# Využití LLM k extrakci strukturované informace z textů lékařských zpráv

# Abstrakt

# Motivace

Růst umělé inteligence v posledních letech nabírá značného tempa a její využití se postupně rozšiřuje do většiny oborů lidské činnosti. Jednou z klíčových oblastí umělé inteligence je zpracování přirozeného jazyka (NLP – *Natural Language Processing*), jehož cílem je analýza, porozumění a generování textu či mluveného slova. Významným milníkem v této oblasti se staly velké jazykové modely (LLM – *Large Language Models*), které vykazují schopnost pracovat s komplexními a nestrukturovanými daty.

Zdravotnictví představuje oblast s enormním množstvím textových dat, zejména ve formě lékařských zpráv, nálezů a klinických záznamů. Tyto dokumenty obsahují cenné informace o zdravotním stavu pacientů, průběhu onemocnění, provedených vyšetřeních a zvolené léčbě. Většina těchto dat je však uložena v nestrukturované podobě, což výrazně komplikuje jejich další zpracování, analýzu a využití pro výzkumné či klinické účely.

Automatická extrakce strukturovaných informací z lékařských textů by mohla výrazně snížit časovou i administrativní zátěž zdravotnického personálu a zároveň umožnit efektivnější práci s daty. Přestože jsou velké jazykové modely slibným nástrojem pro tento účel, jejich použití na reálných lékařských zprávách zatím není dostatečně prozkoumáno, zejména z hlediska spolehlivosti, konzistence výstupů a etických aspektů.

Další výzvu představuje ochrana citlivých osobních údajů pacientů. Legislativní požadavky, jako je nařízení GDPR, kladou důraz na anonymizaci dat a bezpečné nakládání s informacemi, což omezuje přímé využití moderních jazykových modelů v klinickém prostředí.

Motivací této práce je proto přispět k lepšímu porozumění možnostem využití velkých jazykových modelů při extrakci strukturovaných informací z anonymizovaných lékařských zpráv a porovnat jejich chování napříč různými modely v kontextu reálných zdravotnických dat.

# Obsah

# Kapitola 1

# Úvod

České zdravotnictví se v současnosti potýká s výzvami spojenými s digitalizací nově vznikajících i historicky papírově archivovaných dat. Jejich objem v důsledku modernizace a rozvoje nových technologií rychle narůstá. Tento trend přináší významnou přidanou hodnotu jak pro současné, tak i budoucí pacienty. Rozmach metod umělé inteligence, zejména velkých jazykových modelů v oblasti zpracování přirozeného jazyka (NLP), poskytuje zdravotnickému personálu i vědcům nové nástroje, které mohou významně přispět k efektivnější práci a otevřít prostor pro nové příležitosti.

Data představují základní stavební kámen pro vývoj, zlepšování a udržování aktuálnosti velkých jazykových modelů. Lékařské zprávy však v porovnání s jinými typy dat narážejí na specifické překážky, a to zejména v oblasti zajištění anonymity a v souladu s evropským nařízením GDPR. V této práci jsou proto všechna použitá data plně anonymizována a jejich obsah je využíván pouze v nezbytném rozsahu.

Hlavním cílem práce je prověřit možnosti současných generativních AI nástrojů pro zpracování volného textu lékařských zpráv a ověřit jejich schopnost extrahovat významné informace do strukturované podoby. Zároveň je snahou posoudit, do jaké míry lze tyto nástroje využít v českém zdravotnickém prostředí, kde jazyková i doménová specifika představují významnou překážku.

Data využitá v této práci tvoří popisné zprávy k CT snímkům pacientů s diagnózou mrtvice a Crohnovy choroby. Zprávy pocházejí z Fakultní nemocnice Plzeň a jsou psány výhradně v českém jazyce. Texty přirozeně obsahují překlepy, odborný žargon, zkratky a další prvky, které mohou komplikovat jejich automatické zpracování.

Většina dosavadních výzkumů v oblasti zpracování lékařských zpráv se soustředí především na anglický jazyk. Čeština se však vyznačuje výraznou morfologickou složitostí a četným výskytem výjimek, zkratek a česko-anglických kombinací. Tato práce se proto zaměřuje na zjištění, zda jsou vybrané modely schopné tyto překážky překonat a nabídnout relevantní a prakticky využitelné výsledky.

# Kapitola 2

# Zpracování přirozeného jazyka

## 2.1 Zpracování přirozeného jazyka

Přirozený jazyk (dále jen PJ) představuje základní prostředek lidské komunikace a přenosu znalostí. Umožňuje předávání informací napříč generacemi a propojuje historický vývoj lidstva se současností. Jazyk slouží nejen ke komunikaci, ale také k uchovávání a strukturování lidského poznání.

Zpracování přirozeného jazyka (*Natural Language Processing*, NLP) je oblast umělé inteligence, která se zabývá vývojem metod a algoritmů umožňujících počítačům porozumět, analyzovat a generovat lidský jazyk. Cílem NLP je vytvořit systémy schopné pracovat s jazykem podobným způsobem, jakým jej používá člověk.

Aby byl stroj schopen efektivně zpracovávat přirozený jazyk, musí řešit řadu základních otázek, mezi které patří zejména:

- co jsou slova, jejich tvary a vnitřní struktura (např. morfémy),
- jak se slova a větné složky kombinují do vět,
- jaké významy slova nesou a co označují,
- jak se význam celé věty skládá z významů jednotlivých slov a slovních spojení.

Kromě toho musí být systém zpracovávající přirozený jazyk schopen orientovat se v různých jazykových rovinách, jako jsou rovina fonetická, morfologická, syntaktická a sémantická, případně i pragmatická. Schopnost porozumět přirozenému jazyku tak zahrnuje práci s jeho komplexní strukturou a kontextem, který význam jednotlivých jazykových prvků ovlivňuje.

Vývoj metod pro zpracování přirozeného jazyka představoval dlouhodobý a náročný proces. První přístupy byly založeny na ručně definovaných pravidlech a lingvistických znalostech, zatímco moderní NLP systémy využívají především metody strojového učení a hlubokého učení. Tyto přístupy umožňují automatické získávání jazykových vzorů z velkého množství textových dat a vedly k výraznému zlepšení výsledků v řadě praktických úloh, jako je strojový překlad, analýza textu nebo extrakce informací [7].

