



# Question Design Using NLP

Maty Sene Kane<sup>1(✉)</sup>, Alassane Diop<sup>1</sup>, Zinflou Arnaud<sup>2</sup>,  
and El Hadji Mamadou Nguer<sup>3</sup>

<sup>1</sup> Université Numérique Cheikh Hamidou Kane (formerly UVS), Thiès, Senegal  
`maty1.sene@unchk.edu.sn, alassane.diop@unchk.edu.sn`

<sup>2</sup> Hydroquebec-CANADA, Montreal, Canada  
`zinflou.arnaud@hydroquebec.com`

<sup>3</sup> Université Numérique Cheikh Hamidou Kane (formerly UVS),  
Dakar, Diamniadio, Senegal  
`elhadjimamadou.nguer@unchk.edu.sn`

**Abstract.** Cheikh Hamidou KANE Digital University (ex UVS) has more than 60,000 students enrolled in 46 tracks containing more than 1,000 courses, the assessment of which requires the preparation of nearly 2,000 exam topics per year. Since its opening in 2013, it is estimated that nearly 11,000 topics have been prepared, and this number is growing (see [4]). In addition to these numbers, on the one hand we have the lack of an examination management system (EMS). On the other hand, we see enormous difficulties in receiving assessment topics at the teacher level, given the large number of topics a teacher has to design. We propose a framework called JEMS (Jolof Examination Management System) that allows for declarative expression and evaluation of topics close to their design. In order to be able to generate questions automatically, we have made use of other models such as natural language processing based models. This framework aims to automate evaluation questions from an existing question bank for a CE (Constituent Element). Automating evaluations in this context poses a number of scientific challenges that constitute the contributions of the presented work:

- The implementation of a file template in csv format.
- Question models that identify entities in CEs and generate statements against the natural language processing model

**Keywords:** NLP · generate questions automatically · examination management system

## 1 Introduction

Computer-based exams have grown in popularity in recent years as they support e-learning and make it easier to manage (i.e. design, set up, schedule and score, publish results) of exams. However, adopting an appropriate system that meets the needs of a certain institution can be difficult due to several concerns related to integration with existing systems, security, customization, ease of use and cost.

Before this design idea using NLPs, we had set up a prototype whose goal was to bank exam subjects. This project is based on a database model and is UN-CHK, Senegal developed with the php language using the MVC model. This MVC model does not allow us to enrich the bank of questions in an automatic way, so it will be necessary to use other models of information retrieval and/or machine learning, more precisely the NLP models. The paper is organized as follow: We present our subject bank dataset in the next section. Then we explain the NLP approach we used to automatic question generation in section three. We describe the two algorithms we used to generate questions automatically in the fourth section and we finish by a general conclusion in the last section.

## 2 Bibliographic Review

The comprehensive design of computerized testing systems has been explored by many researchers in different institutes around the world such as those in [9–20]. For example in [9–16], the effective reform of college examination management is discussed Based on the concept of scientific development of examination management. In [16], the analysis, design, development, integration, deployment and security of a web-based exam management system which was developed in-house at the German Jordanian University (UGJ) was addressed. An algorithm for the exam scheduling system is recalled in [17]. An algorithm for the exam scheduling system is recalled in [17]. In relation to malpractices, [18] then identified modifications that could help relieve the student from exam-related stress and thus increase the student’s effort towards effective learning and discourage malpractices in the long run. term. The problem of university examination timetable was explained in [19] using topics from Universiti Malaysia Terengganu (UMT). Still for the simplification of exams, [20] proposes the “Serious games” as tools to carry out evaluations by reducing the exam anxiety encountered by learners.

However, we have plenty of EMS that exist today and which offer important solutions for e-exams with competitive prices. We will detail them on the following lines:

- testwe  
testwe is a complete e-Exam solution: online platform for teaching staff and administration, offline software for learners In their [white paper](#) entitled “THE 5 IMPACTS OF THE E-EXAM” they describe how e-exams can help institutions compared to paper-based exams.
- Managexam By bringing together teaching and student administration around the same digital environment, Managexam facilitates the organization and management of evaluations while guaranteeing a high standard of educational integrity at each stage. Thanks to its simple and flexible tool Managaexam allows the design of:
  - Classic subjects
  - Evaluation activity enriched with video, audio recordings or large documents with customization options such as random drawing of questions or even variability of data between candidates

For more details, [go to](#)

- testinville It is an e-examination platform that offers the following solutions:
  - the creation of exams,
  - the design of examination processes,
  - the presentation of online exams,
  - report generation
  - analysis of examination results,
  - the pricing structure.

In their [documentation](#), it specifies that there is no payment but certain sections such as sending exam invitations to candidates by the Test Invite electronic messaging system, starting a candidate’s exam, Video or photo surveillance for exam security are charged. Head to [testinville](#) for more information.

We have many systems that offer interesting solutions for exams, But we propose automatic methods for populating a question bank using NLP.

