VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

Konvoluční neuronové sítě

Kontrolní zpráva

Členové týmu:

Michal Pyšík {xpysik00} (vedoucí týmu)

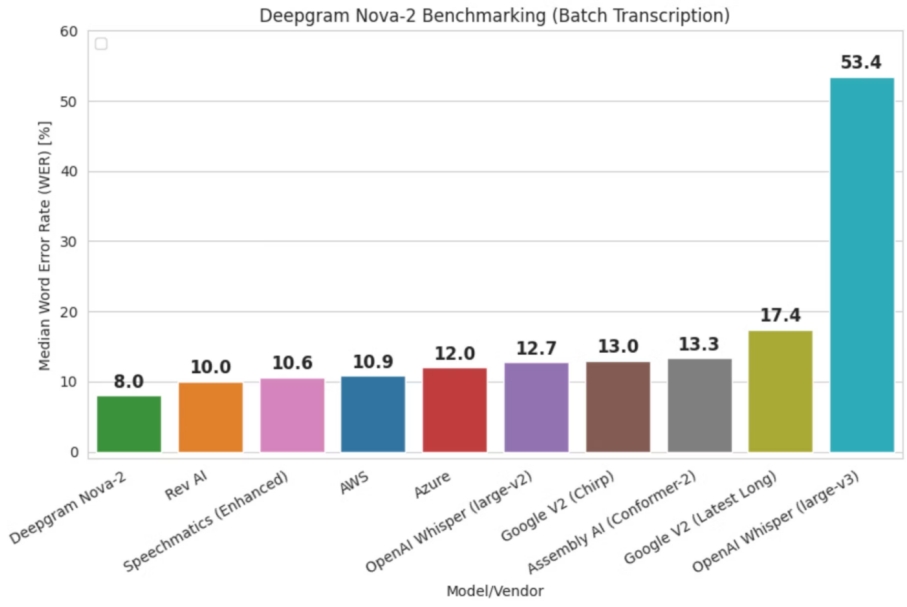
Maxim Plička {xplick04}

Michal Bartošák {xbarto0d}

Github repozitář: https://github.com/MichalPysik/KNN_project

1 Popis problému

Cílem našeho projektu je prozkoumat možnosti zlepšení modelu pro rozpoznávání řeči Whisper. Ačkoli je Whisper v současné době jedním z nejpoužívanějších nástrojů pro přepisování zvukových nahrávek, stále má svá úskalí. Jedním z aktuálně největších problémů dnešního Whisperu je problém halucinací.



Obrázek 1: Srovnání jednotlivých modelů pro rozpoznávání řeči z hlediska WER [4]

Vzhledem k tomu, že vědecká oblast nedospěla k jednotné ustálené definici tohoto pojmu, tak se v této práci budeme opírat o následující formulaci: Halucinace se jeví jako plynulé části výstupu, které na první pohled mohou působit věrohodně, ale ve skutečnosti nemají žádnou spojitost s původním nahrávkou [2]. Příklad takové halucinace může vypadat následovně:

- **Ground truth:** Someone had to run and call the fire department to rescue both the father and the cat.
- **Predikce:** Someone had to run and call the fire department to rescue both the father and the cat. **All he had was a smelly old ol' head on top of a socked, blood-soaked stroller.**

Takovéto výstupy pak mohou obsahovat nepřesné či zavádějící informace, které mohou zmást či dokonce oklamat koncového uživatele a z dlouhodobého hlediska mohou poškodit i samotné společnosti. Proto v tomto projektu budeme zkoumat vliv různých metod pro snížení výskytů halucinací v přepisovaných datech. K tomu bude potřeba vytvořit datovou sadu, na které bude zvolený model halucinovat. Tato datová sada bude nutná k tomu, abychom model na těchto datech mohli dotrénovat, případně upravit a zjistit, zda se vyhodnocování modelu na problematických datech zlepšilo. Na závěr se pokusíme navrhnout metodu pro detekci halucinací a analyzujeme různé přístupy, které použijeme k optimalizaci Whisperu tak, aby snížil svou náchylnost k halucinacím.