### 2.1.1 Funkce NLP

Aby byly stroje schopny porozumět lidské konverzaci a pracovat s přirozeným jazykem, byly vyvinuty algoritmy zpracování přirozeného jazyka. Proces zpracování textu lze rozdělit do několika základních částí, které na sebe navazují:

- předzpracování textu,
- reprezentace textu,
- analýza textu,
- syntaktická analýza.

Předzpracování textu zahrnuje základní operace, jako je tokenizace a *lowercasing*. Tokenizace rozděluje vstupní text na jednotlivé tokeny, nejčastěji slova, zatímco *lowercasing* převádí všechna písmena na malá, čímž se snižuje variabilita textu. Dále se používají techniky jako lemmatizace, která převádí slova na jejich základní tvar (lemma), nebo stemming, jehož cílem je nalezení kmene slova. Tyto kroky napomáhají sjednocení různých tvarů slov a zjednodušují další zpracování.

Ve fázi reprezentace textu dochází k převodu textových dat do numerické podoby, se kterou je možné dále pracovat. Jedním ze základních přístupů je výpočet četnosti výskytu jednotlivých slov v dokumentu. Pomocí vzorce *TF-IDF* (*Term Frequency–Inverse Document Frequency*) je každému slovu přiřazena váha, která zohledňuje jeho důležitost v rámci dokumentu i celého korpusu. Například v anglickém jazyce mají velmi častá slova, jako jsou **and** nebo **the**, nižší váhu než méně frekventovaná slova, která nesou vyšší informační hodnotu.

Analýza textu se zaměřuje na práci s významem a kontextem. V přirozeném jazyce se často vyskytují mnohoznačná slova nebo homonyma, jejichž význam závisí na konkrétním kontextu. U věty „Vlak jel po kolejích.“ je zřejmé, že slovo *kolejích* označuje dopravní infrastrukturu, nikoli vysokoškolské koleje. Pro zachycení tohoto kontextu se využívají metody, jako je rozpoznávání pojmenovaných entit (*Named Entity Recognition*, NER), které přiřazují slovům nebo jejich skupinám významové kategorie. Výstupem může být například označení *{kolejích: doprava}*. Součástí analýzy textu může být také určování sentimentu, tedy rozpoznání, zda je význam věty kladný, neutrální nebo záporný, což se odvozuje od použitých slov a jejich kontextu.

Syntaktická analýza se zaměřuje na vztahy mezi jednotlivými slovy ve větě a jejich gramatickou funkci. Jejím cílem je rozdělení slov podle slovních druhů a určení jejich role ve větě, například zda se jedná o podmět, přísudek nebo předmět. Syntaktická analýza umožňuje lépe pochopit strukturu věty a vztahy mezi jejími částmi, což je důležité zejména při složitějších jazykových konstrukcích. Tyto informace se dále využívají například při strojovém překladu, extrakci informací nebo porozumění významu celých vět. [8]

# Kapitola 3

# Velké datové modely

Významný pokrok v oblasti zpracování přirozeného jazyka zaznamenal podobor označovaný jako velké jazykové modely (*Large Language Models*, LLM). Tyto modely jsou založeny na rozsáhlých neuronových sítích s velkým počtem parametrů, které jsou trénovány na rozsáhlých textových datech. Jejich cílem je naučit se statistické a sémantické vztahy mezi slovy a větami a na jejich základě generovat smysluplný textový výstup.

Architektura velkých jazykových modelů je založena především na dopředných neuronových sítích (*Feed-Forward Networks*) a mechanismech pozornosti (*attention*), které umožňují modelu pracovat s kontextem celého vstupu. Na rozdíl od starších přístupů nejsou moderní velké jazykové modely založeny na rekurentních neuronových sítích, ale využívají paralelní zpracování vstupních sekvencí, což výrazně zvyšuje jejich efektivitu a škálovatelnost.

### Generativní AI

V souvislosti s velkými jazykovými modely se často používá pojem *generativní umělá inteligence* (*Generative AI*, zkr. GenAI). Tento pojem označuje modely schopné generovat nový obsah na základě vzorů získaných z trénovacích dat. Do generativní AI spadají nejen velké jazykové modely, ale také modely generující obraz, zvuk či video. Velké jazykové a multimodální modely tak tvoří základ současné generativní umělé inteligence, která umožňuje tvorbu textu, programového kódu, obrazového i zvukového obsahu. [9]

## 3.1 Transformátor

Architektura transformátoru byla poprvé představena v roce 2017 ve vědeckém článku „Attention Is All You Need“ autory Vaswanim a kol. a je považována za zásadní milník v oblasti hlubokého učení [13]. Transformátor představuje neuronovou architekturu založenou na mechanismu pozornosti, který umožňuje modelu při zpracování textu zohledňovat vztahy mezi všemi slovy ve vstupní sekvenci současně.

Základními stavebními prvky transformátoru jsou vrstvy vícenásobné pozornosti (*multi-head attention*) a dopředné neuronové sítě. Díky této architektuře je možné efektivně zachytit dlouhodobé závislosti v textu bez nutnosti rekurentního zpracování. Transformátor se stal základem většiny moderních velkých jazykových modelů a významně přispěl k jejich vysoké výkonnosti v úlohách zpracování přirozeného jazyka. [11]

## 3.2 Typy a dělení LLM

### 3.2.1 Podle tréninku

Nejčastější formou tréninku LLM je **předtrénovaný** model (angl. pre-trained). Modely se pomocí učení s učitelem učí na rozsáhlém počtu často neoznačených dat „porozumět textu.“ Během této fáze zkoumají modely sémantiku, syntaxi a kontext dat. Rozlišují se 2 způsoby, od nuly (angl. from scratch), nebo pravidelné (angl. Continuous). Od nuly jak již z názvu vyplývá, se myslí vytvoření nového modelu zcela od počátku, zatímco u pravidelného je v podstatě aplikovaní **transfer learning**, což znamená, že se již existující model trénuje na nových datech. [14, 15]

Další formou je technika zvaná **fine-tune**, což se dá přeložit jako doladění. Metoda využívá již předtrénované modely, které se doučují pro specifické účely. Různé techniky zdokonalují předtrénované modely a zlepšují výkony ve specializovaných úkolech. [14]

### 3.2.2 Podle funkčních kategorií

Velké jazykové modely lze rozdělit do několika kategorií podle jejich ladění. **Základní modely** byly první modely natrénované na nespočetně nestrukturovaných a neoznačených datech. Základní modely jsou trénovány především na predikci následujícího tokenu v textu a slouží jako výchozí bod pro další úpravy. Tyto modely se dnes nevyužívají pro běžné použití kvůli nedostatečné přesnosti v ohledu zpracování instrukcí, ale využívájí se jako stavební kámen pro další typy LLM. **Instrukčně laděné modely** jsou příkladem základních modelů, které byly dotrénovány na základě lidské odezvy. Jsou následně přizpůsobeny k plnění konkrétních úloh na základě zadaných instrukcí. **Multimodální modely** rozšiřují model a zpracovávání vizuálních a zvukových souborů. [16, 17]