A	B	C	
No.	QUESTIONS	REPONSE	DISTRACTEURS(OPTIONNEL)
1	Quel attribut HTML est utilisé pour ajouter un style dans un élément?	Style	
2	Quel attribut HTML est utilisé pour ajouter un lien dans un élément?	href	
3	Quel attribut HTML est utilisé pour ajouter une couleur de fond dans un élément?	background	
4	C'est quoi un élément sémantique ?	Un élément qui a une signification.	
5	Lesquelles des balises n'appartiennent pas à HTML 5?	<pre>&lt;center&gt; &lt;font&gt; &lt;strike&gt; &lt;sp&gt; &lt;u&gt;</pre>	

**Fig. 1.** bank of question

### 3 Matériel et Méthodes

In this study, we have taken the case of the UN-CHK, more precisely the multimedia stream with the Web Technology1-HTML/CSS module. We use two methods to populate the question bank:

- Use the negation of a question.
- Duplicate a question by modifying a keyword.

#### 3.1 Presentation of the Dataset Web Technology-Initiation in HTML/CSS

This dataset has three parts: (Fig. 1)

- The question column proposed by teachers or tutors
- The answer column of each question
- The distractors column which is optional. The user can give it but we can propose an algorithm that calculates it from the proposed question.

We propose two techniques for question generation:

1. Use the negation of a question.
2. Duplicate a question by changing a keyword

## 3.2 NLP Approaches Based on Word-Embedding

As in [4], NLP, Natural Language Processing, is a discipline that focuses on the understanding, manipulation and generation of natural language by machines. Thus, NLP is really at the interface between computer science and linguistics. It is about the ability of the machine to interact directly with humans.

### 3.2.1 Description of Our Two Algorithms Stated Above

Preprocessing In NLP, the first step is often preprocessing, which includes the tokenization and text cleaning steps. We are content here with a minimalist preprocessing: removal of punctuation and stop words (for visualization and vectorization methods based on counts). We propose to use the spaCy library which allows better automation in the form of preprocessing pipelines. spaCy is a free open source library for advanced natural language processing (NLP) in Python. If we work with a lot of text, we might want to know more about this. For example, what is it? What do the words mean in their context? Who does what to whom? What companies and products are mentioned? Which texts are similar? spaCy is designed specifically for production use and helps we create applications that process and “understand” large volumes of text. It can be used to create systems for information extraction or natural language understanding, or to preprocess text for deep learning. For more information on spacy see [5].

### 3.2.2 Question Generation Using Negation

Can we reproduce other questions from this topic? It is possible to use the negation or remove the negation.

Example: the question (The imain; tag specifies the main content of a document. True or False and the correct answer is True) One could be modified as follows: The main tag does not specify the main content of a document. True or False and the correct answer is False Likewise for the question (Which of the tags does not belong to HTML 5? Here all the answers are good) In this case the modification is as follows: Which of the tags belong to HTML 5 ? and the answer is no right answer.

To make a question negative, we propose two steps: **Verb detection:** To write this algorithm, we first installed the spacy library in order to be able to use the various useful functions to find the verb in the sentence and import the ‘fr\_core\_news\_sm’ language (Fig. 2).

Once the verb is found, we will define an “affirmative\_en\_negative” function or we use regular expressions (re) with the sub function.

Regular expressions (denoted RE or regex patterns in this document) are essentially a small, highly specialized programming language embedded in Python and whose manipulation is made possible by the use of the re module. Using this small language, you define rules to specify a match against a desired set of strings; these strings can be sentences, email addresses, TeX commands or whatever. You can also use RE to modify a string or split it in different ways. Like our case in the “affirmative\_en\_negative” function, we enclose the verb in



Fig. 2. Verb detection

“ne ... pas” (in french) with the sub function. The sub() function searches for all substrings where the RE matches and substitutes them with a different string (Fig. 3).

### 3.3 Question Generation by Duplicating the Question

The execution of this code gives us the following results:

For this part we used regular expressions with the sub function which is explained above.

1. The user has to give a keyword to replace.
2. Then we build the dictionary of words that we will use
3. From this dictionary, we build a list to have a set of questions.

For more details on regular expressions in Python, please consult the documentation [7]. Try to add more comments about the methods and the performance of the models (Fig. 4).

This algorithm produces the following results (Fig. 5).

```
[ ] 1 import re
2 def affirmative_en_negatives(x):
3     verbe=trouve_le_verbe(x)
4     print(verbe)
5     phraseNeg=re.sub(verbe, ' ne '+verbe+' pas ',x)
6     return phraseNeg
7
```

```
[ ] 1 phrase=input('Donner une phrase: ')

```

Donner une phrase: La balise main spécifie le contenu principal d'un document.

```
[ ] 1 trouve_le_verbe(phrase)

```

'spécifie'

```
[ ] 1 affirmative_en_negatives(phrase)

```

spécifie

'La balise main ne spécifie pas le contenu principal d'un document.'

