

Оптимизација питања за квизове уз помоћ вештачке интелигенције***Optimization of Quiz Questions Using Artificial Intelligence***Милош Јосиповић, *Факултет техничких наука, Нови Сад***Студијски програм – РАЧУНАРСТВО И АУТОМАТИКА**

Кратак садржај – У овом раду представљена је web апликација за креирање и решавање квизова која интегрише могућности вештачке интелигенције у образовни процес. Посебан акценат стављен је на аутоматско генерисање питања за квизове помоћу вештачке интелигенције, чиме се значајно убрзава и поједностављује процес њихове припреме. Поред тога, имплементирани су и додатне функционалности засноване на вештачкој интелигенцији, са циљем унапређења наставног процеса, побољшања квалитета учења и повећања ефикасности рада како наставника, тако и ученика.

Кључне речи (три до пет): вештачка интелигенција, оптимизација, образовни процес, квиз

Abstract – This paper presents a web application for creating and taking quizzes that integrates artificial intelligence into the educational process. Particular emphasis is placed on the automatic generation of quiz questions using AI, significantly accelerating and simplifying the quiz preparation process. In addition, several AI-powered functionalities have been implemented with the aim of enhancing the teaching and learning experience, improving educational outcomes, and increasing the efficiency of both teachers and students.

Keywords: (three to five): artificial intelligence, optimization, educational process, quiz

НАПОМЕНА: Овај рад проистекао је из мастер рада чији ментор је био др Срђан Вукмировић, ред. проф.

1. УВОД

Креирање квалитетних питања за едукативне квизове захтева знање из предметних области, педагошко искуство и значајно време. Традиционални приступ подразумева да наставник осмисли и ручно формулише свако питање, дефинише понуђене одговоре и дефинише тачан одговор за дато питање. Овај процес је временски захтеван и подложен грешкама, посебно приликом креирања већег броја питања.

Напредак у области вештачке интелигенције, а посебно великих језичких модела, отворио је могућност аутоматизације овог процеса. Модели попут

GPT-4.1 способни су да на основу кратког описа теме генеришу питања различитих типова, процењују ниво тежине и идентификују потенцијалне проблеме у формулисању питања. Ове способности могу значајно убрзати креирање образовног садржаја.

Наведене способности ће бити представљене у посебно за то креираној апликацији *Kvizio*. Осим приказа могућности вештачке интелигенције у овом контексту, биће извршена упоредна анализа приступа са употребом вештачке интелигенције за креирање квизова и без ње.

2. УВОД У ВЕШТАЧКУ ИНТЕЛИГЕНЦИЈУ

Вештачка интелигенција представља област рачунарских наука која се бави развојем система способних да обављају задатке који традиционално захтевају људску интелигенцију, попут разумевања језика, препознавања образаца, доношења одлука и генерисања садржаја. Нагли напредак у овој области током последње деценије, поготово развој великих језичких модела заснованих на трансформер архитектури, отворио је могућности примене које су до недавно биле незамисливе: од аутоматског превођења и писања кода до генерисања образовног садржаја и пружања персонализоване помоћи корисницима у реалном времену [1].

2.1. Историјат развоја вештачке интелигенције

Вештачка интелигенција као научна дисциплина формално је установљена 1956. године на Дартмут конференцији, на којој је група истраживача, предвођена Џоном Макартијем, први пут употребила овај термин и дефинисала област чији је циљ креирање машина способних да симулирају аспекте људске интелигенције [2]. Корени идеје сежу и даље од тада, већ 1950. године Алан Тјуринг предложио је чувени Тјурингов тест као критеријум за процену интелигенције машине [3].

Развој вештачке интелигенције кроз деценије обележен је наизменичним периодима ентузијазма и стагнације, познатим као „AI зиме“. То су периоди разочарења и губитка финансирања услед пада интересовања за област због недостигнутих очекивања [1].

Кључни преокрет у модерној ери вештачке интелигенције десио се 2012. године када је мрежа *AlexNet* заснована на дубоким конволуционим неуронским мрежама драстично побољшала резултате

на *ImageNet* такмичењу у препознавању слика [4]. Овај успех покренуо је лавину истраживања и инвестиција у област дубоког учења.

2.2. Велики језички модели

Прекретницу у области обраде природног језика представио је рад „*Attention is All You Need*“ који је увео трансформер архитектуру. Овај рад је објављен 2017. године и написан је од стране осам научника и инжењера запослених у компанији *Google*. Трансформер архитектура заснована је на механизму пажње, који сваком токenu (речи или подречи) у низу дозвољава да директно „обрати пажњу“ на сваки други token, без обзира на растојање у низу. Ово омогућава паралелну обраду читавог улазног низа и ефикасно моделовање контекста [5].

