrénovali model, jenž dosáhl následujících výsledků (Tabulka 4.8).

vstup	vygenerovaný výstup	skóre
I made a new friend today.	It's a friend of mine.	0.0516
I made a new friend today. Where did you meet them?	Yes, I met them at a coffee shop in the city.	0.0218
...

Tabulka 4.8: Vybrané ukázky vygenerovaného výstupu - anglický model E2a (kontextová varianta)

Obdobně jsme natrénovali český model s využitím přeložených dialogů (Tabulka 4.9).

vstup	vygenerovaný výstup	skóre
Dnes jsem si našla nového přítele.	Ne, to je vše, co potřebuji.	0.1218
Dnes jsem si našla nového přítele. Kde jsi je potkal?	Ano, našel jsem přítele v New Yorku.	0.0566
...

Tabulka 4.9: Vybrané ukázky vygenerovaného výstupu - český model E2a (kontextová varianta)

Výsledek tohoto experimentu pro oba jazyky je uveden v následující tabulce (Tabulka 4.10).

Jazyk	Střední hodnota	Směrodatná odchylka
en	0.2401	0.4625
cs	0.1964	0.0650

Tabulka 4.10: Vyhodnocení experimentu E2a

Experiment E2 b)

Stejně jako v předchozím experimentu (Sekce 4.4.2) jsme skládali věty předcházející promluvě v rámci dialogu. Nyní jsme však jednotlivé promluvy oddělovali tokenem <::;> (Tabulka 4.11).

vstup	výstup
I made a new friend today.	Where did you meet them?
I made a new friend today.<::;>Where did you meet them?	We were in the same section at the book store.
I made a new friend today.<::;>Where did you meet them?<::;>We were in the same section at the book store.	Oh, so you must have the same interests.
...	...

Tabulka 4.11: Připravená data pro E2b

S využitím takto upravených dat, jsme natrénovali model, jenž dosáhl následujících výsledků (Tabulka 4.12).

vstup	vygenerovaný výstup	skóre
I made a new friend today.	I'm glad you made a new friend.	0.7664
I made a new friend today.<::;>Where did you meet them?	Yes, I met them on the street.	0.0513
...

Tabulka 4.12: Vybrané ukázky vygenerovaného výstupu - anglický model E2b (kontextová varianta)

Obdobně jsme natrénovali český model s využitím přeložených dialogů (Tabulka 4.13).

vstup	vygenerovaný výstup	skóre
Dnes jsem si našla nového přítele.	Ano, to je skvělé.	0.1857
Dnes jsem si našla nového přítele. Kde jsi je potkal?	To je v pořádku.	0.3517
...

Tabulka 4.13: Vybrané ukázky vygenerovaného výstupu - český model E2b (kontextová varianta)

Výsledek tohoto experimentu pro oba jazyky je uveden v Tabulce 4.14.

Jazyk	Střední hodnota	Směrodatná odchylka
en	0.3294	0.5178
cs	0.2071	0.0648

Tabulka 4.14: Vyhodnocení experimentu E2b

4.4.3 Experiment E3

Ve třetím experimentu jsme opět chtěli zohlednit kontextové závislosti v rámci dialogu spolu s označením změny řečníka, proto jsme použili dva oddělovací tokeny $\langle A \rangle$ a $\langle B \rangle$. Token byl doplněn vždy na začátek sekvence v dané promluvě. Pokud v navazující promluvě v rámci dialogu dojde ke změně řečníka, použijeme opačný token k tokenu, jenž byl použit v předešlé promluvě. Upravená data pak vypadají následovně (Tabulka 4.15).

vstup	výstup
$\langle A \rangle$ I made a new friend today.	Where did you meet them?
$\langle A \rangle$ I made a new friend today. $\langle B \rangle$ Where did you meet them?	We were in the same section at the book store.
$\langle A \rangle$ I made a new friend today. $\langle B \rangle$ Where did you meet them? $\langle A \rangle$ We were in the same section at the book store.	Oh, so you must have the same interests.
...	...

Tabulka 4.15: Připravená data pro E3

Takto upravená data byla použita pro *Fine-tune* anglického modelu. Do-trénovaný model dosahuje následujících výsledků (Tabulka 4.16).

vstup	vygenerovaný výstup	skóre
<A>I made a new friend today.	Yes. I'm a new friend.	0.3034
<A>I made a new friend today.Where did you meet them?	How did you meet them?	0.1809
...

Tabulka 4.16: Vybrané ukázky vygenerovaného výstupu - anglický model E3 (kontextová varianta s ohledem na řečníka)

Obdobně jsme natrénovali český model s využitím přeložených dialogů (Tabulka 4.17).

vstup	vygenerovaný výstup	skóre
<A>Dnes jsem si našla nového přítele.	Ne, to je vše.	0.1790
<A>Dnes jsem si našla nového přítele.Kde jsi je potkal?	Ne, to je vše.	0.1785
...

Tabulka 4.17: Vybrané ukázky vygenerovaného výstupu - český model E3 (kontextová varianta s ohledem na řečníka)

Výsledek tohoto experimentu pro oba jazyky je uveden v následující tabulce (Tabulka 4.18).

