PatternMining: Enhancing Unsupervised Key-phrase Extraction Using Syntactic Selection

Nishanth Krishna Churchmal
Arizona State University
Tempe, Arizona
nchurchm@asu.edu

Sensen Wang
Arizona State University
Tempe, Arizona
swang326@asu.edu

Wei-Cheng Liao
Arizona State University
Tempe, Arizona
wliao18@asu.edu

Shreyas Sharma
Arizona State University
Tempe, Arizona
slsharm1@asu.edu

ABSTRACT

In the rapidly evolving landscape of machine learning, key-phrase extraction emerges as a pivotal challenge, central to various applications ranging from information retrieval to content summarization. While substantial progress has been made, existing methodologies often grapple with comprehensively capturing the nuanced interplay of syntactic structures within texts. This study introduces an innovative approach that leverages pattern mining to extract syntactic information, such as parts of speech, to enhance the efficiency of unsupervised key-phrase extraction. Utilizing the Inspec dataset, our methodology integrates advanced natural language processing techniques, including BERT, T5, and TextRank algorithms, to mine patterns that elucidate the syntactic underpinnings of effective key-phrase identification.

Through a comparative analysis with state-of-the-art methods referenced in seminal works, we assess our approach on metrics such as F1@10, unraveling the potential of pattern mining in augmenting the performance of existing key-phrase extraction frameworks. This comparative examination aims to provide meaningful insights into the impact of syntactic pattern mining on the efficacy of key-phrase extraction methodologies across the Inspec dataset. Our findings highlight the substantial improvements in precision and recall achieved by incorporating syntactic information, thus offering a new direction for future research in unsupervised key-phrase extraction. This endeavor not only underscores the importance of syntactic awareness in text analysis but also opens avenues for further exploration into the integration of linguistic patterns with machine learning techniques for enhanced text understanding.

1 INTRODUCTION

In today's digital ecosystem, every minute sees the creation of massive volumes of text, from tweets and emails to scholarly articles and e-books. This exponential growth in textual data introduces a complex challenge: how to efficiently preprocess, analyze, and derive meaning from such vast quantities of information. As we tackle this daunting task, the discipline of Natural Language Processing (NLP) presents key-phrase extraction as an essential strategy. This technique focuses on identifying and extracting significant words and phrases that succinctly summarize the core content of texts. By doing so, key-phrase extraction serves as a crucial step in converting overwhelming amounts of textual data into manageable, interpretable insights. It allows both machines and humans to quickly identify the main themes and topics within large datasets, facilitating a more efficient navigation through the ever-growing digital textual landscape. Moreover, by highlighting the most relevant information, key-phrase extraction aids in the categorization, indexing, and summarization of texts, making the retrieval of specific information faster and more accurate.

Key-phrase extraction is a crucial technique within Natural Language Processing (NLP) that identifies descriptive words or phrases, encapsulating the main themes and contents of texts. This method addresses the challenges presented by the deluge of digital textual data, facilitating tasks such as summarization, categorization, and information retrieval. As the complexity and volume of textual data expanded, traditional NLP approaches evolved to incorporate more sophisticated methodologies. These include deep learning models like attention and transformer models, graph-based algorithms, and unsupervised learning techniques. This evolution has significantly enhanced the field's ability to process and comprehend text with nuanced detail and context sensitivity.

However, despite these advancements, challenges remain. The effectiveness of these methodologies can be hampered by the intricacies of language, variations in context, and the diversity of text domains. Deep learning models, while potent, often necessitate extensive labeled datasets and considerable computational power, constraints that are not always practical. Unsupervised techniques, vital in the absence of annotated data, have yet to achieve the performance levels of supervised methods consistently. They sometimes struggle to accurately represent the semantic depth and relevance of key-phrases across various contexts. These limitations highlight the necessity for ongoing research and development in NLP to create more efficient, versatile, and resource-conscious key-phrase extraction methods.

In key-phrase extraction, the advent of transformer-based models like BERT, KeyBERT, and T5, alongside graph-based algorithms such as TextRank, TopicRank, and notably PatternRank, has significantly advanced our ability to distill complex texts into concise, informative phrases. Transformer models, leveraging deep contextual embeddings, have shown exceptional skill in capturing textual nuances, with BERT and KeyBERT utilizing bidirectional context to unveil the intricate relationships between words, and T5 adopting a generative approach for direct summarization. These methods excel in grasping the essence of content, yet face challenges such as

the need for substantial computational resources and difficulties in extracting domain-specific phrases without significant fine-tuning.

Concurrently, graph-based techniques offer a complementary perspective by mapping the relational structure of texts to highlight key concepts. TextRank and TopicRank, for instance, prioritize phrases based on their connectivity and thematic prominence, respectively, while PatternRank innovatively employs syntactic patterns for a more nuanced phrase selection. These strategies, less reliant on heavy computational demands, still grapple with the quality of input texts and static rule-based limitations, potentially overlooking pertinent key-phrases outside predefined patterns.