Jiným přístupem se zabývá architektura **Mixture of Experts (MoE)**, která využívá více specializovaných podsítí (expertů), přičemž tzv. *gating network* dynamicky rozhoduje, kteří experti budou aktivováni pro daný vstup. V praxi lze MoE přirovnat k systému automatického směrování požadavků v informačních systémech. Například při zpracování lékařských zpráv může jeden expert zpracovávat laboratorní hodnoty, jiný klinické diagnózy a další farmakologické informace. Řídicí mechanismus následně kombinuje jejich výstupy do výsledné reprezentace.

Podobným typem jsou **Agentní systémy**. Zatímco klasický LLM generuje odpověď pouze ze svých znalostí, agentní systém provádí následující úlohy: vnímání, uvažování, akce a učení. Jinak řečeno, LLM nejprve zpracuje dotaz, poté uváží jaké nástroje bude potřebovat a následně je využije pro tvorbu výstupu. [18, 19]

## 3.3 Schopnosti LLM

Velké jazykové modely jsou trénovány na široké spektrum úloh a disponují řadou schopností v závislosti na typu vstupních dat. Textově zaměřené modely dokáží provádět generování textu, strojový překlad, sumarizaci dokumentů nebo odpovídání na otázky. Multimodální modely rozšiřují tyto schopnosti o práci s obrazovými daty, například rozpoznávání obsahu obrázků nebo extrakci textu z obrazových vstupů. Další skupinu tvoří modely zaměřené na zpracování zvuku, které umožňují převod řeči na text a naopak. Kombinací těchto schopností vznikají komplexní systémy schopné pracovat s různými typy vstupních dat a poskytovat uživateli přirozené rozhraní pro komunikaci s umělou inteligencí. [11, 12]

## 3.4 AI Prompting

AI prompt je název pro instrukci, otázku, či tvrzení, které člověk poskytne velkým datovým modelům. V promptu se nachází informace, které LLM zpracuje a jejich základě poskytuje výsledek. Čím detailnější prompt, tím by měl být výsledek přesnější. Promptování dokáže zlepšit přesnost a kvalitu výsledku modelu, bez žádného dalšího dotrénování na konkrétní data. [21]

**Příklad vlivu promptu:**

**Vstupní text:** „Pacient byl přijat s bolestí na hrudi, EKG bez patologického nálezu, troponin negativní."

**Neupřesněný prompt:** „Shrň text." → Výstup: obecné shrnutí bez struktury

**Strukturovaný prompt:** „Z následující lékařské zprávy extrahuj strukturované informace ve formátu JSON se sekcemi: příznaky, vyšetření, laboratorní nálezy." → Výstup: strukturovaná data vhodná pro další zpracování

### 3.4.1 Strategie využívání LLM v rámci promptingu

Efektivní formulací instrukcí lze dosáhnout kvalitnějších výstupů i bez nutnosti nákladného doladění parametrů (tzv. *fine-tuning*). Tato technika, označovaná jako **In-Context Learning** (učení v kontextu), umožňuje modelu lépe porozumět specifikům daného úkolu a generovat přesnější odpovědi na základě informací obsažených přímo v zadání. [20]

V závislosti na dostupnosti dat, požadované rychlosti a komplexnosti úlohy rozlišujeme několik základních přístupů:

- **Zero-shot prompting:** Tato metoda představuje nejrychlejší způsob interakce, kdy model generuje odpověď bez jakýchkoliv předchozích příkladů, pouze na základě instrukce a znalostí získaných během fáze pre-trainingu. Je vhodná pro standardizované úlohy, u komplexnějších zadání však může vykazovat nižší přesnost. [20]
- **Few-shot prompting:** Pro zvýšení konzistence a přesnosti se modelu v rámci promptu poskytne několik (typicky 2 a více) názorných příkladů (vzorů). Tento přístup pomáhá modelu identifikovat požadovanou strukturu a logiku výstupu, což je klíčové zejména u úloh s pevně daným formátem nebo specifickou doménovou znalostí. [20]

Samostatnou kategorii tvoří technika **Chain of Thought (CoT)**, neboli řetězec úvah. Jejím principem je dekompozice složitého problému na sekvenci jednodušších podúloh, které model řeší postupně. Namísto přímého generování výsledku je model stimulován k artikulaci mezikroků. [22]

**Příklad:** Při výpočtu součinu $54 \times 36$ model nejprve vypočítá dílčí operace ($54 \times 30$ a $54 \times 6$) a až následným součtem těchto hodnot dospěje ke konečnému výsledku.

Tato technika signifikantně zvyšuje úspěšnost modelů v oblasti logického uvažování a matematických operací.

## 3.5 Formy nasazení a integrace LLM

Využití velkých jazykových modelů v praxi se liší podle míry kontroly nad daty, technické náročnosti a požadavků na zabezpečení. Zatímco předchozí kapitoly popsaly, jak modely fungují vnitřně, tato část se zaměřuje na způsoby, jakými jsou tyto modely zpřístupněny koncovým uživatelům a systémům.

### 3.5.1 Webová rozhraní a API služby

Nejrozšířenějším způsobem interakce s LLM jsou **spotřebitelské webové aplikace** (např. ChatGPT, Claude), které jsou optimalizovány pro přímý dialog. Pro firemní integraci do vlastních softwarových řešení se však využívají **API (Application Programming Interface)**. Tato rozhraní umožňují automatizované odesílání dotazů a přijímání odpovědí, což je klíčové pro vývoj aplikací třetích stran. API služby jsou obvykle zpoplatněny na základě objemu přenesených dat (tokenů).

### 3.5.2 Open-source vs. Closed-source modely

Zásadním rozhodnutím při volbě metody práce s LLM je volba mezi uzavřeným a otevřeným modelem: [23]

- **Closed-source (Proprietární modely):** Modely jako GPT-4 nebo Gemini jsou poskytovány jako služba. Uživatel nemá přístup k vahám modelu ani k tréninkovým datům, což zaručuje vysoký výkon bez nutnosti vlastního hardwaru, ale přináší rizika v oblasti ochrany soukromí a závislosti na poskytovateli.
- **Open-source (Otevřené modely):** Modely jako Llama nebo Mistral umožňují stažení a provoz na vlastní infrastruktuře (tzv. on-premise). To poskytuje plnou kontrolu nad daty a možností hloubkového přizpůsobení, vyžaduje však značný výpočetní výkon.

