IJESC

Project Guidance Portal with Plagiarism Detection

Shylajhaa .R .S[1], Swarna Swapna .I[2], Swathy .L[3], Dr. M. Sheerin Banu[4]
BE Student[1, 2, 3,] Professor[4]
Department of Computer Science and Engineering
RMD Engineering College, Tamil Nadu, India

Abstract:
The main aim of this project is to develop a single application that suffices all the requirements of a student, trying to build his/ her own product. The website aims to give A-Z guidance on all aspects of developing a feasible and successful product. The website is integrated with a Natural Language Processing based plagiarism checker that assesses the level of plagiarism in any idea proposed by the user.

1. INTRODUCTION:

1.1 Overview:

Project development and hands-on experience in technologies has become one of the major skills to be possessed among students from college. Knowledge and grades acquired through academics are usually not of great concern, as students enter their respective industries. So, it is vital that students equip themselves with necessary skills and hands-on experiences. Projects that students work on during their course of study is also a major concern. Often many students fail to accomplish this due to poor or nil guidance from experts. In worst cases, students are clueless of how to proceed with their innovative ideas. Our project aims to bridge the gap between students and industry, by providing A-Z guidance on all aspects of building a successful and innovative product. Though there are many useful resources available over the internet, there is no guidance tool that suggests the right technologies to be used to develop a solution. Also, majority of the time spent in building a software is wasted on developing modules that already exist. It requires high decision-making skills to integrate the right open source libraries and apis into one's product. And often brilliant and innovative ideas are rejected due to plagiarism. This often happens due to the lack of awareness among students about existing papers and projects. This website has its own Natural Language Processing based Plagiarism Detector that assess the ideas for their level of innovation and shows the existing works that are similar. This gives a wider vision to one's idea and more space for creativity. The plagiarism detector checks for both text and image plagiarism and canbe used by institutions and technical societies to validate incoming papers. All these facilities are also available through a chatbot integrated with the website. Now, users can get guidance and answers for their queries through a NLP driven Chatbot. The website also provides guidance on professional report and abstract writing.

1.2 Motivation:

The internet has millions of resources on developing products. But it is difficult to make the right, feasible choice among millions of available choices. Also, many papers that clear the plagiarism test are rewritten or slightly modified, and are not innovative. An artificial intelligence based chatbot, data mining techniques and Natural Language Processing based plagiarism detector would solve all the above-mentioned issues

1.3 Existing work:

Existing plagiarism checkers check for text plagiarism based on exact similarity of sentences and words. It often does not detect plagiarism in texts that are slightly modified or re-written. Semantic and image similarity is an issue in these systems. And there is no existing system that guides students in developing their own innovative products.

2. PROPOSED SYSTEM:

Our proposed system is a project guidance portal driven by Machine Learning, Natural Language Processing and Data Mining approaches. Guidance is provided by the website through a trained chatbot that suggests the right technology stack to be adopted to implement any proposed idea. A text and image plagiarism checker is also designed to determine the level of innovation in any idea that it proposed. Plagiarism is checked against abstracts crawled and fetched from the World Wide Web. The result of the plagiarism checker is based on the semantic meaning of the abstract/ paper rather than exact words or sentences. The keywords from the input document are extracted and relevant results are retrieved from the internet.

2.1 Plagiarism Checker:

The user uploads his/her input abstract/paper in any of the formats (.doc/. pdf/.txt). The keywords from the uploaded document are extracted based on POS-tagging followed by an algorithm. Part of Speech (POS) tagging converts the input document into a dictionary of words and their part of speech (noun, conjunction, etc.). Based on these tags words with higher weights are determined and termed as keywords. Keywords that are phrases and not single words are also detected using this technique. These keywords are searched in the World Wide Web and relevant results are retrieved. The results are parsed to retrieve the content and checked for similarity with the input document. Similarity is checked based on meaning of the document and can therefore detect plagiarism in modified abstracts too. A similar process is involved in detecting similarity in images. Image plagiarism detection can help determine stolen UML diagrams, etc.
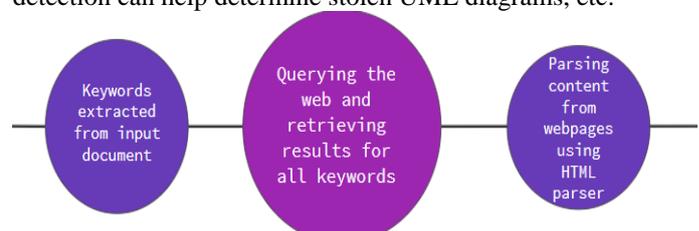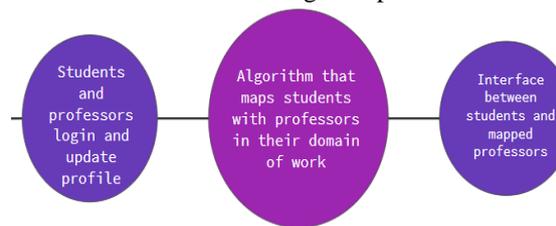
2.2 Guidance Chatbot:

A Machine learning based chatbot that would help students to learn basics of the application development and to resolve the issues faced by them during the phase of developing an application. The chatbot is trained to recommend best frameworks for each development platform. It would also provide learning materials, tutorials, examples and reference documentation of each application framework. This would also try to incorporate the idea of professional product development in students by giving exposure to version control systems, Official Documentation and much more. If the chatbot could not resolve an issue, it would map the issue to a person with required technical knowledge to resolve that issue.
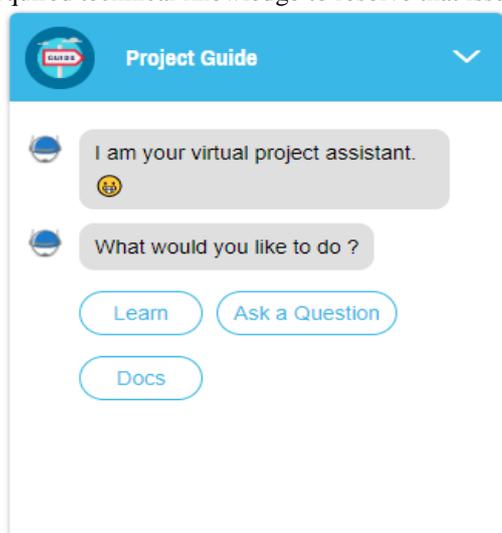
2.3 Forum:

A discussion forum to interact with fellow developers and professionals is set up to post questions and answer the queries of other users. Users can also post their idea abstracts and papers, that will benefit other users. The website also maps professionals to students based on their technology requirement. Students can directly interact with industry experts and acquire knowledge on their domain of work.

2.4 Templates and Tutorials on Report Writing:

This module provides templates for various forms of report writing; e.g. Abstracts, Journals, Project thesis/ report, etc. It provides standardly used templates and animated tutorials on writing professional templates. Users need not spend time and energy in making ready professional templates. By filling up simple forms the tool generates the required type of report.

2.5 Student-Professor Mapping (For Institutions)

It is always a herculean task for colleges in mapping the correct set of professors to students who require guidance in a specific domain. This module provides a login to students and professors of registered colleges and maps students to professors with knowledge and expertise in that domain. For example, a student working on a data science project will be mapped with a professor who is an expert in data science. It is required that students and professors maintain an updated profile in order to make best use of this feature. Students can then interact and clarify doubts through the website. They can also share documents through the portal.

3. ADVANTAGES:

- Integration of all utilitiesinto one website.
- Easily understandable guidance through chat bot
- Readily generated reports and
- Semantic meaning-based plagiarism checker

4. CHALLENGES:

- Huge dataset is to be maintained and updated.
- Training chatbot to answer all types of queries placed by users is also challenging.

5. CONCLUSION:

This application will be of great use and support to the developer community and help bright minds turn their ideas into deployable products. It would also help institutions maintain an easy to use platform to facilitate interaction between their faculty and students. Through meaning-based plagiarism detection, this application insists and recognizes innovation in the developer community.

6. REFERENCES:

[1]. A Web-Enabled Plagiarism Detection Tool by Colin J. Neill and Ganesh Shanmuganthan

[2]. Choosing an NLP Library for Analyzing Software Documentation: A Systematic Literature Review and a Series of Experiments by Fouad Nasser A Al Omran, Christoph Treude

[3]. Detecting plagiarism in images by Prajakta Ovhal