2 Související práce

Na problém halucinací jsme poprvé narazili v článku, který se zabýval automatickým přepisem zvukových nahrávek u osob trpících vadou řeči. Tento článek pro své vyhodnocení využíval jazykový model Whisper (zřejmě verzi 2), který u těchto nahrávek produkoval halucinace [4]. Po důkladné rešerši v rámci tohoto tématu jsme se zjistili, že na verzi modelu Whisper záleží. Článek *„Whisper-v3 Hallucinations on Real World Data“* [1] došel k závěru, že jeho poslední verze 3 (vydaná v listopadu roku 2023) produkuje data, která mají až 4x vyšší Word Error Rate než ostatní modely. Proto jsme se rozhodli používat právě tuto verzi.

Nejprve jsme se zaměřili na přípravu datasetu. Pro náš výzkum jsme chtěli využít dataset, který byl použit v článku s osobami trpícími vadou řeči, ale přístup k tomuto datasetu nám doposud nebyl poskytnut. Z tohoto důvodu jsme rozhodli prozkoumat další možnosti. Nakonec jsme zvolili tvorbu datasetu v rámci metody augmentace dat z článku [2], která bude blíže specifikována v sekci 3.

Dále jsme se zabývali vytvořením metody pro detekci halucinací, kterou jsme původně převzali z článku [2]. Metoda využívala kombinací 3 metrik, jenž měly za úkol zjistit chybovost, sémantickou správnost a plynulost přepisu. Tento přístup se však neosvědčil, proto jsme rozhodli použít vlastní metodu popsanou v sekci 4.

3 Příprava datasetu

Nalezení datasetu, na kterém by model často halucinoval, bylo obtížnější, než jsme očekávali. Problém jsme nakonec překonali pomocí vlastní augmentace dat, která dokázala halucinace vyvolat. Model má tendenci k halucinacím, pokud jsou v řeči přítomny pauzy, šum nebo jiný zvuk v pozadí. Tyto jevy se většinou vyskytují v nahrávkách delšího rozsahu. My se však zaměřujeme především na nahrávky kratší, které jsou zahrnuty v konvenčních anotovaných datasetech, protože věříme, že větší počet vzorků je vhodnější pro účely vyhodnocování (i pro budoucí trénování).

Vytvořili jsme jednoduchou augmentační metodu (`augment_short_audio()` v souboru `data_augmentation.py`), která nahrávku v náhodném bodě rozdělí na dvě části, mezi které vloží 3 až 30 sekund ticha. Kromě toho do výsledku vloží i náhodný šum (whitenoise, případně hluky z reálného světa, jako je například zvuk letadla). Při použití této metody bylo důležité zejména zajistit optimální úroveň hlasitosti šumu, aby nedošlo k významnému potlačení původní nahrávky [2]. V rámci augmentace se pracovalo s korpusem zvukových nahrávek LibriSpeech[1] [5] (test set, „other“ speech), který obsahuje delší jazykově náročnější nahrávky v anglickém jazyce. Samotný výběr datasetu však díky augmentační metodě není příliš podstatný.

4 Evaluace a baseline řešení

Pro evaluaci bylo zapotřebí navrhnout metodu automatické detekce halucinací. Při testování se však tato metoda ukázala být příliš obtížnou, jelikož není možné s jistotou rozlišit, kdy model halucinuje a kdy dochází pouze k chybnému přepisu (např. v rámci fonetických chyb).