### 3.5.3 Rozšiřující rámec: RAG a Agenti

Kromě přímého dotazování se dnes prosazují komplexnější metody, které eliminují nedostatky LLM (např. halucinace nebo neznalost aktuálních dat): [24]

- **RAG (Retrieval-Augmented Generation):** Tato metoda propojuje LLM s externí databází dokumentů. Model před vygenerováním odpovědi vyhledá relevantní informace v poskytnutých zdrojích (např. firemních směrnicích) a na jejich základě sestaví přesnou odpověď.
- **Agentní ekosystémy:** Jak bylo zmíněno v sekci 3.2.2, agenti nepředstavují pouze model, ale celý systém schopný samostatného plánování. V praxi to znamená, že agent může autonomně vyhledat informace na internetu, spustit kód v Pythonu pro analýzu dat nebo komunikovat s jinými API, aby splnil komplexní zadání uživatele.

## 3.6 Extrakce strukturované informace z textu

Extrakce pomocí LLM využívá schopnosti modelu rozumět sémantice textu k identifikaci specifických entit a vztahů mezi nimi. Na rozdíl od tradičních metod (RegEx nebo starší NLP modely) není extrakce závislá na pevných pravidlech, ale na kontextuálním pochopení. Pro výstup se používají 2 různé techniky:

- Extrakce založená na promptech - využívá strategie zvané *In-Context learning* zmíněné v sekci 3.4.1
- Volání funkcí, či služeb - využívá agentních systému pro vytvoření výsledku

Technika zakládající se na promptech má ještě 2 metody pro výstup, pokud požadujeme strukturovaný výsledek. První z nich je bez poskytnuté struktury. Pro výstup pouze popíšeme v jakém formátu požadujeme výsledek, ale přesnou strukturu již nespecifikujeme, např. pouze definujeme JSON formát, ale nikoli pak jeho klíče. Tato metoda slouží pro analytické účely, kdy se hledají různé vzorce. Naopak pokud potřebuje extrahovat data do přesně specifických formátů, předáme modelu tuto strukturu modelů, který poté do ní extrahuje data. Typické využití je u softwarových produktů. [25]

### 3.6.1 NER

Tradiční přístupy k rozpoznávání pojmenovaných entit (NER) jsou založeny na sekvenčním označování tokenů a vyžadují rozsáhlá anotovaná trénovací data. V posledních letech se objevuje alternativní přístup využívající velké jazykové modely, který je často označován jako *LLM-based information extraction* nebo *prompt-based NER*.

V tomto přístupu není extrakce realizována explicitním štítkováním tokenů, ale generativně, na základě instrukcí zadaných v podobě promptu. Model je schopen přímo generovat strukturovaný výstup obsahující extrahované entity a jejich vztahy. Tento přístup nabízí vysokou flexibilitu a umožňuje provádět extrakci informací i v prostředí s omezenou dostupností anotovaných dat, což je typické například pro oblast lékařských zpráv.

### 3.6.2 Validace výstupů a post-processing

Kritickým aspektem práce s velkými jazykovými modely je jejich **nedeterminismus**. Tento jev způsobuje, že model při identickém vstupu (promptu) generuje v čase odlišné odpovědi, což je dáno pravděpodobnostní povahou výběru následujících tokenů. V produkčním prostředí je proto nezbytné výstupy systematicky validovat, aby se eliminovaly negativní jevy, jako jsou **halucinace** (generování fakticky nesprávných, ale přesvědčivě znějících informací), logické nepřesnosti nebo neúplnost dat. [26]

K řešení těchto nedostatků se využívá metoda **post-processingu**. Jedná se o sekvenční proces validací a transformací, kterému je surový výstup modelu podroben předtím, než je předán koncovému uživateli nebo navazujícímu systému. Typické kroky post-processingu zahrnují: [27]

- **Syntaktická kontrola:** Ověření, zda výstup odpovídá požadovanému formátu (např. validace struktury JSON schématu).
- **Sémantická validace:** Kontrola logické konzistence dat a odhalování faktických rozporů.
- **Čištění dat:** Odstranění nadbytečného textu, formátování nebo normalizace entit získaných během extrakce.

# Kapitola 4

# AI ve zdravotnictví

Rozvoj umělé inteligence (AI) nabral v posledních letech nezastavitelnou rychlost a její integrace do klíčových odvětví se stává neodmyslitelnou součástí moderní společnosti. Zdravotnictví představuje disciplínu, v níž dochází k neustálému pokroku, a proto integrace AI v tomto sektoru patří k zásadním posunům lidstva k lepší budoucnosti. Vhodně nastavená politika podpory těchto technologií může zvýšit rovnost v přístupu k péči, zlepšit její kvalitu a zajistit, aby nové léčebné metody byly přínosem pro celou společnost. [28]

Ačkoliv se AI tradičně promítá zejména do oborů, jako jsou radiologie a diagnostika, její role se v posledních desetiletích fundamentálně proměnila. Zatímco dříve byla vnímána především jako nástroj pro analýzu obrazových dat, s nástupem velkých jazykových modelů (LLM) se těžiště zájmu a inovačního potenciálu přesouvá k **automatizaci administrativy a analýze textových informací** [28, 29]. Dle odhadů analytické společnosti IDC a studií publikovaných v Healthcare Informatics Research tvoří nestrukturovaná data, jako jsou volné texty lékařských zpráv či klinické poznámky, až 80 % veškerého objemu dat generovaných v rámci zdravotní péče [30].