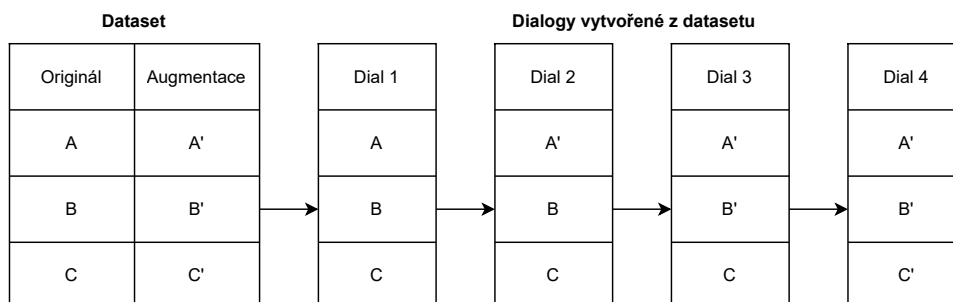
Jazyk	Střední hodnota	Směrodatná odchylka
en	0.3121	0.5277
cs	0.2111	0.0655

Tabulka 4.18: Vyhodnocení experimentu E3

4.4.4 Překlad

V tomto experimentu byla využita augmentace pomocí překladů zmíněná v Sekci 2.4.2. Pro augmentaci jsme vybrali španělský a francouzský jazyk. Způsob úpravy dat jsme zvolili stejný jako v experimentu E3 (Sekce 4.4.3), protože použité tagy obsahují nejvíce informací.

Augmentovány byly všechny promluvy obsažené v našem datasetu (Sekce 3.2). Nové dialogy byly vytvořeny tak, že jsme postupně nahrazovali promluvy jejich augmentací v chronologickém pořadí. Každé nahrazení znamenalo vytvoření nového dialogu. Pro ilustraci je postup znázorněn v následující schématu (Obrázek 4.2).



Obrázek 4.2: Dataset obsahuje jeden dialog, kde jsou jednotlivé promluvy označeny písmeny A, B, C. Písmena s apostrofem označují augmentované promluvy. Z diagramu je vidět, že z dialogu o třech promluvách lze vytvořit tři nové dialogy.

Z důvodu vysokých nároků na výpočetní výkon jsme se rozhodli omezit velikost souboru s daty připravenými pro trénování.

Francouzský překlad

Díky využití augmentace pomocí francouzského jazyka jsme se dostali na 14 097 508 promluv (Tabulka 4.19).

vstup	vygenerovaný výstup	skóre
<A>I made a new friend today.	I made a new friend today.	0.0045
<A>I made a new friend today.Where did you meet them?	i met them in san francisco.	0.1388
...

Tabulka 4.19: Vybrané ukázky vygenerovaného výstupu - anglický model FR překlad (kontextová varianta s ohledem na řečníka)

Výsledek tohoto experimentu je uveden v Tabulce 4.20.

Jazyk	Střední hodnota	Směrodatná odchylka
en	0.5124	0.6141

Tabulka 4.20: Vyhodnocení experimentu FR překlad

Španělský překlad

V tomto experimentu jsme využili augmentaci pomocí španělského jazyka. Vzniklé dialogy celkem obsahují 13 898 961 promluv (Tabulka 4.21).

vstup	vygenerovaný výstup	skóre
<A>I made a new friend today.	i made a new friend today.	0.0031
<A>I made a new friend today.Where did you meet them?	i found a friend in san francisco.	0.0242
...

Tabulka 4.21: Vybrané ukázky vygenerovaného výstupu - anglický model ES překlad (kontextová varianta s ohledem na řečníka)

Výsledek tohoto experimentu je uveden v Tabulce 4.22.

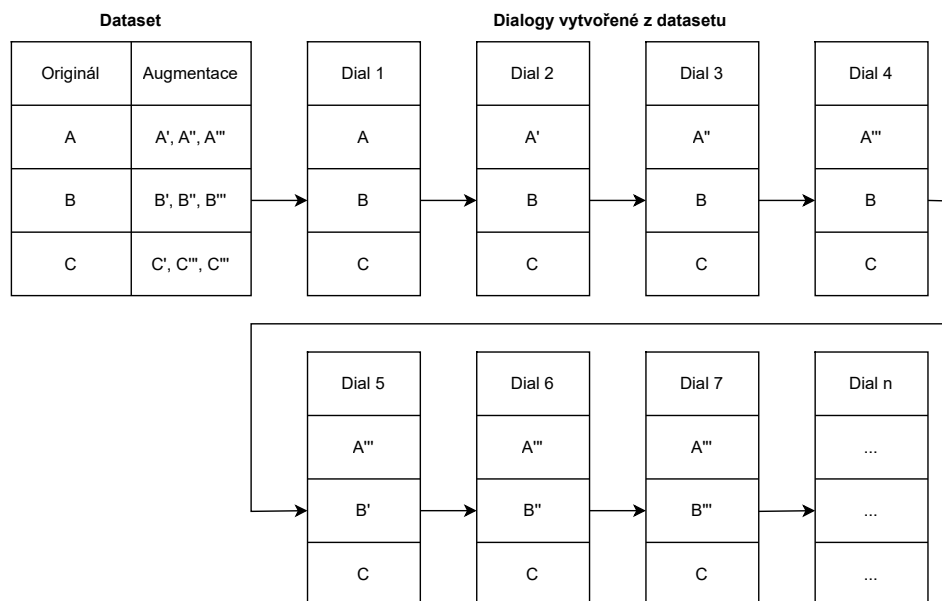
Jazyk	Střední hodnota	Směrodatná odchylka
en	0.5019	0.5739

