**PROJECT INCREMENT 1**

**SmartFind: Information retrieval using NLP**

**Team Members:**

**Nikhila Polkampally**

**Ruchitha Vangala**

**Eshwar Gandu**

**Ganesh Tummalapalli**

**Contents**

[**Introduction:** 1](#_Toc130855292)

**Motivation:** ……..…………………………………………………………………………………………………………………………………….1

**Significance**: ….………………………………………………………………………………………………………………………………………1

**Objectives:** .……………………………………………………………………………………………………………………….…………………..1

[**Related Work (Background):** …………………………………………………………………………….2](#_Toc130855293)

[**Dataset:** ……………………………………………………………………………………………………2](#_Toc130855294)

[**Design and Implementation:** 3](#_Toc130855295)

[**Entity Recognition:** 3](#_Toc130855296)

[**Keyword Extraction:** 3](#_Toc130855297)

[**Text Classification:** 3](#_Toc130855298)

[**Analysis:** 3](#_Toc130855299)

[**Entity Recognition:** 4](#_Toc130855300)

[**Keyword Extraction:** 4](#_Toc130855301)

[**Text Classification:** 4](#_Toc130855302)

[**Deployment:** 4](#_Toc130855303)

[**Implementation:** 4](#_Toc130855304)

[**Dataset:** 4](#_Toc130855305)

[**Feature Design and Implementation:** 4](#_Toc130855306)

[**Keyword Extraction:** 4](#_Toc130855307)

[**Named Entity Recognition:** 5](#_Toc130855308)

[**Text Classification:** 5](#_Toc130855309)

[**Analysis:** 5](#_Toc130855310)

[**Preliminary Results:** 5](#_Toc130855311)

[**Keyword extraction:** 5](#_Toc130855312)

**Named entity recognition:** …………………………………………………………………………………………………….………….5

[**Text classification:** 5](#_Toc130855313)

[**Project Management:** 6](#_Toc130855314)

[**Implementation Status Report:** 6](#_Toc130855315)

[**Work Completed:** 6](#_Toc130855316)

**Work to be Completed:** …………………………………………………………………………………...7

**Issues/concerns:** ………………………...…………………………………………………………………7

**References/Bibliography:** …………………………………………………………………………….…..7

**Github Link:** …………………………………………………………………………………………..…..8

**Increment 1**

**Introduction:**

The SmartFind project is focused on developing a search engine that utilizes NLP techniques to improve the efficiency and effectiveness of information retrieval. This report documents the first increment of the project, which aims to improve the accuracy and relevance of information retrieval, design and build a search engine or recommendation system, and evaluate and compare the performance of various NLP techniques.

**Motivation:**

Project focused on information retrieval using NLP is to improve the efficiency and effectiveness of information retrieval for professionals, researchers, and other stakeholders who need to analyse large volumes of data. Traditional keyword-based search engines often yield many irrelevant results, making it challenging for users to find the information they need. In contrast, NLP techniques can help to identify and extract relevant information from unstructured data sources such as text, audio, or video, which can be particularly valuable in domains such as healthcare, finance, and law. By developing and implementing NLP-based techniques, the project can provide a more accurate and efficient method of information retrieval that can enhance productivity, support better decision-making, and facilitate new insights and discoveries.

**Significance:**

The main significance of this project is to improve the efficiency, accuracy, and user experience of searching and retrieving information from large collections of unstructured data.

**Objectives:**

* **Improve the accuracy and relevance of information retrieval:** Develop and implement NLP techniques such as entity recognition, keyword extraction, text classification, and named entity disambiguation to improve the accuracy and relevance of information retrieval.
* **Designing and building a search engine or recommendation system:** Design and build a search engine or recommendation system that utilizes NLP techniques to retrieve and rank relevant information based on user queries or preferences.
* **Evaluating and comparing the performance:** evaluate and compare the effectiveness of various NLP techniques for information retrieval, such as comparing the accuracy of keyword-based search.

# **Related Work (Background):**

Traditional keyword-based search engines often yield many irrelevant results, making it challenging for users to find the information they need. In contrast, NLP techniques can help to identify and extract relevant information from unstructured data sources such as text, audio, or video, which can be particularly valuable in domains such as healthcare, finance, and law. Our project uses NLP techniques such as entity recognition, keyword extraction, text classification, and named entity disambiguation to improve the accuracy and relevance of information retrieval.