Merging the strengths of both transformer and graph-based methodologies presents a holistic view of current key-phrase extraction efforts, revealing a landscape where significant strides in accuracy and efficiency are occasionally hindered by resource dependencies and adaptability to specialized content. This synthesis underscores the ongoing need for research that balances computational feasibility with the intricacies of language, pushing the boundaries of what automated key-phrase extraction can achieve.

In this paper, our primary objective is the development and implementation of an algorithm, PatternMining, designed to significantly enhance the efficiency of key-phrase extraction. PatternMining introduces a novel methodology by utilizing regex-based patterns of Parts of Speech (POS) tags across N-gram key-phrases, meticulously mining and exploiting these syntactic cues to elevate the precision of extraction. By assigning weights according to the occurrence probabilities of these POS patterns, the algorithm optimizes the selection and ranking process for key-phrases, ensuring a prioritization that reflects their true relevance and syntactic integrity.

The cornerstone of PatternMining lies in its adaptability and compatibility, enabling its application atop existing key-phrase extraction frameworks, regardless of their foundational techniques. Whether integrated with transformer-based models, graph-based algorithms, or any other NLP methodology, PatternMining aims to serve as a complementary layer that augments their performance, addressing domain-specific challenges and enhancing adaptability. By grounding our approach in the probabilistic assessment of POS tag patterns, we seek not only to refine the extraction of key-phrases but also to offer a versatile tool that can adapt across various domains and datasets. In doing so, PatternMining aspires to set a new benchmark for key-phrase extraction, offering a robust solution that enhances both the accuracy and efficiency of extracting meaningful key-phrases from the ever-expanding universe of textual data.

2 RELATED WORK

The domain of key-phrase extraction has significantly evolved, driven by breakthroughs in machine learning and deep learning. This evolution is crucial for distilling meaningful phrases from expansive textual datasets, a challenge that has become increasingly important in the information-rich digital era. Among the innovative contributions to this field, [1] PatternRank emerges as a pioneering method that leverages pretrained language models (PLMs) and part-of-speech (POS) tagging for unsupervised key-phrase extraction. This approach uniquely combines PLMs with POS patterns to enhance key-phrase selection and ranking, showcasing its prowess on benchmark datasets where it outperforms traditional methods in precision, recall, and F1-scores.

The introduction of the Key-phraseVectorizers package by the developers of [1] PatternRank exemplifies the model's adaptability, enabling the customization of POS patterns for key-phrase selection across various domains. This tool is particularly valuable for processing short texts, such as publication abstracts, by utilizing PLMs to rank candidate key-phrases based on semantic similarity, significantly capturing their relevance and importance.

Simultaneously, the [2] Semkey-BERT model utilizes BERT and sentence transformers to extract semantically rich key-phrases from large-scale social media datasets. This model underscores the importance of deep semantic analysis in processing short texts like tweets, achieving notable accuracy improvements over existing models. Semkey-BERT's methodology, encompassing preprocessing and innovative techniques for key-phrase ranking and selection, marks a substantial advancement in social media analytics and information retrieval.

Moreover, comprehensive analyses by various researchers delve into the effectiveness of computational tools in academic writing, exploring the relationship between self-mentions, keywords, and sentiment. [3] Their work sheds light on the discriminative selection of methods for keyword generation and sentiment analysis, contributing to our understanding of authorial stance and engagement.

Further explorations into keyword extraction utilize [4] Support Vector Machines (SVMs) to consider both global and local context information, proposing methods that significantly surpass traditional global context-reliant approaches. These methods' effectiveness, validated through extensive experimentation, demonstrates considerable improvements in extraction metrics, highlighting SVMs' potential in text mining applications.

[5] Survey works provide exhaustive overviews of automatic keyword extraction and text summarization interdependence, categorizing methodologies and summarizing techniques. These surveys identify ongoing challenges and future research directions, signifying the field's dynamic nature and the quest for innovation.

Additionally, innovative methods like [6] [7] TextRank-based keyword extraction from software requirements and interdisciplinary approaches combining probabilistic-entropy, graph theory, and neural networks further broaden the spectrum of strategies explored to enhance keyword extraction accuracy and relevance.

[8] [9] Lastly, the application of sequential pattern mining to keyword extraction introduces a promising direction for capturing more contextually significant keywords, showcasing potential improvements over existing methodologies. This approach is particularly relevant for applications in information retrieval, content management, and digital libraries.

These studies collectively chart the vibrant landscape of key-phrase extraction research, presenting a range of methodologies from deep learning models and graph-based algorithms to the innovative use of probabilistic-entropy and sequential pattern mining. Each contribution inches us closer to resolving the complex puzzle of key-phrase extraction, underscoring the need for ongoing exploration. Our work builds on these foundations, aiming to further the capabilities of key-phrase extraction techniques by addressing challenges related to domain specificity, computational efficiency,

and adaptability, thus enriching the toolkit for effective and efficient key-phrase extraction.

