WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Keyword Extraction and Text Summarization in Emails

I. Ishitha

Department of CSE, MGIT (A), Gandipet, Hyderabad, 500075, Telangana, India.
Email:ilakshmi_cse210585@mgit.ac.in

ABSTRACT

Business people receive an average of 200 emails per day, which will take a long time to read. The purpose of this project is to enable automatic summarization of email text and keywords to be derived so that users have the important information at hand fast. receive an average of 200 emails per day, which will take a long time to read them all. The objective of this project is to automatically carry out email text summarization and keyword extraction in such a way that users can quickly grasp key information. Pre-trained transformer models, for example, T5, are applied for abstractive summarization. The source of data is the ENRON dataset, which comprises about 500,000 emails.

Keywords: Keyword Extraction, Text Summarization, BERT, NER, Text Rank.

1. Introduction

Email constitutes an important means of communication in our daily exchanges, used not only for personal conversation but also as a repository of corporate information. Given the overwhelming number of emails that we have to handle on an everyday basis, it is becoming increasingly important to have efficient access to the important information contained in emails.

Summarization and key-phrase extraction are two complementary techniques which, given a natural language text, extract the most important sentences and the most important words or phrases. Therefore, when applied to an email or an email thread, it is expected that these two procedures, when suitably implemented, would give us the most important sentences and words/phrases contained in them, thereby effectively giving us a “snippet” of the text and mostly reducing the time needed to read an entire email. Once a user reads the snippet, they can either choose to read the complete email/email thread, or the user can choose to read it later. This is similar to the current techniques for Web search, where we read the snippets to determine the purported “value/relevance” of a particular web page rather than going inside each and every web page to determine its value. So effectively, our technology will make the task of processing email more efficient, by assisting people to prioritize email and by giving people a choice between reading an email right away, removing it, or postponing it for future. Several datasets have been released for general-purpose summarization and keyword extraction, but very few of them specifically deal with emails.

Emails have a special graph structure that warrants more intricate treatment than that needed by other types of text documents. The only email summarization corpus we are aware of is due to (Ulrich et al., 2008). This corpus (BC3) comprises 40 email threads (3222 sentences) with annotations for extractive and abstractive summarization, speech act, meta sentences, and subjectivity. While important for being a path-breaker in email summarization research, the corpus is relatively small, it does not give a ranked list of extracted sentences, there is no control over the number of sentences extracted, and perhaps most importantly for our goals, the corpus does not include keywords. The only corpus for keyword extraction from emails (Turney, 2000) has never been released publicly. The corpus, consisting of a total of more than 100,000 words, is available upon request, and thus it is likely to enable new research in this area

1.1 Problem Statement

Growth in the number of research documents getting published is increasing. Finding a research document under interested domain by referring the whole paper has become a tedious task. Keywords, Key phrases gives the summary of the text. Keywords and key phrases help in understanding the information described in the research document. The domain of a research document can be determined based on the keywords and key phrases extracted. Extracting keywords and key phrases manually is a tedious task. Automatic key phrase extraction techniques help in overcoming this challenging task. This paper is a comparative study of unsupervised key phrase extraction algorithms without using corpus. It compares the performance of Position Rank which considers the position of the all words occurrences in the document with RAKE (Rapid Automatic Keyword Extraction).

1.2 Existing System

The data set utilized in this project is the ENRON Email Dataset, a widely utilized and publicly accessible data set of real emails of the erstwhile Enron Corporation. That corpus contains roughly 10,000 messages, loaded with unstructered text data to be processed using natural language processing. The messages are kept in PST (Personal Storage Table) files, a native Microsoft Outlook file format to store the email data. PST files are difficult to process directly as they are nested and compound, and special parsing techniques need to be employed to extract useful text from them.

New in this work is the end-to-end pre-processing pipeline of data performed to enable the raw and unstructured nature of PST files. The emails are subjected to different cleaning, transformation, and analysis tasks. Major natural language processing techniques are utilized, such as:

• BERT (Bidirectional Encoder Representations from Transformers) for classification and unwanted content removal such as email signatures.

• Named Entity Recognition (NER) to pull out and tag significant entities (places, names, organizations) from the body of the email.

• Graph-based algorithmic method TextRank uses keyword extraction through the study of the co-occurrence pattern of the words.

• T5 (Text-to-Text Transfer Transformer), a transformer-based model, is an abstractive text summarization model used to generate brief and descriptive summaries of email content.