Nejprve jsme implementovali evaluaci podle algoritmu z článku [2] (metoda `detect_hallucinations_article()` v souboru `hallucination_detection.py`), ale nepodařilo se nám dosáhnout přijatelných výsledků. Proto jsme se rozhodli zvolit jiný přístup, který vychází z principu KISS („Keep it simple, stupid!“). Přístup spočíval v ruční detekci halucinací, která nám poskytla empirické poznatky. Na základě těchto poznatků jsme poté vytvořili jednoduchou metodu `detect_hallucinations_simple()`, která slouží jako filtr pro identifikaci potenciálních halucinací ve výsledcích testování modelu. Metoda označuje jako potenciální halucinaci případy, kdy délka přepisu modelu překračuje délku skutečného přepisu a současně dosahuje metrika Word Error Rate alespoň hodnoty 0.05 (tento přístup dokáže ignorovat chyby typu „I am“ vs „I'm“). Metoda dále detekuje nejběžnější halucinace pomocí podřetězců. Jedná se o části textů, které se nacházejí pouze vygenerovaném výstupu (např. „bye“ na konci přepisu).

Na námi augmentované datové sadě se nám podařilo dosáhnout halucinace přibližně u každé patnácté nahrávky. To považujeme za více než dostatečné pro budoucí měření účinnosti potlačení problému halucinací. Nejčastějším typem identifikovaných halucinací byly spontánní vsuvky vzniklé přetrénováním modelu na Youtube videích („Thanks for

[1] `https://www.openslr.org/12`

watching“). Dalším typem pak bylo nutkavé doplňování vět způsobem navazujícím na předchozí část nahrávky. Za nejbizarnější nalezenou halucinaci považujeme následující:

- **Ground truth:** no doubt no doubt returned mister lilburn but if thy right eye offend thee pluck it out and cast it from thee for it is profitable for thee that one of my members should perish and not that thy whole body should be cast into hell
- **Predikce:** no doubt no doubt returned mr lilburn but if thy right eye offend thee pluck it out and cast it from thee for it is profitable for thee that one of my members should perish and not that **i hope you enjoyed this video if you did please like and subscribe and hit that bell icon so that you can get notified of my next video and i will see you in the next one by bye** body should be cast into hell

5 Plán pokračování v projektu

Pro nalezení vhodné metody pro zlepšení chování modelu z hlediska halucinací jsme prozkoumali práci [3], která se obecně zabývá jazykovými modely. Práce obsahuje sekci věnující se přístupům řešení halucinací, ze které budeme čerpat metody na úpravu architektury a trénování modelu. V rámci úpravy architektury byly uvedeny následující 3 metody. Metoda úpravy kodéru má za cíl zlepšit sémantickou interpretaci a reprezentaci vstupního textu. Metoda optimalizace mechanismu pozornosti upraví model tak, aby se kladl větší důraz na relevantní části ve vstupních sekvencích. Metoda úpravy dekodéru má za cíl zvýšení věrnosti generovaných tokenů a zároveň snížení pravděpodobnosti halucinačních tokenů. Mezi metody zmírňující halucinace modelu pomocí trénování pak patří posilované učení, víceúlohové učení nebo také metoda plánování a skicování, která omezuje generování výstupu na základě informací o obsahu a jeho pořadí. Pro řešení problému halucinací byly v tomto článku navrženy i další obecné přístupy jako regularizace a loss rekonstrukce. V další fázi tohoto projektu plánujeme některé z těchto metod implementovat.

Reference

[1] Francisco, J. N.: Whisper-v3 Hallucinations on Real World Data. `https://deepgram.com/learn/whisper-v3-results`, 2024, [Accessed 30-03-2024].

[2] Frieske, R.; Shi, B. E.: Hallucinations in Neural Automatic Speech Recognition: Identifying Errors and Hallucinatory Models. 2024, `arXiv:2401.01572`.

[3] Ji, Z.; Lee, N.; Frieske, R.; aj.: Survey of Hallucination in Natural Language Generation. *ACM Comput. Surv.*, ročník 55, č. 12, mar 2023, ISSN 0360-0300, doi:10.1145/3571730. Dostupné z: `https://doi.org/10.1145/3571730`

[4] Koenecke, A.; Choi, A. S. G.; Mei, K.; aj.: Careless Whisper: Speech-to-Text Hallucination Harms. 2024, `arXiv:2402.08021`.

[5] Panayotov, V.; Chen, G.; Povey, D.; aj.: Librispeech: An ASR corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, s. 5206–5210, doi:10.1109/ICASSP.2015.7178964.