3 DATA

3.1 Data Description

The Inspec dataset is a crucial benchmark for evaluating key-phrase extraction and generation models in Natural Language Processing (NLP). It consists of 2,000 scientific paper abstracts sourced from the extensive Inspec database, which encompasses a wide array of disciplines. These abstracts are complemented with detailed annotations, including key-phrases tagged by professional indexers in an uncontrolled environment, free from the constraints of a fixed thesaurus. This richly annotated dataset provides an array of data fields for each document:

- **id**: A unique identifier for each document in the dataset.
- **title**: The title of the scientific paper.
- **abstract**: The abstract of the paper, providing a succinct summary of the content.
- **key-phrases**: A carefully compiled list of reference key-phrases annotated by indexers.
- **prmu**: Categories for each reference key-phrase, classified under the PRMU scheme, which stands for Present-Reordered-Mixed-Unseen, as conceptualized by Boudin and Gallina in 2021.

This PRMU scheme is an insightful addition, offering a sophisticated tool for assessing the performance of key-phrase extraction models. It considers various conditions in which key-phrases appear, such as their explicit presence in the text, alterations in order, mixtures of words, or instances where they might not be directly visible in the text yet remain topically relevant. The inclusion of these varied categories allows for a comprehensive understanding and testing of the algorithms' capabilities to detect and generate key-phrases that truly encapsulate the crux of the scientific documents. With this annotated and categorized corpus, researchers have the opportunity to thoroughly evaluate and enhance their key-phrase extraction methods, thereby pushing forward the frontiers of research in the NLP domain.

3.2 Data Exploration

Data within the Inspec dataset is organized into three distinct splits: training, validation, and testing, with 1,000, 500, and 500 documents respectively. The average abstract contains approximately 118.94 words, with an average of 9.81 key-phrases per abstract. These key-phrases vary in length, providing a comprehensive range of n-gram structures for analysis. The dataset includes several key fields for each document, including a unique identifier (id), the title of the document, the abstract text, a list of reference key-phrases, and the PRMU categorization for each key-phrase.

Figure 1 illustrates the distribution of N-gram key-phrases within the dataset, presenting an overview of the complexity and variety of the key-phrase structures. The dataset predominantly contains bigram (2-word) key-phrases, with a count of 5,080 instances, followed by 2,489 trigram (3-word) key-phrases, and 1,346 unigram (1-word) key-phrases. Less frequent are key-phrases of lengths four to eight words, indicating a higher prevalence of shorter, more concise key-phrases in scientific abstracts.

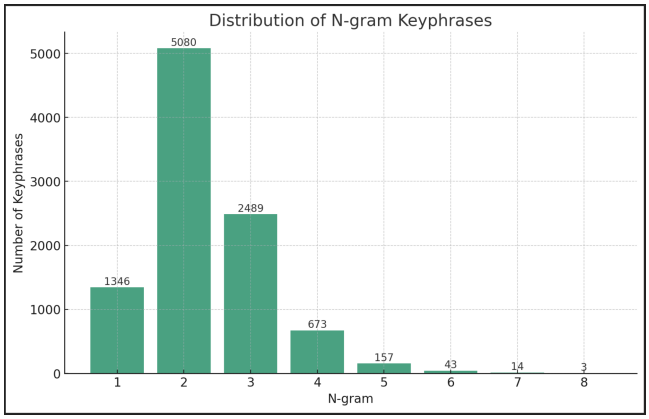

Figure 1: Distribution of N-gram key-phrases in Inspec Dataset

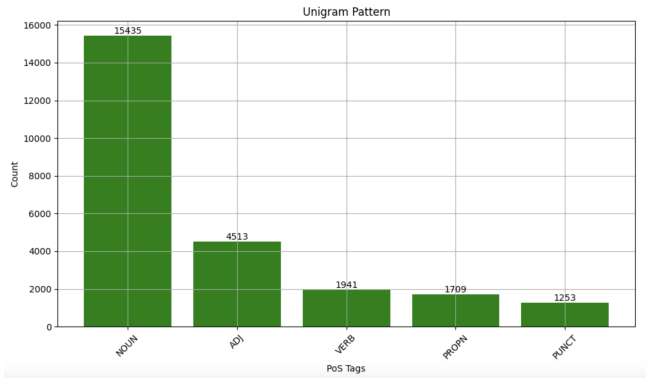

Figure 2: POS Tag Frequency of Unigrams in key-phrases

Upon meticulous examination of syntactic patterns within unigrams of our annotated key-phrases, we observed an enlightening trend: the Noun-Adjective-Verb (NAV) sequence forms the backbone of approximately 90% of all identified key-phrases as shown in Figure 2. This prevalence of the NAV pattern is not merely coincidental but indicative of a fundamental syntactic cornerstone that key-phrases tend to revolve around. The discovery of this NAV dominance has profound implications for the design of key-phrase extraction algorithms. It prompted us to pivot our strategy towards harnessing the power of POS tag patterns within unigrams. Rather than following conventional paths, we envisioned a strategy that would tap into the predictive power of POS tag patterns in unigrams. This conceptual pivot laid the groundwork for a potential algorithm that could take advantage of this syntactic tendency. The envisioned algorithm aims to employ the NAV pattern as a heuristic for more accurately identifying and extracting key-phrases, harnessing the syntactic regularity we uncovered to enhance the semantic depth of key-phrase extraction in large textual datasets.