Tabulka 4.22: Vyhodnocení experimentu ES překlad

4.4.5 Synonyma

Tento experiment je založený na augmentaci pomocí synonym představené v Sekci 2.4.1. Augmentace byla provedena tak, že postupně byla nahrazována veškerá slova v promluvě maskovacím tokenem. Pro každé maskované slovo bylo predikováno pět možných slov. Pokud například uvažujeme větu o pěti slovech, získáváme 25 nových vět spolu se skórem, jenž určuje, jak moc se dané slovo hodí do věty.

Nové dialogy byly vytvořeny stejným způsobem jako v předchozím experimentu (Sekce 4.4.4) s jedním rozdílem. U augmentace pomocí překladu, v případě že byl proveden překlad, vznikla vždy jen jedna nová věta. Nejprve jsme museli určit, jaké augmentované věty využijeme. Pro náš experiment jsme si vybrali vždy tři augmentované věty s nejvyšším skóre pro danou promluvu. Při úpravě dat byla vždy postupně nahrazována každá promluva svými možnými augmentacemi.



Obrázek 4.3: Toto schéma zobrazuje postup vytváření nových dialogů pomocí nových vět získaných díky augmentaci pomocí synonym. Postupně jsou nahrazovány veškeré původní promluvy v dialogu všemi možnými augmentacemi.

Díky augmentaci pomocí synonym náš dataset obsahuje 48 626 287 promluv (Tabulka 4.23). Proto jsme se rozhodli omezit velikost výsledného souboru s daty připravenými pro trénování z důvodu vysokých výpočetních nároků.

vstup	vygenerovaný výstup	skóre
<A>I made a new friend today.	What did you do?	4.2292
<A>I made a new friend today.Where did you meet them?	I met them at a local bar and restaurant.	0.0681
...

Tabulka 4.23: Vybrané ukázky vygenerovaného výstupu - anglický model synonyma (kontextová varianta s ohledem na řečníka)

Výsledek tohoto experimentu je uveden v Tabulce 4.24.

Jazyk	Střední hodnota	Směrodatná odchylka
en	0.3293	0.5048

Tabulka 4.24: Vyhodnocení experimentu Synonyma

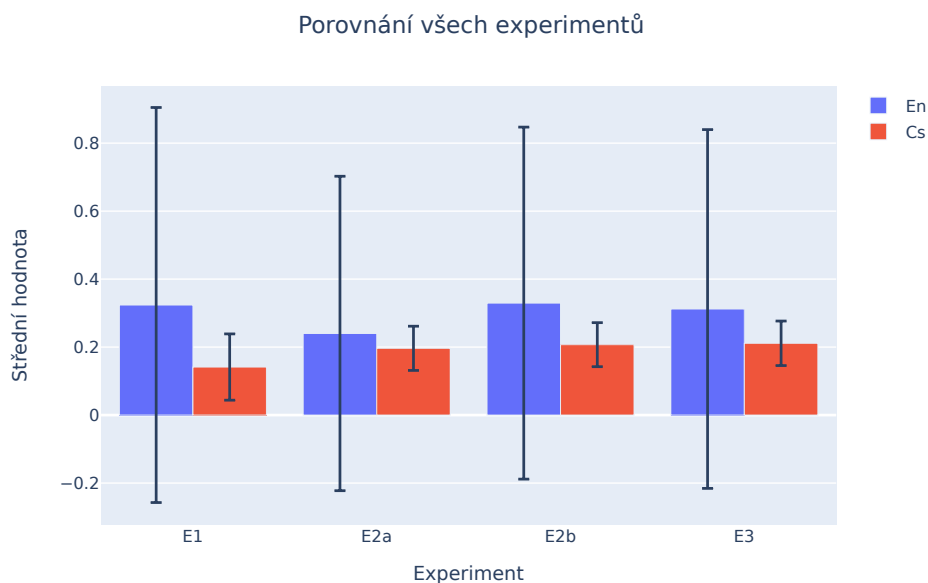
Kapitola 5

Diskuze

V této části se zaměříme na výsledky našeho výzkumu, který se zaměřoval na porovnání různých modifikací a augmentací dat. Provedli jsme několik experimentů, ve kterých jsme upravovali data obsahující přirozené dialogy, pomocí nichž jsme natrénovali modely založené na architektuře T5, jež jsme použili jako jádro pro chatovacího bota. Pro vyhodnocení jsme použili model sémantické návaznosti (Sekce 4.3). Snažíme se dosáhnout co nejnižší střední hodnoty s přijatelnou směrodatnou odchylkou.