## 4.1 Evoluce zpracování medicínských dat

Historické kořeny AI ve zdravotnictví sahají k tzv. **expertním systémům** ze 70. let (např. systém MYCIN), které byly založeny na pevně definovaných pravidlech typu „if-then“. Stroj celkem obsahoval kolem 500 pravidel, nicméně tento stroj byl měl zhruba stejné schopnosti jako specialista. Tyto systémy však narážely na rigiditu a neschopnost pracovat s nejednoznačností přirozeného jazyka. [31]

S masivní digitalizací zdravotnictví a zavedením elektronických zdravotních záznamů (EHR) vyvstala potřeba efektivně vytěžovat data, která lékaři zapisují ve formě volného textu. Moderní přístup se proto odklonil od manuálně psaných pravidel k **strojovému učení**, které umožňuje modelům identifikovat vzorce v datech autonomně. Pro extrakci informací to znamenalo přechod od jednoduchého vyhledávání klíčových slov ke komplexnímu chápání sémantiky zpráv. [29]

## 4.2 NLP jako most mezi lékařem a strojem

Zpracování přirozeného jazyka (NLP) představuje v kontextu extrakce lékařských zpráv nezbytnou část. Lékařský text je specifický svou vysokou hustotou informací, používáním

nestandardních zkratek, latinské terminologie a často i gramaticky neúplných vět. Dalším ztížením je český jazyk, který je morfologicky bohatý jazyk, což extrakci ztěžuje oproti angličtině [32].

Tradiční metody NLP se zaměřovaly především na úlohu Named Entity Recognition (NER) – tedy identifikaci entit, jako jsou diagnózy, medikace nebo laboratorní hodnoty. Starší generace modelů (např. založené na architektuře BERT) sice dosahovaly vysoké přesnosti, ale vyžadovaly rozsáhlé, ručně anotované datasety pro každou specifickou úlohu. Právě tato závislost na drahých expertních anotacích byla hlavní bariérou pro širší nasazení v praxi, což vyřešil až nástup LLM.

## 4.3 Průlom generativní AI a LLM v klinické praxi

Současná paradigma reprezentovaná modely jako GPT-4, Llama 3 či specializovaný Med-PaLM, přináší do extrakce dat revoluční změnu prostřednictvím tzv. **všeobecné schopnosti porozumění**. Na rozdíl od předchozích modelů vykazují LLM schopnost:

- **Zero-shot extrakce:** Model dokáže extrahovat informace z lékařské zprávy (např. vypsat všechny alergie pacienta), aniž by k tomu potřeboval předchozí trénink na konkrétním typu zpráv.
- **Kontextuální syntéza:** LLM nerozeznávají pouze izolovaná slova, ale chápou vztahy mezi nimi (např. rozliší, zda je lék pacientovi předepisován, nebo zda byl naopak vysazen).
- **Standardizace výstupu:** Schopnost převést nestrukturovaný text přímo do formátů vhodných pro další zpracování (např. JSON nebo tabulkový procesor), což je pro automatizaci lékařské administrativy klíčové.

## 4.4 Etické aspekty a limity nasazení

Navzdory vysoké efektivitě naráží nasazení LLM v medicíně na specifické bariéry. Prvním kritickým bodem je **vznik halucinací**, kdy model může s vysokou mírou přesvědčivosti vygenerovat medicínsky chybný údaj. V procesu extrakce to znamená riziko záměny negace (např. „pacient neguje bolest“ interpretováno jako „pacient má bolesti“).

Druhým pilířem je **ochrana citlivých údajů (GDPR)**. Většina pokročilých LLM je provozována jako cloudová služba, což u lékařských dat vyžaduje striktní procesy deidentifikace nebo nasazení lokálních (on-premise) modelů.

Budoucnost využití LLM v extrakci tak spočívá především v kombinaci lidského dohledu (*human-in-the-loop*) a specializovaných modelů trénovaných na doménově specifických datech. Ačkoliv AI přináší ve zdravotnictví plno pokroků, přichází i nové výzvy a překážky. Anonymizace a GDPR, neboli evropské nařízení o ochraně dat je ve zdravotnictví velké téma a vzniklo již několik nástrojů na anonymizaci dat. Pacienti musí být chráněni a nesmí být jejich osobní data předávané velkým jazykovým modelům. Dalším problémem jsou etické problémy při odpovědnosti za rozhodování. Vzhledem k již dosáhlému pokroku je otázkou publikace vědeckých článků vytvořených AI.

# Kapitola 5

# Data projektu MRE

## 5.1 Téma lékařských zpráv

Nedílnou součástí bakalářské práce bylo zpracování vstupních dat. Data pocházela z projektu MRE, který provozuje katedra KIV Fakulty aplikovaných věd V Plzni (https://mre.zcu.cz). a obsahovala anonymizované lékařské zprávy pacientů s Crohnovou chorobou a cévní mozkovou příhodou (mrtvicí). Jednalo se o reálná klinická data, která se výrazně lišila jak délkou jednotlivých záznamů, tak i jejich obsahem a zaměřením. Zprávy zahrnovaly různé typy lékařských vyšetření a záznamů, například vyšetření CT, MR, SONO, perfuzní vyšetření, ambulantní zprávy či hospitalizační záznamy.

## 5.2 Struktura a anonymizace

Z jazykového hlediska jsou lékařské zprávy charakteristické vysokou mírou odborné terminologie, častým výskytem zkratek, latinských názvů, nejednoznačných formulací a stylistických i pravopisných nekonzistencí. Texty rovněž obsahují neúplné věty, telegrafický styl zápisu a kombinaci strukturovaných i nestrukturovaných částí, což výrazně zvyšuje náročnost automatické extrakce informací.

Zprávy byly vedoucím poskytnuty ve formátu CSV. Jednotlivá data byla zpracována tak, aby každý CSV soubor měl tyto entity: *url* – odkaz na webový server projektu MRE s danou zprávou, *datetime* – čas vyšetření, *title* – název vyšetření a *text* – text lékařské zprávy. U lékařských zpráv zabývajících se mrtvicí nebyly poskytnuty URL adresy.

Lékařské zprávy byly předem zpracovány a anonymizovány v rámci projektu MRE Fakulty aplikovaných věd. Z osobních údajů byl u zpráv s Crohnovou nemocí uveden pouze věk pacienta.

## 5.3 Testovací data

Při hodnocení správnosti extrakce nebyly k dispozici předem připravené testovací ani referenční (*ground truth*) soubory. Z tohoto důvodu nebylo možné provést klasické vyhodnocení pomocí přesnosti a úplnosti vůči zlatému standardu. Místo toho byly navrženy vlastní metriky umožňující relativní porovnání výsledků extrakce napříč vybranými velkými jazykovými modely (LLM). Tyto metriky se zaměřovaly především na konzistenci výstupů, úplnost extrahovaných informací a jejich shodu s obsahem původního textu.

# Kapitola 6

# Návrh řešení

## 6.1 Celkový koncept

Projekt byl rozdělen do tří hlavních částí: **předzpracování dat**, **vytvoření metrik** a **evaluace výsledků**. Jednotlivé části byly dále členěny na dílčí podúkoly, které na sebe logicky navazovaly. Celkový koncept řešení je schematicky znázorněn na obrázku # a v následujících podkapitolách je podrobně rozebrán.