4 METHODS

4.1 Preprocessing

Preprocessing of the dataset's text is a multi-tiered approach aimed at standardizing the input for more effective pattern analysis. Initially, we perform lowercasing to eliminate the variability caused by capitalization, thereby reducing the complexity of the data and ensuring that words with the same root are processed uniformly. Following this, tokenization is conducted using the Spacy library's 'en-core-web-sm' model, which includes a specialized rule to avoid splitting hyphenated words such as 'graph-based', preserving their integrity as single tokens.

Lemmatization is then applied to each token, a crucial step that strips words down to their base or dictionary form, further reducing inflectional forms and derivationally related forms of a word to a common base form. Accompanied by stemming, which is performed using Porter's stemmer from the Natural Language Toolkit (NLTK), these processes together serve to align the extracted key-phrases with their root forms in the source text, a vital aspect for consistent key-phrase identification.

Subsequently, each word within the dataset's key-phrases undergoes Part-Of-Speech (POS) tagging through the Spacy library to ascertain its grammatical role. This step is instrumental in defining the grammatical context of words, crucial for the subsequent pattern recognition tasks. The identified POS tags are then meticulously converted into regular expression (regex) patterns. This conversion is essential as it maps the syntactic sequences of the words, enabling the recognition of grammatical structures in the pattern mining process. By capturing the syntactic essence of key-phrases through regex patterns, we establish a systematic foundation for our Pattern Mining technique. This comprehensive preprocessing phase not only simplifies the textual data but also enriches the syntactic and morphological quality of the input, paving the way for a precise and methodical key-phrase extraction.

4.2 PatternMining Algorithm

Pattern Mining focuses on Part-Of-Speech (POS) tag patterns within unigrams to refine key-phrase extraction. This method is centered on leveraging the syntactic structures indicated by POS tags, which are pivotal in representing the underlying syntax of phrases within the dataset. By quantifying the occurrence of these POS patterns, we gain insights into the prevalence of specific syntactic forms across the corpus.

In the development of the PatternMining algorithm, it was evident that the majority of key-phrases were composed predominantly of Noun-Adjective-Verb (NAV) patterns, a pivotal idea emerged. We postulated that by inverting the traditional key-phrase extraction process, we could enhance the algorithm's effectiveness. Instead of directly seeking n-gram key-phrases, we would concentrate on extracting significant unigrams with a high likelihood of forming part of a key-phrase based on their POS tags. These unigrams could then be strategically assembled to construct n-gram key-phrases. This reverse-engineering approach could potentially streamline the identification of relevant key-phrases by focusing on the syntactic foundation from which they are built.

Following the selection of unigrams that fit the NAV pattern, the subsequent step is to discern and prioritize the most significant ones. For this purpose, we employ the TextRank algorithm, which operates on the principle of capturing the importance of words within a graph-based model. TextRank evaluates the relationships between words in the text and assigns a score to each unigram, effectively mirroring their perceived importance based on their connections within the graph. This process yields a ranked list of unigrams, systematically spotlighting those that are most pivotal within the NAV pattern. The outcome is a set of prioritized unigrams, from which we can construct key-phrases with a high likelihood of embodying the core thematic elements of the corpus.

After the extraction of unigrams that align with the NAV pattern, it is crucial to evaluate and quantify the importance of each unigram. This importance is represented as a score that reflects the potential of each unigram to contribute meaningfully to a key-phrase. While the TextRank algorithm provides initial rankings based on intra-document relationships, it lacks the ability to incorporate broader linguistic knowledge from external sources.

To address this limitation, we further refine our approach by utilizing the T5 (Text-to-Text Transfer Transformer) model, a pre-trained transformer that has a comprehensive understanding of language derived from its training on extensive corpora. By inputting the document and the extracted unigrams into the T5 model, we are able to obtain scores that not only reflect the internal textual context but also leverage external linguistic insights. These scores for the unigrams, now termed 'Candidate Unigrams,' offer a more nuanced reflection of each unigram's relevance, incorporating both the specific context of the document and the general importance of the words as learned from diverse linguistic data. This dual-level analysis ensures a more robust and informed selection of key unigrams, significantly enhancing the precision of our key-phrase extraction process.

Following the scoring of Candidate Unigrams using the T5 model, the next phase involves constructing n-gram phrases from these scored unigrams. To achieve this, we implemented a Sliding Window algorithm with a configurable window of length 'n'. This algorithm traverses through the document, combining various configurations of the Candidate Unigrams to form potential n-gram phrases.

The weighting of these n-gram phrases is determined by a comprehensive aggregation method. Specifically, the weight of each n-gram is computed by aggregating the individual scores of the unigrams that constitute the n-gram. This aggregation not only reflects the importance of each unigram within the broader textual context but also incorporates a multiplicative factor from the probabilities of the corresponding POS regex patterns. This factor enhances the relevance measure by ensuring that syntactically coherent combinations of unigrams, which are more likely to form meaningful phrases, receive higher scores. Thus, this approach integrates both the semantic significance and the syntactic propriety of the phrases, culminating in a robust method for key-phrase generation.