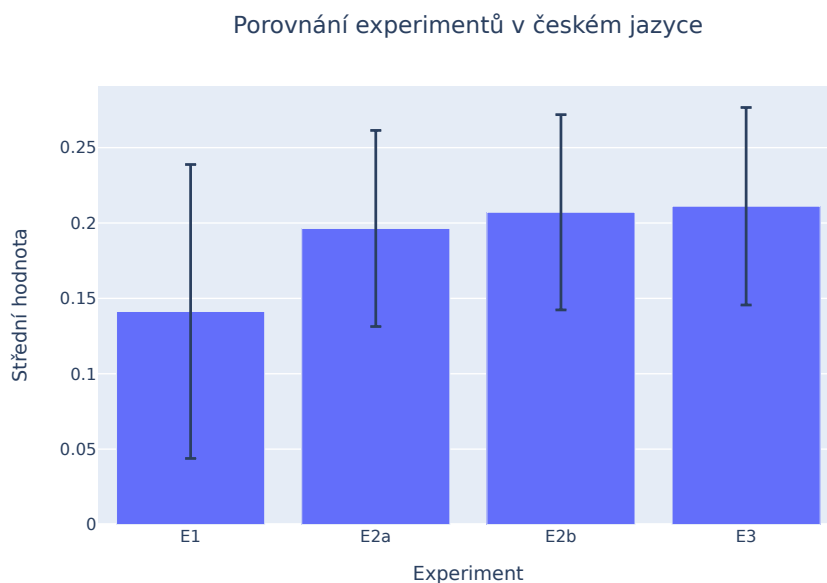
5.1 Shrnutí výsledků

V prvních čtyřech experimentech jsme modifikovali data pomocí vkládání různých tagů mezi promluvy. Tyto tagy měli zapříčinit lepší predikci odpovědi v závislosti na předchozích promluvách. Nutno podotknout, že způsob, pomocí kterého jednotlivé modely vyhodnocujeme, je experimentální, a tak jsou výsledky pouze orientační. Výsledky těchto experimentů jsou zobrazeny v následujícím grafu (Obrázek 5.1). Rozdělovací tagy byly mezi promluvy vkládány bez mezer. Je možné že lepších výsledku by bylo dosaženo vkládáním tagů oddělených mezerami.



Obrázek 5.1: Sloupce zobrazují výslednou střední hodnotu jednotlivých experimentů spolu se směrodatnou odchylkou. Barva sloupce odlišuje jazyk, ve kterém byl experiment prováděn.

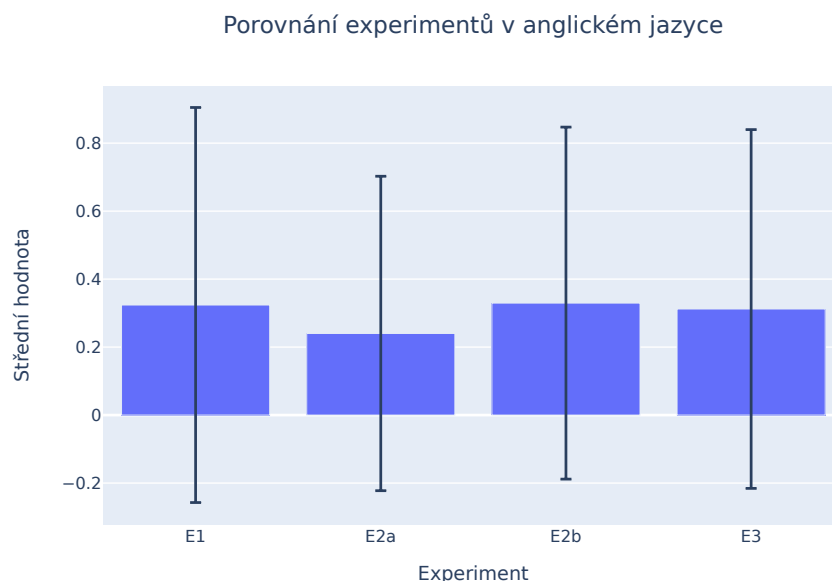
Na první pohled můžeme vidět velký rozdíl mezi českými a anglickými modely, proto budeme jednotlivé jazyky porovnávat zvlášť. Jedny z důvodů, proč nemůžeme české a anglické modely porovnávat, jsou zmíněny v Sekci 4.3 a v Sekci 2.4.2. Nejprve porovnáme výsledky modelů natrénovaných na českém jazyce (Obrázek 5.2).



Obrázek 5.2: Výsledné střední hodnoty jednotlivých experimentů spolu se směrodatnou odchytkou. (České modely)

Modely natrénované na datech v českém jazyce nedosáhly dobrých výsledků. Jednotlivé úpravy trénovacích dat vedly ke zhoršení predikcí. Na první pohled se může zdát, že směrodatné odchytky modelů jsou poměrně nízké. Příčinou těchto hodnot jsou podobné predikce na veškeré vstupy, to znamená, že takto malá směrodatná odchytky není žádoucí.