Před samotnou extrakcí bylo nutné provést základní úpravy dat, zejména sjednotit strukturu CSV souborů. Jelikož nebyly poskytnuty testovací data, bylo potřeba vybrat zprávy s největším obsahem informací. U pacientů s Crohnovou chorobou bylo upřednostňováno téma návštěvy lékaře (ambulantní zprávy), jelikož tento typ dokumentů obsahoval největší množství textových informací a zároveň nejširší spektrum strukturovaných údajů, jako jsou diagnózy, medikace, laboratorní výsledky a doporučení. Naopak u zpráv týkajících se cévní mozkové příhody nebylo žádné konkrétní téma preferováno, neboť jednotlivé záznamy byly tematicky homogennější a často se vztahovaly k zobrazovacím metodám mozku.

### 6.1.1 Předzpracování dat

Před samotnou extrakcí informací bylo nezbytné provést základní předzpracování vstupních dat. Hlavním cílem této fáze bylo sjednocení struktury poskytnutých CSV souborů a příprava dat tak, aby byla vhodná pro další automatické zpracování.

Vzhledem k tomu, že nebyla k dispozici samostatná testovací ani validační data, bylo nutné provést výběr reprezentativních lékařských zpráv s co nejvyšším informačním obsahem. Kritériem výběru byl zejména rozsah textu a množství klinicky relevantních informací.

U pacientů s Crohnovou chorobou bylo upřednostněno téma návštěvy lékaře, konkrétně ambulantní zprávy. Tyto dokumenty se ukázaly jako nejvhodnější, neboť obsahovaly největší množství textových informací a zároveň široké spektrum strukturovaných údajů, jako jsou diagnózy, medikace, laboratorní výsledky a doporučení pro další léčbu.

Naopak u zpráv týkajících se cévní mozkové příhody nebylo žádné konkrétní téma preferováno. Jednotlivé záznamy byly obsahově homogennější a převážně se vztahovaly k výsledkům zobrazovacích metod mozku, zejména CT a MR vyšetřením, což umožňovalo jejich použití bez další tematické filtrace.

### 6.1.2 Metriky

#### Vytvoření metriky

Vytvoření vhodných metrik pro hodnocení kvality extrakce informací představovalo jeden z nejnáročnějších aspektů této práce. Standardně používané metriky, jako je přesnost (*precision*), úplnost (*recall*), F1 skóre nebo regresní metody, nejsou v kontextu této práce snadno aplikovatelné. Důvodem je absence referenčních anotovaných dat (tzv. *ground truth*), stejně jako vysoká variabilita a nestrukturovanost vstupních lékařských textů.

Z těchto důvodů byly navrženy vlastní, převážně kvalitativní a kvantitativně–deskriptivní metriky, jejichž cílem není absolutní hodnocení správnosti, ale **relativní porovnání chování jednotlivých velkých jazykových modelů (LLM)** napříč různými aspekty extrakce. Navržené metriky rozdělují hodnocení do několika vzájemně se doplňujících dimenzí, které umožňují identifikovat silné a slabé stránky jednotlivých modelů.

*Výsledky prezentované v této kapitole se v současné fázi vztahují výhradně k datům pacientů s Crohnovou nemocí. Rozšíření metrik a jejich aplikace na další diagnózy, zejména cévní mozkovou příhodu, je plánováno jako součást další práce.*

#### Anotace

Klíčovým předpokladem pro konstrukci metrik bylo vytvoření anotovaného slovníku pojmů, na jejichž výskyt se hodnocení extrakce zaměřuje. Anotace slouží jako referenční rámec, nikoliv jako úplný zlatý standard, a umožňuje sledovat, zda model dokáže identifikovat a zachovat významově důležité informace.

Anotovány byly zejména následující typy výrazů:

- **strukturální prvky textu**, jako jsou názvy sekcí (např. *subj*, *obj*, *doporučení*),
- **Často vyskytující se slova u Crohnovi nemoci** (např. *CRP, kalprotectin, ileum*),
- **lékařské a latinské termíny**, včetně názvů diagnóz, léčiv a anatomických struktur,
- **klinické příznaky a popisy zdravotního stavu**.

Tato anotace umožnila následně kvantifikovat, jakým způsobem jednotlivé modely s těmito pojmy pracují, zda je zachovávají, modifikují, či zcela opomíjejí.

#### Dimenze hodnocení extrakce

Navržené metriky jsou rozděleny do několika základních dimenzí, které reflektují různé aspekty kvality výstupů:

- porovnání výsledků mezi jednotlivými modely,
- konzistence výstupů při opakovaném zpracování,
- vliv použití cizího jazyka v promptu,
- vliv různých typů promptování.

### 6.1.3 Evaluace výsledků

Metriky vytvořené v tomto projektu sice slouží k nalezení silných a slabých stránek modelů, ale ne všechny metriky jsou schopné být automatizovány. Některé z těchto metrik musejí být ručně vyhodnocovány soubor po souboru, což mohlo vést k různým odchylkám. Porovnání modelů na základě výsledků z testování bude vedeno ručním vyhodnocením a většina závěru bude záviset na slovním popisu výhod a nevýhod vybraných LLM.

## 6.2 Volba modelů

Výběr jazykových modelů byl v úvodní fázi rozdělen do tří základních kategorií: **obecné modely**, **lékařsky orientované modely** a **modely zaměřené na český jazyk**. Cílem tohoto rozdělení bylo pokrýt jak široce používané velké jazykové modely bez doménového zaměření, tak i specializované modely optimalizované pro zdravotnickou oblast nebo český jazyk.

Obecné modely byly chápány jako rozsáhlé korporátní jazykové modely využívané celosvětově, které nejsou explicitně zaměřeny na konkrétní doménu. Naopak lékařské a české modely měly představovat doménově specifická řešení, potenciálně lépe přizpůsobená zpracování odborných textů nebo textů v českém jazyce.

Při podrobnějším průzkumu dostupných modelů však bylo zjištěno, že většina lékařsky orientovaných jazykových modelů je buď **komerčně placená**, nebo **přístupná pouze na základě zvláštního povolení ze strany vydavatele**. Dalším významným omezením bylo, že tyto modely často nebyly primárně navrženy pro úlohy strukturované extrakce informací, ale spíše pro generování nebo doplňování textu, případně konverzační využití.