Once the n-grams are formed and weighted, the algorithm selects the top 'n' key-phrases based on their aggregated scores. These key-phrases represent the most contextually and syntactically relevant phrases within the document, effectively summarizing its key content.

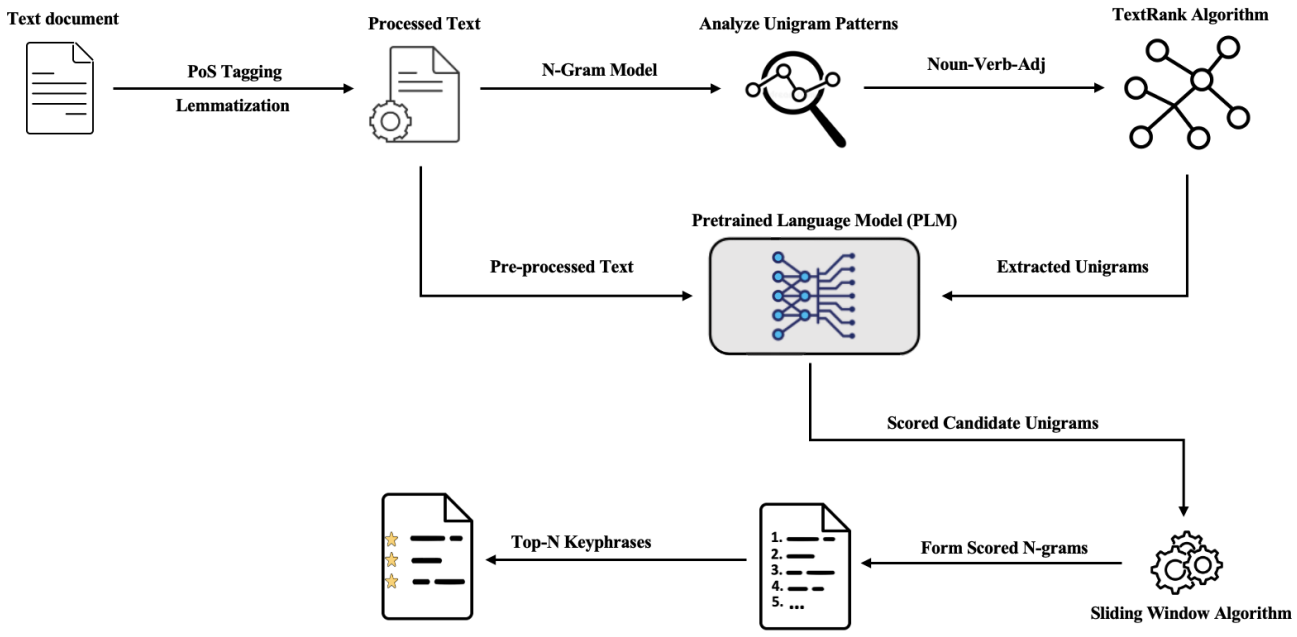

Figure 3: Model Architecture

4.3 Evaluation and Metrics

In evaluating the performance of key-phrase extraction models, the **F1 score** is a critical metric that balances the trade-off between precision and recall. The F1 score can be formally defined as the harmonic mean of precision and recall, providing a single measure to assess the model's accuracy in identifying relevant items. The formula for the F1 score is given as follows:

$$F1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{1}$$

Precision and recall, the components of the F1 score, are defined as follows:

- **Precision**, measures the proportion of true positive predictions in the set of all positive predictions made by the model. It is calculated using the formula, where TP represents the number of true positives, and FP represents the number of false positives.

$$\text{precision} = \frac{TP}{TP + FP} \tag{2}$$

- **Recall**, measures the proportion of true positives that were identified correctly by the model out of all actual positives. It is calculated as the formula mentiond below, where FN represents the number of false negatives.

$$\text{recall} = \frac{TP}{TP + FN} \tag{3}$$

We also utilized a comprehensive framework to assess the performance of key-phrase extraction models. We employ three evaluation strategies: **Exact match**, **Partial match**, and **Average match** i.e. mean of exact and partial match scores. For each model, we calculate Precision@N, Recall@N, and F1@N scores, considering the top-N extracted key-phrases for analysis. The set of gold key-phrases, against which we benchmark all extracted key-phrases, comprises the entire corpus of manually assigned key-phrases, irrespective of N. Furthermore, to ensure consistency, both the gold and extracted key-phrases are converted to lowercase and duplicates are removed.

[10] Following the method outlined by Rousseau and Vazirgiannis (2015), we compute these metrics for each document individually and then aggregate the results using a macro-average across the entire dataset. This method effectively captures the overall performance of the models across diverse textual content.

The exact match evaluation strictly identifies true positives as those extracted key-phrases that perfectly match the gold standard key-phrases. While this method is stringent, it can penalize models that generate semantically similar but syntactically varied key-phrases compared to the gold standard (Rousseau and Vazirgiannis, 2015; Wang et al., 2015). To address this limitation, the partial match evaluation considers both the gold and extracted key-phrases at the unigram level. A true positive in this scenario occurs when any of the unigram components of the extracted key-phrases matches with any from the gold key-phrases, thus acknowledging the semantic value of the extracted phrases even when exact syntactic correspondence is lacking. This approach offers a more nuanced assessment of a model's ability to capture the essential themes and concepts of the text.