GPT (Generative Pre-trained Transformer) је серија великих језичких модела које развија компанија *OpenAI*. Модели су тренирани на огромним корпусима текста, са циљем предвиђања следећег токена у низу. Повећањем количине података за обуку и броја параметара модела, *GPT* модели показују способност разумевања контекста, генерисања кохерентног текста и извршавања широког спектра задатака без експлицитног додатног тренирања за сваки појединачни задатак.

GPT-4 представља значајно унапређење у односу на претходне генерације модела, са бољим разумевањем сложених упита, већом тачношћу одговора и способношћу решавања комплекснијих задатака [6].

2.3. Промптно инжењерство

Промптно инжењерство је дисциплина која се бави обликовањем улазних захтева ка језичким моделима са циљем добијања жељених излаза.

Кључни елементи ефикасног *prompt*-а укључују:

- Системска порука – инструкција која дефинише улогу модела, тон, језик и општа ограничења
- Јасна спецификација излазног формата – језички модели могу генерисати излаз у *JSON*, *XML*, *Markdown* или обичном тексту. Експлицитним навођењем очекиваног формата у одговору значајно се повећава поузданост обраде одговора у аутоматизованим системима.
- *Few-shot* примери – навођење једног или више примера жељеног понашања у самој инструкцији помаже моделу да разуме тачна очекивања без потребе за финим подешавањем.
- Ограничења и забране – експлицитно навођење шта модел не сме да ради смањује вероватноћу нежељених излаза [7].

2.4. Примена вештачке интелигенције у образовању

Примена вештачке интелигенције у образовању представља интердисциплинарни област која проучава употребу технологија вештачке интелигенције у циљу унапређења процеса учења, наставе и евалуације знања [8].

Савремени системи вештачке интелигенције користе се за аутоматску евалуацију и оцењивање знања, укључујући писање одговора и пружање непосредних повратних информација. Поред тога, генеративни модели вештачке интелигенције омогућавају аутоматско генерисање образовних материјала, питања за проверу знања, сажетака и додатних објашњења, чиме се смањује оптерећење наставника и повећава доступност образовних ресурса [9].

Истраживања показују да персонализоване повратне информације и адаптивни приступи учењу могу позитивно утицати на ангажовање ученика, мотивацију и образовне исходе. Због тога се ове технологије све чешће интегришу у савремене образовне платформе као подршка учењу и континуираном праћењу напретка ученика [10].

3. KVIZIO АПЛИКАЦИЈА

3.1. Идеја и инспирација

Апликација је креирана у слободно време као хоби а инспирација је дошла због жеље за учењем о могућностима интеграције вештачке интелигенције у пројекте, те је тако створена идеја за израду мастер рада на поменутој тему.

3.2. Коришћене технологије

У реализацији софтверског решења за овај рад коришћене су следеће технологије и алати:

- *ASP.NET Core* – радно окружење отвореног типа које је развила компанија *Microsoft*, а користи се за развој *web* апликација.
- *Entity Framework Core* – објектно-релациони мапер за *.NET* који омогућава рад са релационим базама података коришћењем објеката уместо директног писања *SQL* упита. У пројекту је коришћен *Code-First* приступ који омогућава аутоматско управљање шемом базе путем миграција.
- *SQL Server Express* – бесплатан систем за управљање релационим базама података компаније *Microsoft*. Изабран је због једноставне интеграције са осталим *Microsoft* технологијама.
- *JSON Web Token (JWT)* – стандард за безбедан пренос података између клијента и сервера у облику дигитално потписаног токена. У пројекту је коришћен за реализацију *stateless* аутентификације и заштиту *API* ресурса.
- *AutoMapper* – библиотека која аутоматизује мапирање између објеката различитих типова. Његова примена смањује количину понављајућег кода и поједностављује конверзију између ентитета и *DTO* објеката.
- *React – JavaScript* библиотека за развој корисничких интерфејса заснованих на компонентама. У пројекту је коришћен за имплементацију клијентског дела апликације и изградњу динамичког корисничког интерфејса.
- *Material-UI (MUI)* – библиотека *React* компоненти заснована на *Material Design*

принципима. Омогућава брзу израду конзистентног и респонзивног корисничког интерфејса коришћењем готових компоненти.