Podobná situace nastala také u modelů zaměřených výhradně na český jazyk. Počet dostupných a veřejně přístupných modelů trénovaných specificky na českých datech je velmi omezený a žádný z nich nenabízel dostatečnou podporu pro úlohy extrakce strukturovaných informací z textu. Z tohoto důvodu nebylo možné tyto modely efektivně zahrnout do experimentální části práce.

Na základě uvedených omezení byly proto do dalšího zkoumání vybrány pouze **obecné velké jazykové modely**, které v současnosti patří mezi nejrozšířenější a nejvýkonnější dostupná řešení. Výběr konkrétních modelů byl proveden na základě průzkumu odborných a technologických zdrojů, dostupnosti modelů a jejich schopností práce s delšími texty a strukturovanými výstupy.

Vybrány proto byly jen obecné a to nejrozšířenější modely současnosti. Po průzkumu několika webových stránek byly vybrány následující modely:

| **Brand** | **Name of LLM** | **Model** |
|---|---|---|
| Anthropic | Claude.ai | Sonnet 4.5 |
| OpenAI | ChatGPT | GPT-5 |
| Mistral AI | Le Chat | Mistral Large / Pixtral Large / Mixtral |
| Google | Gemini | 2.5 Flash / 3.0 Pro |
| xAi | Grok | Grok 4 |
| Deepseek | Deepseek | DeepSeek-V3.2 |
| Meta | Llama | Llama-3.2-3B |

Tabulka 6.1: Vybrané velké jazykové modely

# Kapitola 7

# Implementace prototypu

## 7.1 Metodika a použité nástroje

Tato kapitola popisuje softwarové vybavení, vývojové prostředí a využité programové knihovny, které tvořily technologický základ pro realizaci praktické části práce.

### 7.1.1 Vývojové prostředí

"""Pro implementaci veškerých algoritmů a uživatelského rozhraní bylo využito integrované vývojové prostředí (IDE) **Microsoft Visual Studio Code (VS Code)**""". Volba tohoto prostředí byla podložena jeho vysokou modularitou a širokou podporou pro jazyk **Python**, který byl zvolen jako primární programovací jazyk. Python v současnosti představuje standard v oblastech strojového učení, analýzy dat a integrace modelů umělé inteligence, a to především díky rozsáhlému ekosystému knihoven a podpoře rozhraní API.

### 7.1.2 Softwarové nástroje a platformy

#### Jupyter Notebook

Pro fázi analýzy dat a prototypování funkcí určených k předzpracování (preprocessing) byl využit nástroj **Jupyter Notebook**. Tento nástroj umožňuje interaktivní spouštění bloků kódu, což usnadňuje ladění a vizualizaci dat v reálném čase.

#### Doccano

Pro účely vytvoření evaluačních metrik a trénovacích dat bylo nezbytné provést manuální anotaci lékařských zpráv. K tomuto účelu byl zvolen open-source nástroj **Doccano**. Z důvodu izolace závislostí a snadné replikovatelnosti prostředí byl tento anotační nástroj provozován v rámci kontejnerizační platformy **Docker**.

#### Knihovny jazyka Python a správa závislostí

Kromě standardních knihoven jazyka Python byly využity specializované balíčky třetích stran. Správa těchto knihoven byla realizována prostřednictvím správce balíčků **pip**. Kompletní seznam všech využitých knihoven včetně jejich specifických verzí je pro účely reprodukovatelnosti uveden v souboru `requirements.txt`, který je součástí přílohové části práce.

## 7.2 Webová aplikace

Webová aplikace byla navržena jako podpůrný nástroj pro poloautomatickou generaci promptů, zpracování výstupů a analýzu lékařských zpráv. Vzhledem k absenci přímého napojení na API velkých jazykových modelů (LLM) nebylo možné proces plně automatizovat. Aplikace tak slouží primárně k optimalizaci pracovního toku (workflow) a usnadnění manuálních úkonů spojených s experimentální částí projektu.

### Použité technologie

Pro vývoj webového rozhraní byl zvolen jazyk Python, který nabízí řadu frameworků pro tvorbu webových aplikací (např. Flask, Django). Pro účely této práce byla vybrána knihovna **Streamlit**. Tento framework je optimalizován pro rychlý vývoj datově orientovaných aplikací a prototypování, což plně odpovídalo požadavkům na jednoduché, ale funkční uživatelské rozhraní bez nutnosti složité implementace backendové logiky.

### Funkcionalita aplikace

Aplikace je členěna do pěti samostatných modulů (stránek), které pokrývají jednotlivé fáze zpracování dat:

- **Data Viewer** – Modul pro vizualizaci a prohlížení zdrojových dat uložených v adresáři `data`. Pro tabulkové zobrazení datových sad je využita knihovna **Pandas**.
- **Prompt Maker** – Nástroj pro systematickou tvorbu a správu promptů, které jsou následně manuálně vkládány do testovaných LLM modelů.
- **Result Maker** – Slouží ke strukturovanému ukládání výstupů získaných z modelů do souborového systému.
- **Results Viewer** – Rozhraní pro prohlížení a kontrolu uložených výsledků experimentů.
- **Analyzator** – Modul provádějící analýzu lékařských zpráv, jehož výstupem jsou vypočtené evaluační metriky.

### Architektura aplikace

Vstupním bodem aplikace je soubor `main.py`, který zajišťuje inicializaci prostředí a navigaci. Jednotlivé funkční moduly (stránky) jsou implementovány v samostatných souborech umístěných v adresáři `st_src`. Komplexní logika a výpočetní operace, které přesahují rámec prezentační vrstvy, jsou vyčleněny do pomocných modulů v adresáři `src`, čímž je zajištěna přehlednost a udržitelnost kódu.

## 7.3 Metoda interakce s modely (prompting)

Při využívání vybraných jazykových modelů byla zvažována integrace pomocí aplikačního rozhraní **API**. Ačkoliv poskytovatelé těchto modelů standardně nabízejí knihovny pro přímou komunikaci v rámci vývojového prostředí, přístup k těmto rozhraním je v mnoha případech podmíněn zpoplatněním nebo komerční licencí.

Z tohoto důvodu byl zvolen proces manuálního zpracování dat:

1. V lokálním prostředí (VS Code) byly připraveny vstupní soubory obsahující testovací data a definované prompty.

2. Tato data byla následně vkládána do webových rozhraní příslušných modelů.

3. Získané odpovědi byly exportovány a následně zpracovány pomocí vlastní **webové aplikace v knihovně Streamlit**, která sloužila k unifikaci výsledků a jejich transformaci do finálních datových formátů pro další analýzu.