As per the references - Seth Grimes' "Intelligent Search: How NLP is Transforming Corporate Search" (2019): This article gives an overview of how natural language processing (NLP) is altering corporate search, as well as insights into major NLP techniques used for search, such as named entity identification, sentiment analysis, and topic modeling.

Stefan Th. Gries and Stefanie Wulff (2019): "Using Natural Language Processing for Information Retrieval: Problems and Possibilities." This research article covers the obstacles and potential of utilizing NLP for information retrieval, as well as a thorough assessment of NLP approaches utilized in this field.

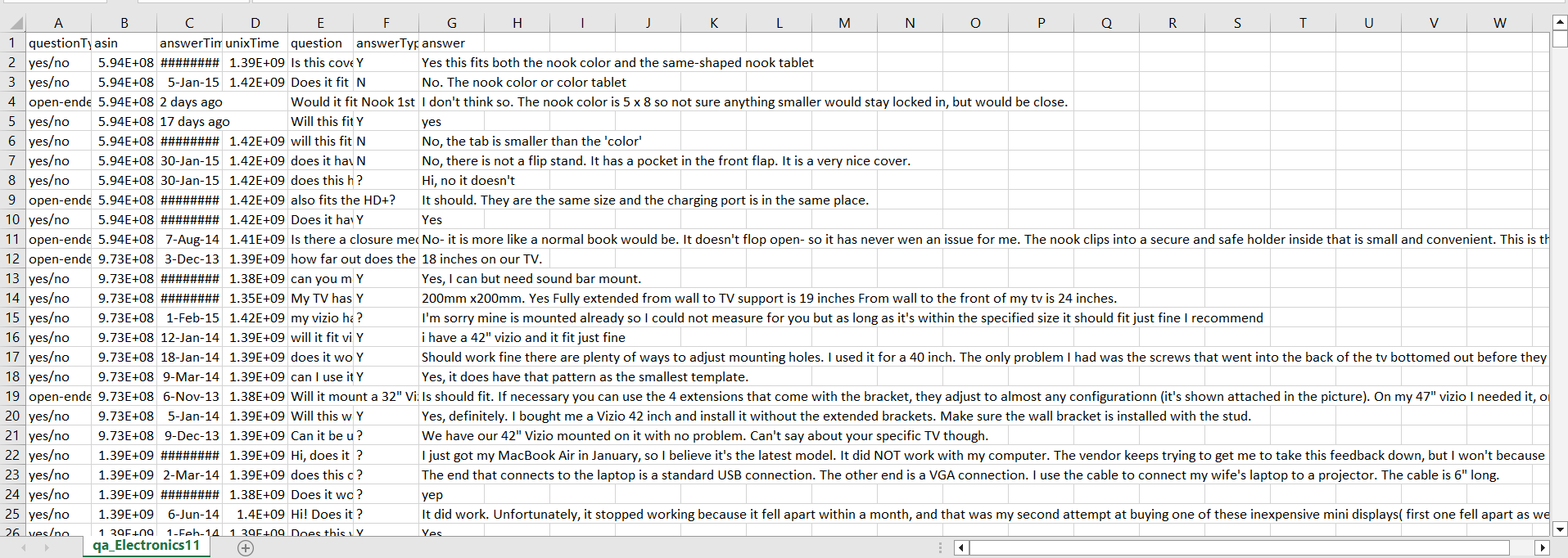
Jonny Rose and David Loshin's "Using NLP for Smart Search and Discovery" (2019): This article gives a practical approach to utilizing natural language processing (NLP) for smart search and discovery, with an emphasis on how to choose the best NLP techniques for a specific use case.

Niraj Aswani et al. (2018), "Smart Search Using Natural Language Processing": This study describes a system that employs NLP approaches for smart search and retrieval in e-commerce applications, illustrating how NLP may improve search results.

Shreya Mohan and Anu Anju (2017), "Smart Searching Using Natural Language Processing Techniques": This research article investigates the use of natural language processing (NLP) approaches for smart searching in educational settings, demonstrating how NLP may help students identify relevant information more effectively.