- *Recharts* – *React* библиотека намењена визуелизацији података кроз графиконе.
- *Axios* – *JavaScript* библиотека за слање *HTTP* захтева и комуникацију са *REST API* сервисима.
- *OpenAI API* – омогућава приступ напредним моделима вештачке интелигенције за обраду и генерисање природног језика. Употребљен је за аутоматско генерисање питања за квизове и друге функционалности вештачке интелигенције.
- *Docker* – платформа за развој и извршавање апликација у изолованим контејнерима. Користи се за поједностављење развоја, дистрибуције и покретања комплетне апликације у различитим окружењима.

3.3. Архитектура апликације

Kvizio је реализован као вишеслојна *web* апликација заснована на клијент-сервер архитектури. Серверска страна излаже *RESTful API*, док клијентска страна користи тај *API* за преузимање и приказ података кориснику. Оваква архитектура обезбеђује јасно раздвајање одговорности и омогућава независни развој и скалирање сваке од тих страна.

Апликација се састоји од три основна слоја:

1. Клијентска страна (*frontend*) – *React* апликација која се извршава у прегледачу корисника. Одговорна је за приказ корисничког интерфејса, управљање локалним стањем и комуникацију са серверским *API*-јем.
2. Серверска страна (*backend*) – *ASP.NET Core Web API* који прима *HTTP* захтеве, обрађује пословну логику и комуницира са базом података и спољашњим сервисима.
3. База података – *SQL Server Express* који трајно чува све податке система: кориснике, квизове, питања, одговоре, резултате и логове интеракција са *AI* системом.

4. ИНТЕГРАЦИЈА ВЕШТАЧКЕ ИНТЕЛИГЕНЦИЈЕ

Централна тема овог рада је интеграција вештачке интелигенције у систем за управљање квизовима, са циљем оптимизације квалитета питања и унапређења искуства учења.

4.1. Архитектура слоја вештачке интелигенције

Сва комуникација са *OpenAI API* сервисом централизована је унутар сервиса *AIQuizService*. Сваки позив ка *OpenAI API* сервису обликује се као *chat completion* захтев који садржи поруке:

- Системска порука – дефинише улогу модела и глобална ограничења
- Корисничка порука – садржи конкретан задатак и улазне податке специфичне за сваки позив.

Модел коришћен у свим позивима је *GPT-4.1*, који је у тренутку развоја представљао најповољнији а довољно напредан модел у оквиру *OpenAI* понуде да извршава задате функционалности на српском језику, са напредним способностима разумевања контекста и генерисање структурисаног текста.

4.2. Генерисање квиза помоћу вештачке интелигенције

Прва и најзначајнија функционалност вештачке интелигенције унутар апликације је аутоматско генерисање комплетног квиза на основу: теме квиза, категорије, нивоа тежине и броја питања.

На основу задатих параметара, *GPT-4.1* генерише квиз у *JSON* формату који садржи наслов, опис, временско ограничење и листу питања са свим одговорима и тачним одговорима. Тако генерисани квиз приказује се администратору као преглед пре чувања.

4.3. Провера квалитета квиза

Функционалност провере квалитета квиза омогућава администратору да пре објављивања квиза затражи анализу садржаја квиза. Сервис шаље текст свих питања и одговора моделу који их анализира и враћа листу уочених проблема.

Сваки пронађени проблем се враћа са описом, озбиљношћу и бројем питања на које се односи. Овакав механизам доприноси образовној вредности квизова јер идентификује садржајне грешке које би могле да доведу кориснике у заблуду.

4.4. Систем наговештаја током решавања квиза

Систем наговештаја пружа кориснику могућност да током решавања квиза затражи *AI* помоћ за конкретно питање на ком се налази. Серверска страна шаље моделу текст питања, тип питања и листу понуђених одговора (без означавања тачног), а модел је инструисан да пружи наговештај који усмерава размишљање без директног откривања одговора.

Наговештај је доступан само једном по питању односно након што је приказан, дугме се деактивира. Ово ограничење обезбеђује да наговештај не буде тривијална пречица, већ алат који корисника подстиче на даље размишљање

4.5. Објашњење тачних одговора

Након завршетка квиза, корисник може затражити објашњење за свако питање у квизу. Ова функционалност осмишљена је као едукативни алат који кориснику помаже да разуме зашто је одређени одговор тачан.

Сервис шаље моделу листу свих питања квиза заједно са тачним одговорима и одговорима корисника. Модел генерише засебно објашњење за свако питање у низу, на српском језику.

5. ЕВАЛУАЦИЈА ПЕРФОРМАНСИ ГЕНЕРИСАЊА КВИЗОВА ПОМОЋУ ВЕШТАЧКЕ ИНТЕЛИГЕНЦИЈЕ

У овом поглављу приказано је мерење времена потребног за генерисање квизова различитих величина

уз поређење са проценом времена које би исти посао захтевао при ручном креирању.