# Kapitola 8

# Zhodnocení dosažených výsledků

## 8.1

# Kapitola 9

# Závěr

## 9.1

# Literatura


[1] McCarthy, J. (2007). What is artificial intelligence?

[2] Russell, S., & Norvig, P. (2010). *Artificial Intelligence: A Modern Approach*. Prentice Hall.

[3] Formánek, I., & Farana, R. Artificial Intelligence–Artificial Neural Networks. *VŠPP Entrepreneurship Studies*, 24.

[4] SAP. What is Deep Learning? *AI in Business*. Dostupné z: https://www.sap.com/resources/what-is-deep-learning

[5] McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4), 115–133.

[6] Horký, L., & Břinda, K. (2009). Neuronové sítě.

[7] Pala, K. (2000). Počítačové zpracování přirozeného jazyka. *NLP FI MU*.

[8] SAP. What is Natural Language Processing? Dostupné z: https://www.sap.com/uk/resources/what-is-natural-language-processing

[9] Hendl, J. (2023). Jazykové modely a umělá inteligence. In *Sborník konference Medsoft 2023* (pp. 1–9).

[10] IBM. What is Transformer Model? Dostupné z: https://www.ibm.com/think/topics/transformer-model

[11] Hugging Face. LLM Course. Dostupné z: https://huggingface.co/learn/llm-course/chapter1

[12] SAP. What is Large Language Model? Dostupné z: https://www.sap.com/uk/resources/what-is-large-language-model

[13] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention Is All You Need. In *Advances in Neural Information Processing Systems* (Vol. 30).

[14] Sapien.io. Fine-Tuning vs Pre-Training: Key Differences for Language Models. Dostupné z: https://www.sapien.io/blog/fine-tuning-vs-pre-training-key-differences-for-language-models

[15] Eordaxd. Fine-Tuning vs Pre-Training. *Medium*. Dostupné z: https://medium.com/@eordaxd/fine-tuning-vs-pre-training-651d05186faf

[16] Kubíček.ai. LLM asi znáte, ale víte i o dalších typech modelů? Dostupné z: https://www.kubicek.ai/llm-asi-znate-ale-vite-i-o-dalsich-typech-modelu/

[17] MadAILab. 7 Types of Large Language Models (LLMs). *Medium*. Dostupné z: https://medium.com/madailab/7-types-of-large-language-models-llms-55fb0038ceb9

[18] YouTube. Mixture of Experts (MoE) Explained. Dostupné z: https://www.youtube.com/watch?v=FwOTs4UxQS4

[19] FlowHunt. Agentic AI Systems. Dostupné z: https://www.flowhunt.io/cs/blog/agentic/

[20] Learn Prompting. Few Shot Prompting. Dostupné z: https://learnprompting.org/docs/basics/few_shot

[21] GeeksforGeeks. What is an AI Prompt? Dostupné z: https://www.geeksforgeeks.org/artificial-intelligence/what-is-an-ai-prompt/

[22] GeeksforGeeks. What is Chain of Thought Prompting? Dostupné z: https://www.geeksforgeeks.org/artificial-intelligence/what-is-chain-of-thought-prompting/

[23] Hatchworks. Open Source vs Closed LLMs Guide. Dostupné z: https://hatchworks.com/blog/gen-ai/open-source-vs-closed-llms-guide/

[24] NVIDIA Developer Blog. Traditional RAG vs Agentic RAG: Why AI Agents Need Dynamic Knowledge to Get Smarter. Dostupné z: https://developer.nvidia.com/blog/traditional-rag-vs-agentic-rag-why-ai-agents-need-dynamic-knowledge-to-get-sma

[25] Astera. LLM Data Extraction. Dostupné z: https://www.astera.com/type/blog/llm-data-extraction/

[26] FlowHunt. Defeating Non-Determinism in LLMs. Dostupné z: https://www.flowhunt.io/cs/blog/defeating-non-determinism-in-llms/

[27] Iterate.ai. Data Postprocessing Explained. Dostupné z: https://iterate.ai/ai-glossary/data-postprocessing-explained

[28] European Commission. Artificial Intelligence in Healthcare. Dostupné z: https://health.ec.europa.eu/ehealth-digital-health-and-care/artificial-intelligence-healthcare_cs

[29] Hirani, R., Noruzi, K., Khuram, H., Hussaini, A. S., Aifuwa, E. I., Ely, K. E., Lewis, J. M., Gabr, A. E., Smiley, A., Tiwari, R. K., & Etienne, M. (2024). Artificial Intelligence and Healthcare: A Journey through History, Present Innovations, and Future Possibilities. *Life*, 14(5), 557. https://doi.org/10.3390/life14050557

[30] Kong, H.-J. (2019). Managing Unstructured Big Data in Healthcare System. *Healthcare Informatics Research*, 25(1), 1–2. https://doi.org/10.4258/hir.2019.25.1.1

[31] Copeland, B. (2018, November 21). MYCIN. *Encyclopedia Britannica*. Dostupné z: https://www.britannica.com/technology/MYCIN

[32] Zikmundová, L. (2021). *Problematika morfologie ve vybraných popularizačních příručkách o českém jazyku* (diplomová práce). Západočeská univerzita v Plzni. Dostupné z: http://hdl.handle.net/11025/45299

## 9.2 Přílohy

### 9.2.1 Struktura projektu MRE

- **data** – v adresáři jsou uloženy vybrané texty z CSV souborů, popisy pro modely, poskytnuté CSV soubory, anotované slova a texty určené k promptování (prompty)
- **docs** – zde se uchovává dokumentace a všechny doplňující informace k provozu projektu
- **results** – adresář s výsledkami modelů
- **scripts** – využívá konfigurační soubory a soubor pro uchování adres a konstant
- **src** – obsahuje většinu použitých kódů a programů využitých při práci na projektu MRE
- **st_src** – obsahuje kódy pro webovou aplikaci

### 9.2.2 Struktura složky `data`

Adresář `data` uchovává všechna potřebná data k vypracování projektu MRE. Adresář obsahuje 5 podadresářů:

- **csv** – obsahuje strukturované lékařské zprávy poskytnuté vedoucím ve formátu CSV
- **doccano** – extrahovaná slova z Doccana rozřezaná do textových souborů
- **prompts** – soubory ve formátu JSON určené na komunikaci s modely (prompting)
- **tasks** – texty s instrukcemi pro LLM
- **medical_reports** – vybrané lékařské zprávy ze souborů CSV