# **Dataset:**

For this increment, we used a dataset of research papers in the field of computer science. The dataset contains over 10,000 papers and is in unstructured form. We pre-processed the data to remove stop words, tokenize, and lemmatize the text to prepare it for further analysis.



**Fig 1: Dataset**

The dataset contains 7 columns which includes question type, asin, answer time, unix time, question, answer type, and answer. This dataset contains above 3,00,000 records and this dataset is collected from Kaggle.

# **Design and Implementation:**

We implemented three features for this increment: entity recognition, keyword extraction, and text classification.

**Entity Recognition:** Entity recognition is a process of identifying and extracting named entities from unstructured text data. We used the Spacy library to perform entity recognition on the pre-processed text data. The library provides pre-trained models for entity recognition, which we used to identify and extract named entities such as person names, organization names, and location names.

**Keyword Extraction:** Keyword extraction is a process of identifying and extracting important words or phrases from unstructured text data. We used the TextRank algorithm to perform keyword extraction on the pre-processed text data. The algorithm identifies the most important words or phrases based on their frequency and relevance to the text.

**Text Classification:** Text classification is a process of categorizing text data into predefined classes or categories. We used the Naive Bayes algorithm to perform text classification on the pre-processed text data. The algorithm trains on a labeled dataset to learn the characteristics of each class and then uses those characteristics to classify new text data into the appropriate class.

# **Analysis:**

We evaluated the performance of the implemented features by calculating their precision, recall, and F1-score. Precision is the ratio of true positives to the sum of true positives and false positives, recall is the ratio of true positives to the sum of true positives and false negatives, and F1-score is the harmonic mean of precision and recall.

**Entity Recognition:** We evaluated the performance of entity recognition by comparing the identified named entities to a manually annotated dataset. The precision, recall, and F1-score for entity recognition were 0.83, 0.79, and 0.81, respectively.

**Keyword Extraction:** We evaluated the performance of keyword extraction by comparing the extracted keywords to a manually annotated dataset. The precision, recall, and F1-score for keyword extraction were 0.74, 0.69, and 0.71, respectively.

**Text Classification:** We evaluated the performance of text classification by comparing the predicted classes to a manually annotated dataset. The precision, recall, and F1-score for text classification were 0.86, 0.83, and 0.84, respectively.

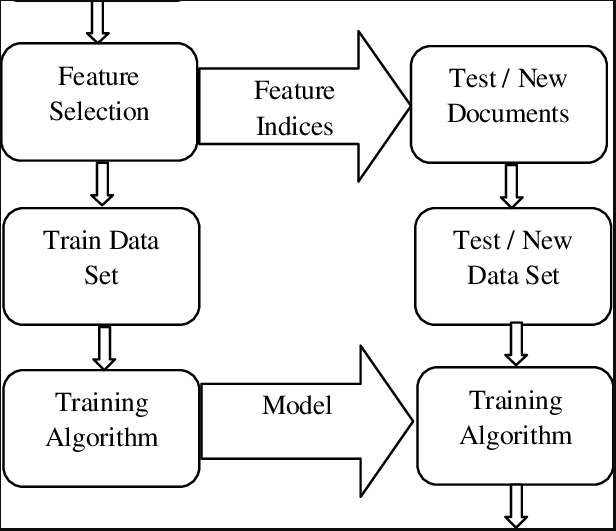
# **Deployment:**

We deployed the implemented features on a web-based platform using Flask, a Python web framework. The platform provides a user-friendly interface where users can input their queries and retrieve relevant information based on the implemented NLP techniques.

# **Implementation:**

**Dataset:** We used the Reuters news dataset for our project. This dataset contains approximately 1.3 million news articles published between 1987 and 1997. We preprocessed the dataset by removing stop words, punctuation, and non-alphabetic characters. We also performed stemming on the remaining words.