5.1. Методологија мерења

Мерење је спроведено кроз серију позива ка *AI* генератору квизова у оквиру апликације, при којима су бирани различити бројеви питања (5, 10, 15 и 20). Време је мерено на серверској страни помоћу класе *StopWatch* из *.NET* стандардне библиотеке, од тренутка слања захтева ка *OpenAI API* до тренутка када сервер прими и парсира комплетан одговор.

За процену времена ручног креирања, узете су следеће претпоставке засноване на радном процесу:

- Формулисање једног питања са понуђеним одговорима: просечно 3 до 5 минута (укључује осмишљавање питања, дефинисање одговора и проверу тачног одговора)
- Подешавање метаподатака квиза (наслов, опис, тежина, временско ограничење) што је око 2 минута

Табела 1. Поређење времена *AI* генерисања и ручног креирања квизова

Број питања	<i>AI</i> генерисање (s)	Ручно креирање (s)	Убрзање (x)
5	8,94	1320	147,7
10	9,33	2520	270,3
15	12,34	3720	301,4
20	20,45	4920	240,6

5.2. Анализа резултата

Приказани резултати из табеле 1 указују на драстичну разлику у времену потребном за креирање квизова између приступа потпомогнутог вештачком интелигенцијом и ручног приступа. За квиз од десет питања, генератор испоручује комплетан садржај за 9,33 секунди, док би исти квиз ручно захтевао процењених 2520 секунди (42 минута), што представља убрзање од 270,3 пута.

Потребно је нагласити да генерисање вештачком интелигенцијом не само да је брже, већ и генерише разноврснији садржај. Систем аутоматски комбинује различите типове питања и прилагођава формулацију одговарајућем нивоу тежине.

6. ЗАКЉУЧАК

У овом раду описани су развој и евалуација *web* апликације *Kvizio*, прецизније, система за креирање и решавање квизова који користи велики језички модел *GPT-4.1* као централни елемент четири функционална модула: аутоматско генерисање питања, наговештаји током решавања, објашњења тачних одговора и провера квалитета квиза.

Мерења спроведена у току употребе показала су да генерисање квиза од десет питања траје око десет секунди, на супрот процењених 42 минута при ручном раду. Ова разлика није само ствар брзине, већ сервис истовремено комбинује различите типове питања и прилагођава формулацију задатом нивоу тежине, што би ручно захтевало додатно време и искуство аутора.

Систем је контејнеризован помоћу *Docker* технологије и може се покренути на било којој машини без претходне конфигурације окружења, што га чини прикладним за даљи развој и тестирање.

Даље унапређење могло би обухватити адаптивни систем тежине који прати напредак индивидуалног корисника као и проширење на више језичких модела ради поређења квалитета генерисаног садржаја.

7. ЛИТЕРАТУРА

- [1] S.J. Russell, P. Norvig, "Artificial Intelligence: A Modern Approach", Hoboken, Pearson, 2021.
- [2] J. McCarthy, M.L. Minsky, N. Rochester, C.E. Shannon, "A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence", Hanover, 1955.
- [3] A.M. Turing, "Computing Machinery and Intelligence", *Mind*, Vol. 59, No. 236, pp. 433-460, 1950.
- [4] A. Krizhevsky, I. Sutskever, G.E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", *Advances in Neural Information Processing Systems*, Vol. 25, pp. 1097-1105, 2012.
- [5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, "Attention Is All You Need", *Advances in Neural Information Processing Systems*, Vol. 30, pp. 5998-6008, 2017.
- [6] OpenAI, "GPT-4 Technical Report", *arXiv preprint arXiv:2303.08774*, 2023.
- [7] <http://developers.openai.com/api/docs/guides/prompt-engineering> (pristupljeno u junu 2026.)
- [8] W. Holmes, M. Bialik, C. Fadel, "Artificial Intelligence in Education: Promises and Implications for Teaching and Learning", Boston, Center for Curriculum Redesign, 2019.
- [9] UNESCO, "Guidance for Generative AI in Education and Research", Paris, UNESCO, 2023.
- [10] M.A. Chaudhry, E. Kazim, "Artificial Intelligence in Education (AIED): A High-Level Academic and Industry Note 2021", *AI and Ethics*, Vol. 2, pp. 157-165, 2022.

Кратка биографија:



Милош Јосиповић рођен је у Ужицу 2000. године. Дипломирао је на Факултету техничких наука 2023. године, на смеру Примењено софтверско инжењерство.

Контакт:

milos.josipovic2000@gmail.com