4.4 Experimental Design

Initial Hypothesis: To determine whether we could mine patterns in the structure of key-phrases using regex representations of their

POS tags. By converting all the annotated key-phrases into a regex of POS tags, we identified a total of 9,805 key-phrases, among which only 668 were unique. This approach, termed Pattern Mining, harnesses the syntactic structures of text to refine key-phrase extraction. It relies on regex patterns, derived from POS tagging, to represent the syntax of phrases within the dataset. Each pattern's occurrence is quantified, reflecting the frequency of syntactic forms across the corpus.

The quantified data is sorted by n-gram length, from unigrams to extended phrases, enabling precise probability assessments for words or phrases to be considered key-phrases. This sorting allows for nuanced probability-based analysis tailored to each n-gram category.

Probabilities from this analysis act as statistical weights in Pattern Mining. When evaluating key-phrase candidates, these probabilities are employed to adjust the scores from key-phrase extraction models like T5 or TextRank. This synthesis of syntactic probabilities with semantic and structural evaluations from existing models provides a composite score—a holistic metric that embodies both semantic relevance and syntactic probability. This composite score is crucial for key-phrase ranking, combining established model outputs with Pattern Mining's syntactic probabilities to yield a robust, multidimensional approach to key-phrase extraction.

Challenges with Viterbi Algorithm: Upon assigning probabilities to a regex using the Viterbi Algorithm and incorporating this probability as a weight to rank the key-phrases extracted by the T5 model, the results were not as encouraging as anticipated. This unexpected outcome prompted a reevaluation of our approach.

Revised Methodology: Subsequent analysis revealed a distinctive pattern in the POS tags of unigrams present in the annotated key-phrases. This discovery led us to refine our methodology to better capture and utilize these patterns.

Implementation and Performance Comparison: Initially, the method was implemented and tested locally on macOS. To further assess and enhance performance, we transferred our experiments to Google Colab. The use of Colab, with its more robust computational resources (including high RAM and GPU support), yielded significantly improved results. This platform provided a more conducive environment for handling the intensive computational demands of our NLP tasks, leading to better algorithm performance and more reliable data processing.

5 RESULTS

The analysis of key-phrase extraction methods as depicted in the Table 1 reveals insightful trends and performance benchmarks for each model across varying retrieval sizes. $\text{PatternRank}_{\text{PoS}}$ consistently achieved high precision and F1 scores, notably at the @20 retrieval size. This suggests that the model is adept at identifying the exact lexical frame of key-phrases when provided with a larger pool of candidates. The precision in $\text{PatternRank}_{\text{PoS}}$ underscores its algorithmic capability to discern and prioritize the most relevant phrases, a desirable feature for applications requiring high specificity in key-phrase identification.

$\text{PatternMining}_{\text{NAV}}$, while showing lower precision in exact matches, demonstrates a higher recall, particularly at @10 and @20. This high recall is indicative of the model's strength in identifying a broader array of relevant key-phrases. The lower precision may be attributed to its more inclusive selection criteria, which, while capturing a wide spectrum of pertinent phrases, may also include less relevant candidates. This trade-off between precision and recall is a critical consideration for end-user applications where the breadth of information retrieval is prioritized over the pinpoint accuracy of individual phrases.

The Partial Match category provides a different perspective, emphasizing the model's ability to recognize key-phrases that may not be exact but are semantically related. Here, $\text{PatternMining}_{\text{NAV}}$ outperforms $\text{PatternRank}_{\text{PoS}}$ in F1 scores at @20, highlighting its proficiency in capturing the essence of key-phrases even when the exact wording is not matched. This is particularly useful in exploratory search scenarios where variations of key-phrases could lead to the discovery of related concepts or themes.

For Average Match, which averages the performance across exact and partial matches, both $\text{PatternMining}_{\text{NAV}}$ and $\text{PatternRank}_{\text{PoS}}$ show a balanced precision and recall. This balance is crucial for systems that require a harmonized approach to key-phrase extraction, neither skewed towards inclusivity at the cost of relevance nor towards exclusivity at the risk of omitting related phrases.

YAKE and SingleRank models display moderate performance, with lower F1 scores in Exact Match but better performance in Partial and Average Match categories. This might suggest that these models are more attuned to capturing general key-phrase themes rather than specific terminologies. KeyBERT's overall lower performance across the match types could indicate a propensity for over-specification, potentially overlooking semantically related key-phrases that deviate slightlyfrom the standard nomenclature or phrasing. KeyBERT's stringent adherence to exactness might, therefore, limit its utility in applications where semantic variation is both expected and beneficial.

The nuanced differential in performance across the models hints at an underlying divergence in methodological approaches to key-phrase extraction. Models like $\text{PatternRank}_{\text{PoS}}$ optimize for high precision, reflecting a methodology finely tuned to the retrieval of highly relevant and specific phrases at the expense of broader thematic coverage. In contrast, $\text{PatternMining}_{\text{NAV}}$ demonstrates an affinity for recall, suggesting a model design that favors the extensive retrieval of content, thereby increasing the probability of capturing diverse but pertinent phrases.