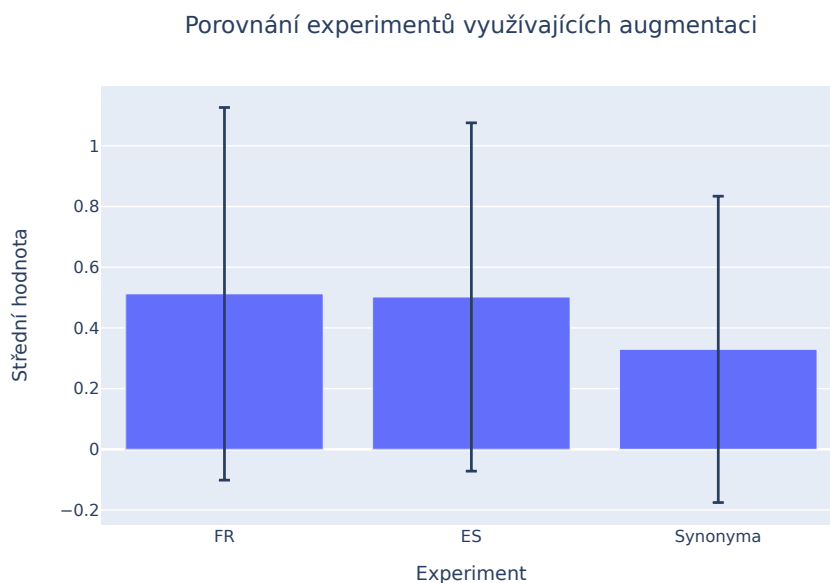
U českých modelů jsme nedosáhli zlepšení výsledků v závislosti na uvažovaném kontextu. Naopak u modelů natrénovaných na anglických datech jsme dosáhli zajímavější výsledků (Obrázek 5.3).



Obrázek 5.3: Výsledné střední hodnoty jednotlivých experimentů spolu se směrodatnou odchylkou. (anglické modely)

Nejllepších výsledků dosáhl model (Experiment E2a - Sekce 4.4.2) natrénovaný na datech modifikovaných mezerou. Střední hodnota výsledků tohoto modelu dosáhla značně lepších hodnot, než ostatní modely. Nejlépe také dopadla směrodatná odchylka, kdy model predikuje rozmanité odpovědi a zároveň se odpovědi drží tématu. Ostatní experimenty v anglickém jazyce překvapivě nedosáhly očekávaných výsledků. Jejich predikce jsou přijatelné, ale kontextové verze modelů používající tagy nedosáhly lepších výsledků než bezkontextová verze.

Experimenty, ve kterých byla upravena data augmentací, není vhodné porovnávat s ostatními experimenty, protože jsme z důvodu výpočetní náročnosti omezili velikost trénovacích dat na 12GB. Při omezování velikosti nedošlo k nekorektnímu rozdělení závěrečného dialogu. To znamená, že data, použítá pro trénování, nemusela obsahovat celý rozsah původních datasetů. Například se v nich mohly vyskytovat pouze části původních datasetů, tudíž trénování bylo prováděno na omezené pestrosti dialogů. Graf 5.4 interpretuje výsledky těchto experimentů.

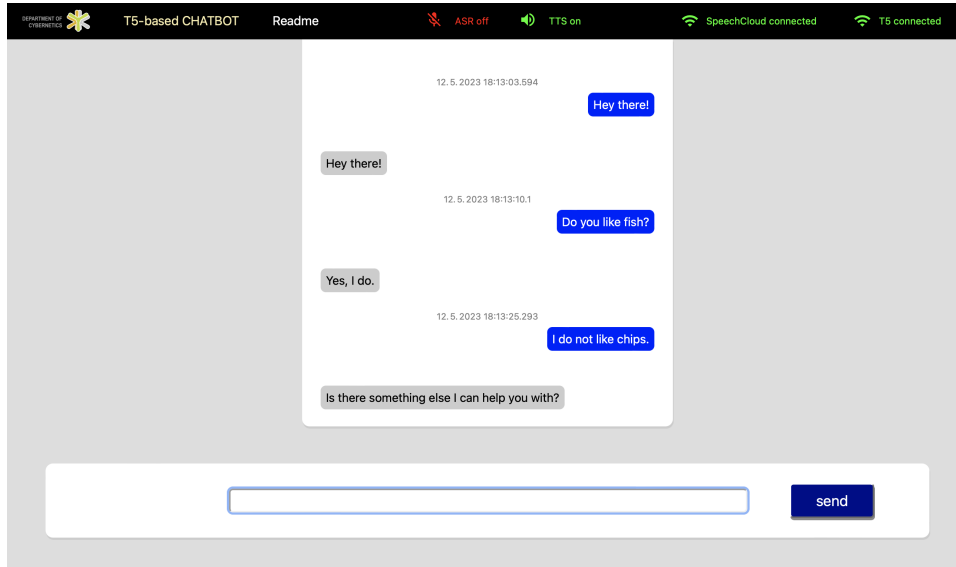


Obrázek 5.4: Sloupce zobrazují výslednou střední hodnotu jednotlivých experimentů spolu se směrodatnou odchylkou.

Experimenty, které byly augmentovány pomocí překladů, dosáhly velmi podobných výsledků. Lepšího výsledku pak dosáhl model založený na augmentaci pomocí synonym. Tento rozdíl však může být zapříčiněn právě různou pestrostí dialogů v trénovacích datech. Ačkoliv jsou trénovací soubory stejně velké, mohou obsahovat jiný rozsah původních datasetů.