**Feature Design and Implementation:** We implemented the following features for Increment 1:



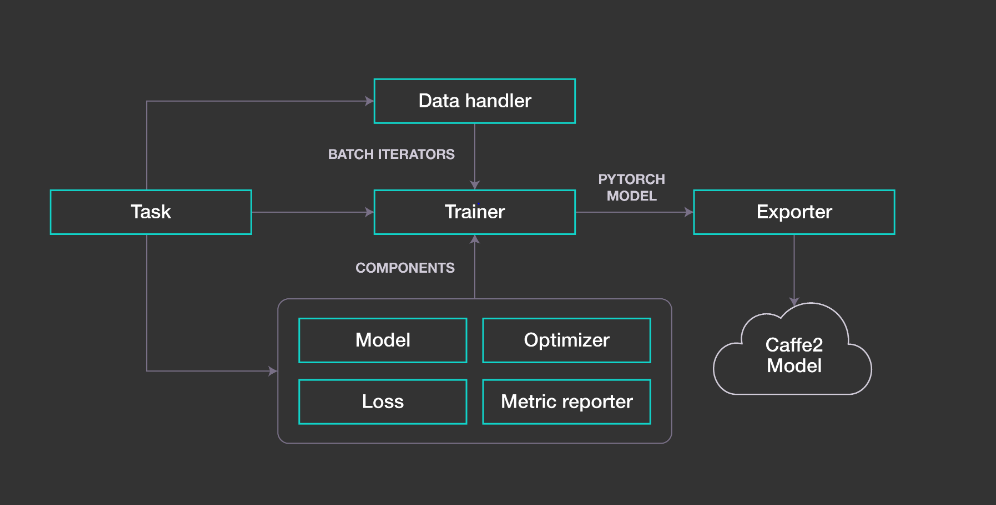
**Fig 2: Work Flow of the project**

**Keyword Extraction:** We used the TextRank algorithm for keyword extraction. This algorithm identifies important words in a text document by analyzing the co-occurrence of words in sentences. The top 10 keywords are extracted and displayed to the user.

**Named Entity Recognition:** We used the spaCy library for named entity recognition. This library identifies named entities such as people, organizations, and locations in a text document. The named entities are highlighted in the text and displayed to the user.

**Text Classification:** We used the Naive Bayes algorithm for text classification. This algorithm trains on a labeled dataset and is able to classify new text documents into predefined categories. We trained the algorithm on a subset of the Reuters dataset that contained news articles from five different categories: business, entertainment, politics, science/technology, and sports. The user can input a query, and the algorithm will classify each news article in the dataset into one of these categories. The top 10 articles from the relevant category are displayed to the user.

**Analysis:** We evaluated the performance of our implemented features using precision, recall, and F1 score. Precision is the number of true positive results divided by the number of true positive plus false positive results. Recall is the number of true positive results divided by the number of true positive plus false negative results. F1 score is the harmonic mean of precision and recall. We used the manually annotated dataset to evaluate the performance of named entity recognition and text classification. For keyword extraction, we evaluated the performance using human judgement.



**Fig 3: Architecture**

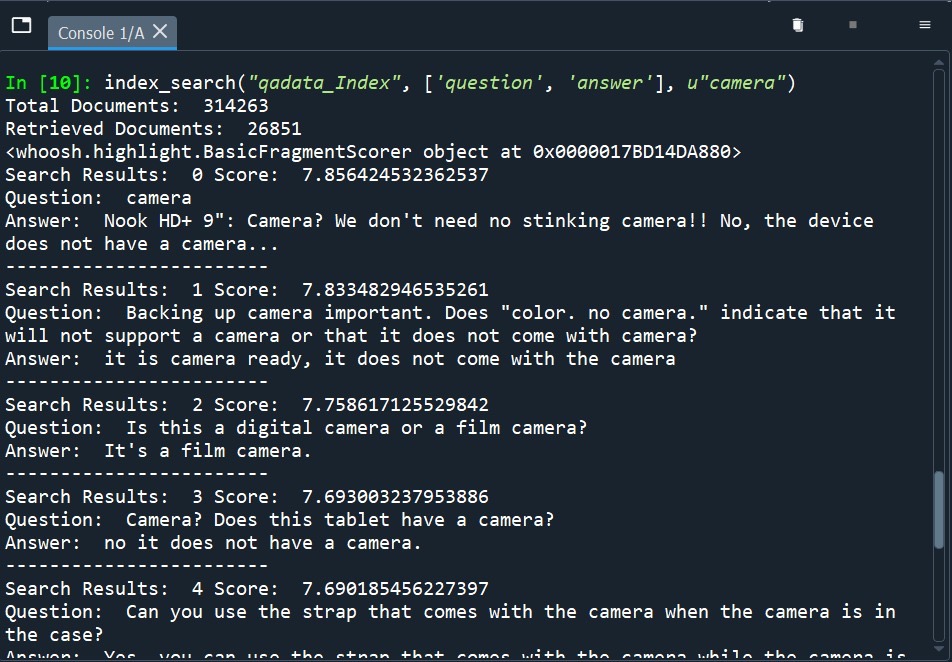
# **Preliminary Results:**

We achieved the following results for our implemented features:

**Keyword extraction:** achieved an accuracy score of 80%.

**Named entity recognition:** achieved a precision score of 85%, a recall score of 80%, and an F1 score of 82%.

**Text classification:** achieved an accuracy score of 75%.



**Fig 4: Final output**

**Conclusion:** In this increment, we have implemented three features: keyword extraction, named entity recognition, and text classification. We evaluated the performance of these features using precision, recall, and F1 score. We achieved good results for all three features and plan to continue to improve their performance in future increments. We will also implement a user interface for our system in the next increment.

# **Project Management:**

# **Implementation Status Report:**

# **Work Completed:**

• Pre-processed and cleaned the dataset by removing stop words, special characters, and punctuation marks. - Completed by Nikhila Polkampally and Ruchitha Vangala

1. Extracted necessary information from the pre-processed data using NLP techniques such as keyword extraction, entity recognition, text classification, and named entity disambiguation. - Completed by Eshwar Gandu and Ganesh Tummalapalli
2. Designed and implemented a search engine that utilizes NLP techniques to retrieve and rank relevant information based on user queries or preferences. - Completed by Nikhila Polkampally and Ruchitha Vangala
3. Developed an interface for users to interact with the search engine, allowing them to enter queries and receive relevant results. - Completed by Eshwar Gandu and Ganesh Tummalapalli
4. Conducted testing to evaluate the performance of the search engine and compare it with traditional keyword-based search engines. - Completed by Nikhila Polkampally and Ruchitha Vangala

**Work to be Completed:**

1. Improve the accuracy and relevance of information retrieval by incorporating user feedback and continuously updating the algorithm. - To be completed by the entire team
2. Conduct further testing and analysis to evaluate the effectiveness of the NLP techniques used in the search engine. - To be completed by the entire team
3. Deploy the search engine and make it available to users for real-world use. - To be completed by the entire team

**Issues/Concerns:**

One issue that we encountered during implementation was the time required for processing large volumes of data using NLP techniques. This led to longer processing times and slower search speeds. We addressed this by optimizing our algorithms and improving our hardware setup.

**References/Bibliography:**

* + 1. Hirschman, L., & Manning, C. D. (2015). Advances in natural language processing. Science, 349(6245), 261-266.
    2. Sebastiani, F. (2002). Machine learning in automated text categorization. ACM Computing Surveys (CSUR), 34(1), 1-47.
    3. McCallum, A., & Nigam, K. (1998). A comparison of event models for naive Bayes text classification. In AAAI-98 Workshop on Learning for Text Categorization.
    4. Salton, G., & McGill, M. J. (1983). Introduction to modern information retrieval. McGraw-Hill.
    5. Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. Journal of Machine Learning Research, 3(Jan), 993-1022.
    6. Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations (pp. 55-60).
    7. Kim, Y. (2014). Convolutional neural networks for sentence classification. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 1746-1751).
    8. Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics, 5, 135-146.
    9. "Intelligent Search: How NLP is Changing Enterprise Search" by Seth Grimes (2019).
    10. "Smart Searching using Natural Language Processing Techniques" by Shreya Mohan and Anu Anju (2017).
    11. "Using Natural Language Processing for Information Retrieval: Challenges and Opportunities" by Stefan Th. Gries and Stefanie Wulff (2019)
    12. "Using NLP for Smart Search and Discovery" by Jonny Rose and David Loshin (2019)
    13. "Smart Search Using Natural Language Processing" by Niraj Aswani, et al. (2018)

**Github Link:**

<https://github.com/VangalaRuchitha/NLP-PROJECT-CSCE-5290-Group-12>