These findings have profound implications for the implementation of key-phrase extraction systems in real-world scenarios. The choice between $\text{PatternRank}_{\text{PoS}}$ and $\text{PatternMining}_{\text{NAV}}$ could be contingent upon the specific use case requirements—be it the precise extraction of key terms for high-stakes data analysis or the comprehensive aggregation of concepts for content discovery.

6 DISCUSSION

The analytical investigation of the $\text{PatternMining}_{\text{NAV}}$ model, as delineated in Table 1, underscores its robustness in key-phrase extraction when benchmarked against conventional state-of-the-art methodologies. It's noteworthy that $\text{PatternMining}_{\text{NAV}}$ demonstrates superior recall in the partial match category, indicative of its ability to capture an expansive set of relevant key-phrases. The model's discerning approach to phrase selection, which leverages

Method	@5			@10			@20		
	P	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**
Exact Match									
YAKE	26.16	11.71	15.37	20.88	18.45	18.50	16.45	27.78	19.65
SingleRank	38.11	16.55	21.97	33.29	27.27	28.55	27.24	38.84	30.80
KeyBERT	12.97	6.08	7.82	11.42	10.53	10.30	9.75	17.14	11.76
$\text{PatternRank}_{\text{PoS}}$	41.76	18.44	24.35	36.10	29.63	30.99	27.80	39.42	31.37
$\text{PatternMining}_{\text{NAV}}$	30.45	17.77	21.66	38.07	28.65	**32.90**	19.28	36.80	24.30
Partial Match									
YAKE	77.45	19.49	29.91	68.20	33.46	42.67	59.69	45.58	48.69
SingleRank	75.54	19.36	29.56	68.63	33.98	43.24	58.82	53.68	53.68
KeyBERT	77.48	20.06	30.55	65.78	32.90	41.67	57.11	45.37	48.34
$\text{PatternRank}_{\text{PoS}}$	82.49	21.61	32.79	74.79	37.50	47.48	63.21	57.66	57.71
$\text{PatternMining}_{\text{NAV}}$	66.05	39.44	**46.40**	59.24	54.81	**53.77**	54.09	67.12	**59.90**
Avg. Match									
YAKE	51.81	15.60	22.64	44.54	25.96	30.59	38.07	36.68	34.13
SingleRank	56.83	17.96	25.77	50.96	30.63	35.90	43.03	46.26	42.24
KeyBERT	45.23	13.07	19.19	38.60	21.72	25.99	33.43	31.23	30.05
$\text{PatternRank}_{\text{PoS}}$	62.13	20.03	28.57	55.45	33.57	39.24	45.51	48.54	44.54
$\text{PatternMining}_{\text{NAV}}$	48.26	28.60	**33.78**	42.88	41.73	**39.96**	34.18	51.96	39.18

Table 1: Performance Evaluation of the PatternMining Approach Versus State-of-the-Art Models on the Inspec Dataset. The table quantifies Precision (P), Recall (R), and F1-Score (F1) for top N = 5, 10, and 20 retrieved key-phrases. Evaluation metrics are computed based on exact match, partial match, and the average of the two, demonstrating the efficacy of our $\text{PatternMining}_{\text{NAV}}$ model against established methods.

both syntactic and semantic patterns, attributes to this success. Notwithstanding, the precision at broader retrieval sizes suggests a propensity towards inclusivity, which, while beneficial in capturing thematic breadth, could integrate non-essential phrases.

7 CONCLUSION

Our $\text{PatternMining}_{\text{NAV}}$ framework stands out as a compelling advancement in the realm of key-phrase extraction. It effectively balances the nuanced demands of high recall and respectable precision, thereby addressing the limitations of previous models which often skewed towards one metric at the expense of the other. This equilibrium positions $\text{PatternMining}_{\text{NAV}}$ as a versatile tool, suitable for a wide array of applications ranging from academic research to comprehensive content analysis in various domains.

7.1 Future Work

The prospective trajectory for enhancing the $\text{PatternMining}_{\text{NAV}}$ model includes refining the balance between precision and recall. This involves the development of advanced filtering mechanisms that can differentiate with higher specificity between relevant and peripheral phrases within the inclusion criteria. Additionally, future iterations will explore the integration of dynamic context-aware algorithms to adapt the model's performance across diverse datasets and domains.

Further investigations will also focus on the scalability of the model, ensuring that it maintains efficacy as data volume and complexity grow. Integrating $\text{PatternMining}_{\text{NAV}}$ with other AI components, such as recommendation systems and automated content summarization tools, could broaden its applicability, providing a springboard for interdisciplinary research and practical applications.

In the pursuit of continuous improvement, we will solicit feedback from end-users and subject matter experts to inform the iterative development process. By incorporating real-world insights into the model's evolution, we aim to align our technological progress with the practical needs and challenges faced by users across various sectors.

Ultimately, the envisioned future for $\text{PatternMining}_{\text{NAV}}$ is one where it not only leads as a key-phrase extraction tool but also serves as a foundational component in a suite of advanced text analysis and information retrieval systems.