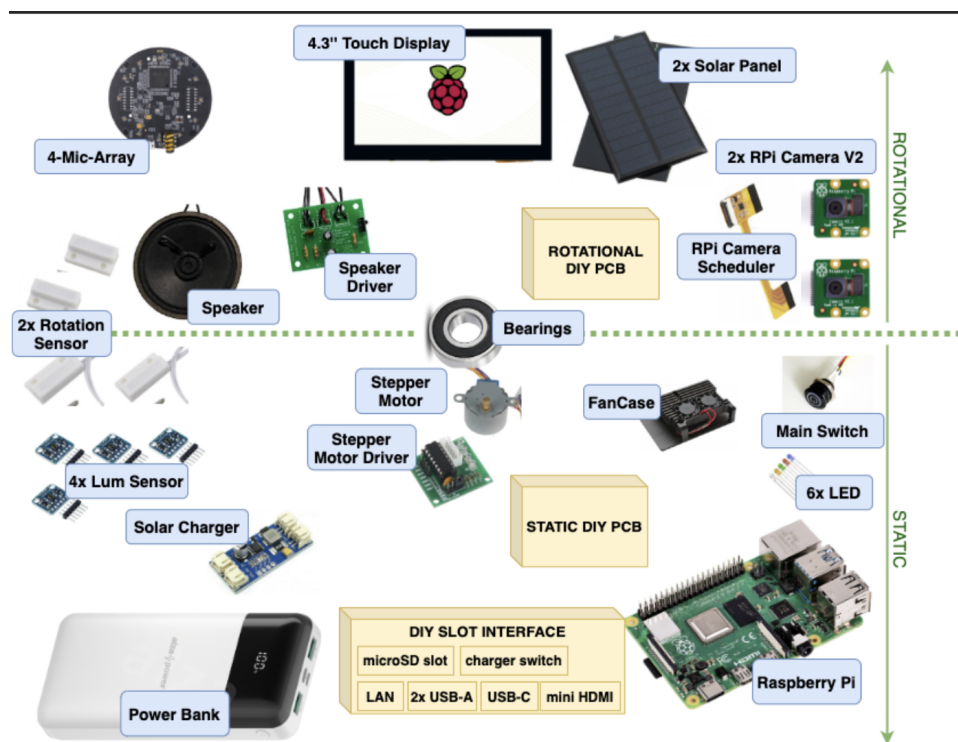
5.2 Reálné využití

Pro naše natrénované modely jsme vytvořili chatovací aplikaci s webovým rozhraním (Obrázek 5.5).



Obrázek 5.5: Webové rozhraní pro chatovací aplikaci

Dále jsme modely implementovali do robota (Obrázek 5.6) postaveného na platformě Raspberry Pi. Zjistili jsme, že výkon použitého zařízení není dostatečný pro provoz modelu, a tak jsme využili requestů z robota na platformě Raspberry Pi na počítač s dostatečným výkonem.



Obrázek 5.6: Schéma robota

Kapitola 6

Závěr

Cílem této bakalářské práce bylo porozumění problematice zpracování přirozené řeči v souvislosti s neuronovými sítěmi založenými na architektuře Transformer, které v posledních letech dosáhly velmi dobrých výsledků.

Teoretická část této práce byla věnována popisu architektury Transformer, kde jsme vysvětlili principy jejího fungování. Dále jsme se zaměřili na moderní architektury T5, GPT a BERT, jenž jsou založené na architektuře Transformer. V závěru teoretické části jsme popsali funkčnost strojového překladu a metody, které jsme použili pro augmentování dat. Celá následující kapitola byla věnována popisu získaných datasetů, ze kterých byl následně vytvořen dataset používaný pro naše experimenty.

V experimentální části jsme se věnovali modifikaci trénovacích dat s cílem dosáhnout co nejlepších odpovědí na různé vstupní promluvy. Pro vyhodnocení jsme použili model sémantické návaznosti, který porovnával jednotná trénovací data a generované predikce. Nejprve jsme natrénovali model, ve kterém jsme neuvažovali kontext promluv v rámci dialogu. V dalším experimentu jsme uvažovali veškeré promluvy předcházející odpovědí. Promluvy jsme rozdělovali pomocí předem určených tagů. Ve třetím experimentu jsme tagy zvolili tak, aby reflektovaly změnu řečníka v rámci dialogu. Z těchto experimentů vyšla nejlépe úprava využívající rozdělávání promluv pomocí mezer pro anglický model. České modely dosáhly podobných výsledků, kdy jejich predikce byly jednotvárné a proto i jejich směrodatná odchylka byla nízká. Poslední experimenty využívají datasety, které prošly augmentací. Použili jsme augmentace pomocí synonym a s využitím překladů. Pro trénovací data jsme zvolili tagy obsahující informace o změně řečníka a zjistili jsme, že největší zlepšení způsobí augmentace pomocí synonym, ačkoliv jsme omezili velikost trénovacího souboru z důvodu výpočetní náročnosti.

6.1 Práce do budoucna

Některé nápady na budoucí práci jsou uvedené v následujících bodech.