Fig. 3. Affirmative to negative

```
1 import re
2 qu=input('Une question')
3 listMots=[]
4 while nbq in(range(3)):
5     nbq=int(input('Combien de question voulez vous reproduire?'))
6
7 for j in range(nbq):
8     m=input('saisir un mot')
9     listMots.append(m)
10 print(listMots)
11 mot_a_replacer=input('Donnez le mot cles a remplacer')
12 listqu=[qu]
13 i=0
14 for mot in listMots:
15     quNouveau=re.sub(mot_a_replacer,mot,qu)
16     listqu.append(quNouveau)
17     i=i+1
18 listqu
```

Fig. 4. Duplicate question

Une questionQuel attribut HTML est utilisé pour ajouter un style dans un élément?

saisir un motlien

saisir un motjeu de caractere

saisir un motcouleur de font

['lien', 'jeu de caractere', 'couleur de font']

Donnez le mot cles a remplacerstyle

['Quel attribut HTML est utilisé pour ajouter un style dans un élément?',

'Quel attribut HTML est utilisé pour ajouter un lien dans un élément?',

'Quel attribut HTML est utilisé pour ajouter un jeu de caractere dans un élément?',

'Quel attribut HTML est utilisé pour ajouter un couleur de font dans un élément?']

Fig. 5. results of the duplication

## 4 Results

We have a bank of questions that we can enrich over time with our two methods. These results are presented in a CSV file, and we plan to set up a platform to automate the populating of this dataset.

## 5 Conclusion

In this paper, we have recalled the context and the goal of automating evaluations. Subsequently, we used NLP to give a proposal for automatic question generation by proposing two features (negation, and duplication).

On related works such as [1, 2] and [3], we have multiple choice generations question that are based on wikipedia topics and courses respectively. In all cases, it is a generation of questions on knowledge bases. To improve these rich scientific works, we propose a reformulation of the questions already existing in a question bank by presenting two functions. One modifies the question by taking its negation, the other duplicates the question by modifying a main keyword.

In perspective, we are looking:

- The quality of the questions produced by our two functions.
- The generation of distractors from the questions
- The implementation of the application in a general way.

discuss other ways of generating questions such as information retrieval.

## References

1. Graesser, A.C., Wisher, R.A.: Question generation as a learning multiplier in distributed learning environments. US Army Research Institute for the Behavioral et Social Sciences, Alexandria VA (cf. p. 1, 21, 23, 85, 122) (2001)
2. Raynaud, T.: Génération de questions á choix multiples thématiques á partir de bases de connaissances. <https://tel.archives-ouvertes.fr/tel-02901501/document>
3. Nwafor, C.A., Onyenwe, I.E.: An automated multiplechoice question generation using natural language processing techniques. <https://airconline.com/ijnlc/V10N2/10221ijnlc01.pdf>
4. uvs(DISI) Université Virtuelle du Sénégal[en ligne] (modifié en mars 2023). <https://www.uvs.sn/luniversite-2/luniversite-en-chiffres/>. Accessed 21 June 2022
5. DATASCIENTEST [en ligne] (Created on 22/07/2020). <https://datascientest.com/introduction-au-nlp-natural-language-processing>. Accessed 11 Feb 2022
6. spacy[en ligne] (created since 2016). <https://spacy.io/usage/spacy101#whats-spacy>. Accessed 21 Oct 2022
7. Kuchling, A.M.: (Documentation Python 3.11.2) (modified on March 30, 2023). <https://docs.python.org/fr/3/howto/regex.html>. Accessed 30 Oct 2022
8. waset. Conférence internationale sur le système de gestion des examens Riga, Lettonie du 18 au 19 juin 2020. <https://jemsenegal.wordpress.com/2021/04/06/jems-un-modele-de-systeme-demanagement-des-examens-pour-les-universites-du-senegal/>

9. Liu, N., Dong, X.: Research on the reform and security of college examination. *Journal of Shenyang Architectural University* (2008). [Citation Time(s):1]
10. Wang, L., Liu, H.-Y.: Analysis on the Restrict of College Examination Reform. *Higher Engineering Education Research* (2006)
11. Han, S.-J.: The exploration of evaluation mode in the high professional university. *Educ. Prof.* **29**, 166–167 (2011)
12. Li, X.-Q., Shu, H.: The Exploration and Practice of Examination Model Reformed by Stratified Teaching in Higher Vocational (2011)
13. Mathematics Curriculum. *Journal of Chifeng University*, vol. 9, pp. 252–253
14. Niu, W.-X.: Quality education calls for the examination reform. *Educ. Naval Acad.* **3**, 73–74 (2000)
15. Zhao, J.-Z.: The research on the present situation of the examination reform in higher vocational colleges. *Chin. Exam.* **7**, 40–44 (2010). [Citation Time(s):1]
16. Al-hawari, F., Althawbih, H., Alshawabkeh, M., Abu Nawas, O.: Integrated and secure web based examination management system. *Computer Applications in Engineering Education* (2019)
17. Khan, T.: Examination Scheduling System with Proposed Algorithm, s. d., 1
18. Omole, C.: Modifying Assessment Modes in the Science Classroom as a Solution to Examination Malpractice, s. d., 1
19. Jamaluddin, N.F., Aizam, N.A.H.: Timetabling Communities' Demands for an Effective Examination Timetabling Using Integer Linear Programming, s. d., 1
20. Ajith, R., Bijlani, K.: Serious Game as a Performance Assessment Tool That Reduces Examination Anxiety, s. d., 1