REFERENCES

[1] Schopf, T., Klimek, S., and Matthes, F. 2022. PatternRank: Leveraging Pretrained Language Models and Part of Speech for Unsupervised key-phrase Extraction. In Proceedings of the 14th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management - KDIR 2022, 243-248. DOI: 10.5220/0011546600003335.

[2] Devika, R., Vairavasundaram, S., Mahenthar, C. S. J., Varadarajan, V., and Kotecha, K. 2021. A Deep Learning Model Based on BERT and Sentence Transformer for Semantic key-phrase Extraction on Big Social Data. IEEE Access, 9, 165252–165261. DOI: 10.1109/ACCESS.2021.3133651.

[3] Thushara, M. G., Mownika, T., and Mangamuru, R. 2019. A Comparative Study on different Keyword Extraction Algorithms. In Proceedings of the 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 969–973. DOI: 10.1109/ICCMC.2019.8819630.

[4] Zhang, K., Xu, H., Tang, J., Li, J. 2006. Keyword Extraction Using Support Vector Machine. In: Yu, J.X., Kitsuregawa, M., Leong, H.V. (eds) Advances in Web-Age Information Management. WAIM 2006. Lecture Notes in Computer Science, vol 4016. Springer, Berlin, Heidelberg. https://doi.org/10.1007/11775300_8.

[5] Bharti, S. K., Babu, K. S., and Jena, S. K. 2017. Automatic Keyword Extraction for Text Summarization: A Survey. arXiv preprint arXiv:1704.03242. Available at: https://arxiv.org/abs/1704.03242.

[6] Delgado-Solano, I. P., Núñez-Varela, A. S., & Pérez-González, G. H. 2018. Keyword Extraction From Users' Requirements Using TextRank and Frequency Analysis, and Their Classification into ISO/IEC 25000 Quality Categories. In 2018 6th International Conference in Software Engineering Research and Innovation (CONISOFT), San Luis Potosi, Mexico, 88–92. DOI: 10.1109

[7] Selivanov, A. A., Moloshnikov, I. A., Rybka, R. B., & Sboev, A. G. 2020. Keyword Extraction Approach Based on Probabilistic-Entropy, Graph, and Neural Network Methods. In Proceedings of the Russian Conference on Artificial Intelligence (RCAI 2020), Artificial Intelligence, Lecture Notes in Computer Science (LNAI), Vol. 12412, pp. 284–295. Springer. DOI: 10.1007/978-3-030-59535-7_21.

[8] Feng, J., Xie, F., Hu, X., Li, P., Cao, J., & Wu, X. 2011. Keyword Extraction Based on Sequential Pattern Mining. In Proceedings of the Third International Conference on Internet Multimedia Computing and Service (ICIMCS '11), New York, NY, USA, 34–38. Association for Computing Machinery. https://doi.org/10.1145/

[9] Jiajia Feng, Fei Xie, Xuegang Hu, Peipei Li, Jie Cao, and Xindong Wu. 2011. Keyword extraction based on sequential pattern mining. In Proceedings of the Third International Conference on Internet Multimedia Computing and Service (ICIMCS '11). Association for Computing Machinery, New York, NY, USA, 34–38. https://doi.org/10.1145/2043674.2043685

[10] F. Rousseau and M. Vazirgiannis. 2015. Main core retention on graph-of-words for single-document keyword extraction. In A. Hanbury, G. Kazai, A. Rauber, and N. Fuhr (Eds.), Advances in Information Retrieval (pp. 382–393). Cham: Springer International Publishing. DOI:10.1007/978-3-319-16354-3_42

[11] Papagiannopoulou, E. and Tsoumakas, G. (2019). A review of keyphrase extraction. Wiley Interdisciplinary Re- views: Data Mining and Knowledge Discovery, 10.

[12] Reimers, N. and Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Proceedings of the 2019 Conference on Empiri- cal Methods in Natural Language Processing and the 9th International Joint Conference on Natural Lan- guage Processing (EMNLP-IJCNLP), pages 3982– 3992, Hong Kong, China. Association for Computa- tional Linguistics.

[13] Rose, S. J., Engel, D. W., Cramer, N., and Cowley, W. (2010). Automatic keyword extraction from individ- ual documents.

[14] Schopf, T., Braun, D., and Matthes, F. (2021). Lbl2vec: An embedding-based approach for unsupervised doc- ument retrieval on predefined topics. In Proceedings of the 17th International Conference on Web Informa- tion Systems and Technologies - WEBIST, pages 124– 132. INSTICC, SciTePress.

[15] Schopf, T., Weinberger, P., Kinkeldei, T., and Matthes, F. (2022). Towards bilingual word embedding models for engineering: Evaluating semantic linking capabil- ities of engineering-specific word embeddings across languages. In 2022 4th International Conference on Management Science and Industrial Engineering (MSIE), MSIE 2022, page 407–413, New York, NY, USA. Association for Computing Machinery.

[16] Sharma, P. and Li, Y. (2019). Self-supervised contextual keyword and keyphrase retrieval with self-labelling.