

Framework pro úlohy vyžadující lidské posouzení

Adam Chýlek¹

1 Úvod

Během výzkumu a vývoje v oblastech umělé inteligence je u mnohých systémů zapotřebí lidského vyhodnocení jejich kvality. Mezi takové systémy patří např. syntéza mluvené řeči z textu, kde se stále uplatňuje subjektivní hodnocení vytvářených hlasů a objektivní míry nejsou běžně využívány. Jedná se nejčastěji o preferenční testy, v kterých je porovnávána kvalita výstupu jednoho systému proti výstupu jiného, tzv. MOS testy kde se hodnotí kvalita jednoho systému na škále od 1 do 5 nebo tzv. MUSHRA testy u kterých subjekty hodnotí kvalitu více systémů relativně vůči sobě na škále od 0 do 100.

Pro tyto účely bylo vytvořeno inovativní prostředí využívající moderní webové technologie. Výhodou vytvořeného frameworku oproti předchozím podobným nástrojům je především vysoká variabilita v možnostech definice zobrazení úloh. Díky tomu je možné využívat tento framework jak pro vyhodnocování, tak pro anotace či transkripce statických i interaktivních vzorků. V použití tak nejsme omezeni pouze na předdefinované typy testů, jako např. MUSHRA nebo MOS, ale aplikaci lze využít i pro mnohé další úlohy, mezi které patří hodnocení dialogových systémů, anotace emocí v řeči, segmentace obrazů apod.

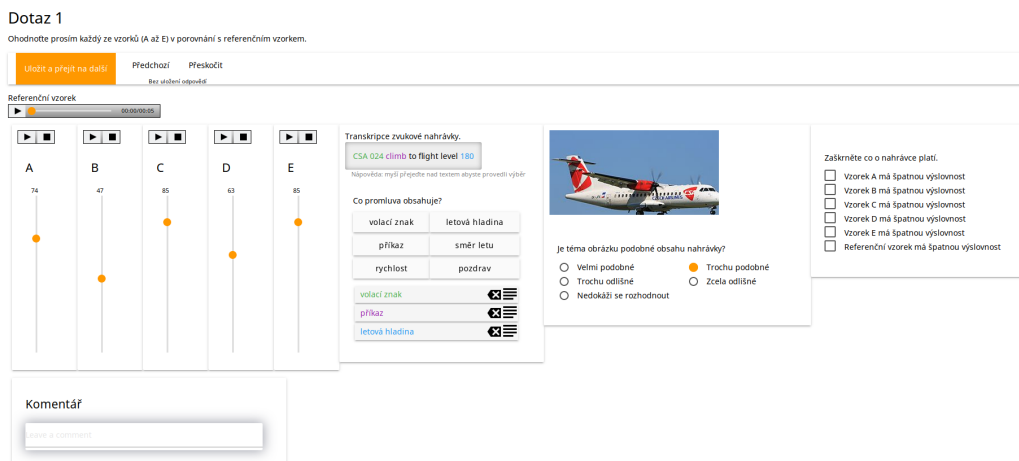
2 Vlastnosti frameworku

Frontend je tvořen webovou stránkou napsanou v jazyce TypeScript nad frameworkem Angular. Nevyžaduje instalaci žádných dodatečných nástrojů na straně uživatele a podporuje i starší prohlížeče jako Internet Explorer 10 (vydaný roku 2012). Backend této aplikace je napsán v PHP (kompatibilní s verzí minimálně 5.3) a pro uchování dat využívá MySQL databázi. Tyto technologie byly zvoleny z důvodu velmi rozšířené podpory u hostingových služeb a usnadnění přechodu ze stávajících nástrojů.

Důraz byl kladen nejen na rozšiřitelnost a kompatibilitu, ale také na možnost napojení na externí služby zajišťující oslovení a správu anotátorů, jako je např. Amazon Mechanical Turk. Systém umožňuje také jednoduchou správu uživatelů (registrace, statistiky dokončení, blokování) a stanovení podmínek pro přístup k danému testu. V případě hodnocení syntézy textu je tak možné stanovit omezení na řečníky, kteří během registrace zadali určitý mateřský jazyk. Omezení a informace, které uživatel musí zadat při registraci, stanovuje návrhář daného testu. Je tak možné během registrace požadovat zadání věku, informace o zkušenostech s anotátorskou činností apod.

V aplikaci je možno udržovat 2 databáze uživatelů - tzv. globální uživatele, kteří mají své registrační údaje sdíleny napříč testy a lokální uživatele, kteří jsou registrováni pouze pro konkrétní test a jejich přihlašovací a registrační údaje nejsou dostupné v jiných testech. Navíc rozlišujeme tzv. původ uživatele, např. že přistupuje ze služby Mechanical Turk. Na základě

¹ student doktorského studijního programu Aplikované vědy a informatika, obor Kybernetika, e-mail: chylek@students.zcu.cz



Obrázek 1: Frontend aplikace s ukázkou možných vstupních prvků

údajů z těchto databází je možné povolovat či zakazovat přihlašování nebo registraci konkrétní skupině uživatelů. Je tak možné provádět např. interní testy s předem ověřenou množinou uživatelů a zakázat přístup všem ostatním.

Základem pro návrh testu jsou *dotazy*. Každý *dotaz* je pro uživatele samostatnou stránkou, která obvykle obsahuje zkoumané *vzorky* a otázky s možnými *odpověďmi*, které jsou reprezentovány různými standardními vstupními prvky (obr. 1). Uživatelé pak postupně *dotazy* zobrazují a *odpovědi* se zaznamenávají do databáze pro pozdější vyhodnocení.

V současné době lze jako *vzorky* použít zvukový soubor, obrázek či text. Textový vzorek navíc umožňuje výběr a anotaci částí daného textu.

Jako prvky pro *odpovědi* je možné použít textová pole, zaškrávací pole, přepínače, posuvníky a tlačítka (ta navíc mohou být ve stromové struktuře).

Aby bylo možné vytvářet komplexní avšak přehledné testy je možné jednotlivé *vzorky* i *odpovědi* sdružovat do *kategorí* a *skupin*. Těmto prvkům je pak možné stanovit různou orientaci (horizontální, vertikální) či zalamování (např. vynutit 2 zaškrávací pole na řádek) a zarovnání (volně či do mřížky).

Samotný soubor s definovanými *dotazy* a dalšími parametry testu (jako např. umístění databáze či informace požadované od subjektů při registraci) jsou uchovávány v JSON objektu uloženého do textového souboru. Na jeho základě pak frontend generuje grafické rozhraní. Pokud návrhář vytváří často testy stejného typu lze generování tohoto souboru automatizovat, avšak tato automatizace již není součástí frameworku. Podobně také vyhodnocení zaznamenaných odpovědí je třeba provést mimo aplikaci na základě informací uchovaných v databázi.

3 Závěr

Vytvořený framework pro úlohy vyžadující lidské posouzení byl úspěšně nasazen pro účely vyhodnocení kvality systémů syntézy řeči z textu na Katedře kybernetiky ZČU a ve společnosti Nuance a v oblasti vyhodnocení dialogových systémů. Aplikace je dále aktivně rozšiřována, v nejbližší době tak bude možné využít jako vzorky např. soubory s videem. Široké možnosti sestavování testů pak vybízí k použití v dalších oblastech vývoje a výzkumu.

Poděkování

Příspěvek byl podpořen grantovým projektem SGS-2016-039.