**Current Problem:**

The lack of keywords is a recurrent problem in the automatic categorization of articles. The keywords are entered manually by hand which is limiting the search results and is leaving a wide scope for human errors and disparity which can be avoided by using *Machine Learning models for natural language processing (NLP) and natural language understanding (NLU).*

Therefore, methods must be found to automatically extract concepts and keywords from a text. In order to evaluate the relevance of an automatically extracted set of keywords, datasets often compare the keywords extracted by an algorithm with keywords extracted by several humans.

**Overview:**

1. **Rapid Automatic Keyword Extraction (RAKE)** is a Domain-Independent keyword extraction algorithm in **Natural Language Processing** which can be inputted by using **Python.**
2. It is an Individual document-oriented dynamic Information retrieval method.
3. Concept of RAKE is built on three matrices

* Word Degree (deg(w))
* Word Frequency (freq(w))
* Ratio of the degree to frequency (deg(w)/freq(w))

1. Another Python NLP library is **spaCy.**
2. **Textrank** is a **Python tool** that extracts keywords and summarizes text. The algorithm determines how closely words are related by looking at whether they follow one another.

### **Why Is Keyword Extraction Important?**

### With keyword extraction you can find the most important words and phrases in massive datasets. And these words and phrases can provide valuable insights into topics your customers are talking about.

*Keyword Extraction helps you automatically index data, summarize a text, or generate tag clouds with the most representative keywords.*

#### **Scalability**

Automated keyword extraction allows you to analyse as much data as you want. You could read texts and identify key terms manually, but it would be extremely time-consuming. Automating this task gives you time to concentrate on other parts of job.

#### **Consistent criteria**

Keyword extraction acts based on rules and predefined parameters. You don’t have to deal with *inconsistencies*, which are common in manual text analysis.

***USE CASES OF NLP***

One of the top **use cases** of **natural language processing** is:

1. Translation.
2. Autocorrect
3. Autocomplete
4. Conversational AI
5. **Search Engine**

*NLP and NLU make semantic search more intelligent through tasks like* ***normalization, typo tolerance, and entity recognition.***

These underlying techniques are often used in higher-level NLP capabilities, such as–

**Parsing** - Parsing is all about splitting a sentence into its components to find its meaning.

**Stemming** - Stemming is a method of reducing the usage of processing power, thus shortening the analysis time.

**Lemmatisation** - Lemmatisation differs a bit from stemming in that it reduces words into their most basic forms.

**Named Entity Recognition**- Named Entity Recognition (NER) also called entity identification or entity extraction - is the process of matching named entities with pre-defined categories.

**Stop Words Removal** – It means to get rid of words that provide you with little semantic value. It, usually, removes prepositions & conjunctions, along with words like “is,” “to”, “my,”, etc.

**Letter Normalization** - normalization would be the handling of letter case and not making it case sensitive.

**Tokenization -** The next normalization challenge is breaking down the text the searcher has typed in the search bar and the text in the document.

**Typo Tolerance and Spell Check –** The best typo tolerance should work across both query and document, edit distance generally works best for retrieving and ranking results. (the [Damerau-Levenshtein Distance algorithm](https://www.lemoda.net/text-fuzzy/damerau-levenshtein/" \t "_blank))

**Conclusion**

Among the methods tested, RAKE seems to be quite relevant to the subject of the text. In order to better compare these approaches, we would obviously have to evaluate these different models on a larger number of examples. There are different metrics to measure the relevance of these keyword extraction models with our required needs.