- *Filtrace nevhodných promluv*
Rozpracovali jsme filtraci nevhodných promluv pomocí predikce

sentimentu založené také na architektuře T5. Natrénovali jsme několik modelů na datasetech ČSFD a IMDB, které obsahují recenze filmů spolu s anotací sentimentu založené na hodnocení. Tyto modely jsme nestihli odexperimentovat na prediccích našeho chatovací bota a zároveň tato filtrace nemusí být úplně efektivní z důvodu filtrace negativního sentimentu, jenž nemusí být vždy špatný natolik, aby byl filtrován.

- *Kombinace augmentací a modifikací dat*
V této práci jsme nezkoušeli kombinovat různé modifikace a stejně tak jsme nezkoušeli kombinovat způsoby augmentace dat. Zajímavé by bylo zjistit, jestli by různé kombinace měly vliv na lepší predikce. Těchto kombinací a způsobu je téměř neomezené množství.
- *Použití celých trénovacích dat*
U modelů, které využívali augmentovaná trénovací data, jsme omezovali velikost souborů z důvodu výpočetní náročnosti. Cílem by bylo data upravit a rozdělit tak, abychom využili maximální potenciál augmentace.
- *Natrénování více modelů*
Veškeré modely jsme trénovali za pomoci stejných hyper-parametrů. Cílem by bylo natrénovat velké množství modelů, kde každý by byl natrénován při jiných hyper-parametrech. Pro všechny experimenty jsme využívali architekturu T5. Další možností může být vyzkoušet i jiné architektury a porovnat je mezi sebou.

Literatura

- [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] Murat Sazli. „A brief review of feed-forward neural networks“. In: *Communications Faculty Of Science University of Ankara* 50 (led. 2006), s. 11–17. doi: 10.1501/commua1-2_0000000026.
- [3] Edward Loper Bird Steven a Ewan Klein. *Natural Language Processing with Python*. O’Reilly Media Inc., 2009.
- [4] Dzmitry Bahdanau, Kyunghyun Cho a Yoshua Bengio. *Neural Machine Translation by Jointly Learning to Align and Translate*. 2016. arXiv: 1409.0473 [cs.CL].
- [5] Yonghui Wu et al. *Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation*. 2016. arXiv: 1609.08144 [cs.CL].
- [6] Yanran Li et al. *DailyDialog: A Manually Labelled Multi-turn Dialogue Dataset*. 2017. arXiv: 1710.03957 [cs.CL].
- [7] Ashish Vaswani et al. *Attention Is All You Need*. 2017. arXiv: 1706.03762 [cs.CL].
- [8] Rani Horev. *BERT Explained: State of the art language model for NLP*. Accessed 24 April 2023. 2018. url: <https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270>.
- [9] Alec Radford a Karthik Narasimhan. „Improving Language Understanding by Generative Pre-Training“. In: 2018.
- [10] Jacob Devlin et al. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2019. arXiv: 1810.04805 [cs.CL].
- [11] Yinhan Liu et al. *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. 2019. arXiv: 1907.11692 [cs.CL].
- [12] Soujanya Poria et al. *MELD: A Multimodal Multi-Party Dataset for Emotion Recognition in Conversations*. 2019. arXiv: 1810.02508 [cs.CL].
- [13] Maarten Sap et al. *SocialQA: Commonsense Reasoning about Social Interactions*. 2019. arXiv: 1904.09728 [cs.CL].
- [14] Ashkan Eliasy a Justyna Przychodzen. „The role of AI in capital structure to enhance corporate funding strategies“. In: *Array* 6 (čvc. 2020), s. 100017. doi: 10.1016/j.array.2020.100017.
- [15] Rohan Jagtap. *OpenAI GPT: Generative Pre-Training for Language Understanding*. Accessed 24 April 2023. 2020. url: <https://medium.com/dataseries/openai-gpt-generative-pre-training-for-language-understanding-bbbdb42b7ff4>.

- [16] Colin Raffel et al. *Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer*. 2020. arXiv: 1910.10683 [cs.LG].
- [17] Pragati Baheti. *Activation Functions in Neural Networks [12 Types Use Cases]*. Accessed 9 May 2023. 2021. url: <https://www.v7labs.com/blog/neural-networks-activation-functions>.
- [18] Martin Bulín. „On Using Multi-Agent Technologies to Build Neural Networks“. In: (2021).
- [19] Ketan Doshi. *Transformers Explained Visually (Part 3): Multi-head Attention, deep dive*. Accessed 26 April 2023. 2021. url: <https://towardsdatascience.com/transformers-explained-visually-part-3-multi-head-attention-deep-dive-1c1ff1024853>.
- [20] Dale Markowitz. *Transformers, Explained: Understand the Model Behind GPT-3, BERT, and T5*. Accessed 25 April 2023. 2021. url: <https://daleonai.com/transformers-explained>.
- [21] Wanshun Wong. *What is Residual Connection?* Accessed 20 April 2023. 2021. url: <https://towardsdatascience.com/what-is-residual-connection-efb07cab0d55>.
- [22] Pei Zhou et al. „Commonsense-Focused Dialogues for Response Generation: An Empirical Study“. In: *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue*. Singapore a Online: Association for Computational Linguistics, 2021. url: <https://arxiv.org/abs/2109.06427>.
- [23] Jonte Dancker. *A Brief Introduction to Recurrent Neural Networks*. Accessed 1 May 2023. 2022. url: <https://towardsdatascience.com/a-brief-introduction-to-recurrent-neural-networks-638f64a61ff4>.
- [24] Thomas van Dongen. *Demystifying efficient self-attention*. Accessed 25 April 2023. 2022. url: <https://towardsdatascience.com/demystifying-efficient-self-attention-b3de61b9b0fb>.
- [25] Keon Lee, Kyumin Park a Daeyoung Kim. *DailyTalk: Spoken Dialogue Dataset for Conversational Text-to-Speech*. 2022. arXiv: 2207.01063 [eess.AS].
- [26] Arjun Sarkar. *All you need to know about ‘Attention’ and ‘Transformers’ — In-depth Understanding — Part 1*. Accessed 21 April 2023. 2022. url: <https://towardsdatascience.com/all-you-need-to-know-about-attention-and-transformers-in-depth-understanding-part-1-552f0b41d021>.
- [27] Zixiu Wu et al. „Anno-MI: A Dataset Of Expert-Annotated Counseling Dialogues“. In: *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2022.
- [28] FAWAD ALI. *GPT-1 to GPT-4: Each of OpenAI’s GPT Models Explained and Compared*. Accessed 24 April 2023. 2023. url: <https://www.makeuseof.com/gpt-models-explained-and-compared/>.
- [29] Bernard Marr. *ChatGPT: What Are Hallucinations And Why Are They A Problem For AI Systems*. Accessed 21 May 2023. 2023.
- [30] Mehreen Saeed. *A Gentle Introduction to Positional Encoding in Transformer Models, Part 1*. Accessed 25 April 2023. 2023. url:

- <https://machinelearningmastery.com/a-gentle-introduction-to-positional-encoding-in-transformer-models-part-1/>.
- [31] J. Švec et al. *Asking Questions: an Innovative Way to Interact with Oral History Archives*. Accessed 15 May 2023. 2023.

Příloha A1

Zpracovaný dialogový korpus CS

d01_000001	1	1	<A>	Byl jsem tak naštvaný, že už jsem to nemohl udržet
d01_000002	1	2		Nadechl ses?
d01_000003	1	3	<A>	Udělal jsem, vzplanul jsem vztekem
d01_000004	1	4		Musíte se uklidnit, je to jen videohra
d01_000005	1	5	<A>	Vím, neměl bych to nechat takhle dopadnout.
d01_000006	1	6		vypusťte páru a vraťte se
d01_000007	2	1	<A>	Po porážce jsem byl tak zničený.
d01_000008	2	2		Měl jsi tomu věnovat více úsilí!
d01_000009	2	3	<A>	Ano, já vím! Odešel jsem v návalu vzteku poté, co jsem byl poražen v soutěži videoher.
d01_000010	2	4		Příště se musíš připoutat!
d01_000011	2	5	<A>	Já vím, budu.
d01_000012	3	1		Byl jsem tak naštvaný, že jsem včera prohrál.
d01_000013	3	2	<A>	co jsi hrál?
d01_000014	3	3		Byla to soutěž videoher.
d01_000015	3	4	<A>	Proti komu jsi hrál?
d01_000016	3	5		Byli to hráči z města vedle nás.
d01_000017	3	6	<A>	To se občas stávalo. Příště víc štěstí.
d01_000018	4	1		Pamatuješ si, když jsem si hrál s klukama, když jsi odcházel?
d01_000019	4	2	<A>	Rozbil jsi něco?
d01_000020	4	3		Jo, kluci se do mě pustili a rozbili jsme dveře. Tak jsem zavolał mámě a řekl jí to.
d01_000021	4	4	<A>	To, co říkala?
d01_000022	4	5		Byla opravdu naštvaná a řekla mi, abych zavolał opraváře.
d01_000023	4	6	<A>	No, alespoň ještě nejste uzemnění.
d01_000024	5	1		Myslím, že je na mě Taylor naštvaná.
...

Příloha A2

Zpracovaný dialogový korpus EN

d01_000001	1	1	<A>	I got so mad, I couldn't contain it any-more
d01_000002	1	2		Did you huff off?
d01_000003	1	3	<A>	I did, I flared up into anger
d01_000004	1	4		You need to calm down, it's just a video game
d01_000005	1	5	<A>	I know, I should not let it get to me like this.
d01_000006	1	6		blow off some steam and come back
d01_000007	2	1	<A>	I was so devastated after the defeat.
d01_000008	2	2		You should have put more effort into it!
d01_000009	2	3	<A>	yes, I know! I left in a huff of rage after I was beaten in the video game competition.
d01_000010	2	4		You need to buckle up next time!
d01_000011	2	5	<A>	I know, I will.
d01_000012	3	1		I was so upset that I lost yesterday.
d01_000013	3	2	<A>	What did you play?
d01_000014	3	3		It was a video game competition.
d01_000015	3	4	<A>	Who did you play against?
d01_000016	3	5		They were players from the town next to us.
d01_000017	3	6	<A>	That happened sometimes. Better luck next time.
d01_000018	4	1		Remember when I was playing with the guys while you were leaving?
d01_000019	4	2	<A>	Did you break something?
d01_000020	4	3		Yeah, the boys tackled me and we broke the door. So I called my mom and told her.
d01_000021	4	4	<A>	What did she say?
...