1.3 Proposed System

It is devoted to a comparative study of single unsupervised key phrase extraction algorithms. The primary aim is to identify the best strategies towards automatic frequent keyword and key phrase detection from text data without previous domain-specific training. The work compares Position Rank, an algorithm for maximizing the keyword importance based on the positional importance of words in the document, with other well-known algorithms such as Text Rank and RAKE (Rapid Automatic Keyword Extraction).

Text Rank is a PageRank-style graph-ranking algorithm with words as nodes and edges as co-occurrence. It strongly ranks the most central words based on connectivity in the text. RAKE is a lightweight yet efficient algorithm that extracts keyword phrases considering both word frequency of occurrence and word co-occurrence.

By cross-comparison of these methodologies, the study will be in a position to determine which algorithm yields more precise and contextual keywords, thereby facilitating information retrieval and summarization.

2. Literature Review

[1] M. G. Thushara, T. Mownika and R. Mangamuru, "A Comparative Study on different Keyword Extraction Algorithms", ICCMC, 2019.

Since there is an increasing number of research documents published every year, the documents available on the Internet will also be increasing rapidly. This poses the need to categorize the available research articles into their respective domain to ease the search process and find their research documents under the specific domain. This classification is a tiresome and prolonged process, which can be avoided by using keywords and key phrases. Key words or key phrases provides a summary or information described in a research document. The domain of a research paper can be determined based on extracted key words and key phrases. It is monotonous to manually extract keywords and key phrases. Automatic extraction of keyword techniques helps to overcome this challenging task. The classification of these research papers can be achieved more efficiently by using the keywords applicable to a particular domain. This survey aims to compare key extraction algorithms such as Text Rank, Position Rank, key phrase extraction algorithm (KEA) and Multi-purpose automatic topic indexing (MAUI).

[2] Y. Jin, C. Luo, W. Guo, J. Xie, D. Wu and R. Wang, "Text classification based on conditional reflection", IEEE Access, vol. 7, pp. 76712-76719, 2019.

Text classification is an essential task in many natural language processing (NLP) applications; each sentence may have only a few words that play an important role in text classification, while other words have no significant effect on the classification results. Finding these keywords has an important impact on the classification accuracy. In this survey, a network model is proposed, named RCNNA, recurrent convolution neural networks with attention (RCNNA), which models on the human conditional reflexes for text classification. The model combines bidirectional LSTM (BLSTM), attention mechanism, and convolutional neural networks (CNNs) as the receptors, nerve centers, and effectors in the reflex arc, respectively. The receptors get the context information through BLSTM, the nerve centers get the important information of the sentence through the attention mechanism, and the effectors capture more key information by CNN. Finally, the model outputs the classification result by the softmax function.

NLP algorithm is tested on four datasets containing Chinese and English for text classification, including a comparison of random initialization word vectors and pre training word vectors. The experiments show that the RCNNA achieves the best performance by comparing with the state-of-the-art baseline methods.

[3] Á. Hernández-Castañeda, R. A. García-Hernández, Y. Ledeneva and C. E. Millán- Hernández, "Extractive automatic text summarization based on lexical semantic keywords", IEEE Access, vol. 8, pp. 49896-49907, 2020.

The automatic text summarization (ATS) task consists in automatically synthesizing a document to provide a condensed version of it. Creating a summary requires not only selecting the main topics of the sentences but also identifying the key relationships between these topics. Related works rank text units (mainly sentences) to select those that could form the summary. However, the resulting summaries may not include all the topics covered in the source text because important information may have been discarded. In addition, the semantic structure of documents has been barely explored in this field. Thus, this study proposes a new method for the ATS task that takes advantage of semantic information to improve keyword detection. This proposed method increases not only the coverage by clustering the sentences to identify the main topics in the source document but also the precision by detecting the keywords in the clusters. The experimental results of this work indicate that the proposed method outperformed previous methods with a collection.

[4] S. K. Bharti and K. S. Babu, "Automatic keyword extraction for text summarization: A survey", arXiv preprint arXiv:1704.03242, 2017.

In recent times, data is growing rapidly in every domain such as news, social media, banking, education, etc. Due to the excessiveness of data, there is a need of automatic summarizer which will be capable to summarize the data especially textual data in original document without losing any critical purposes. Text summarization is emerged as an important research area in recent past. In this regard, review of existing work on text summarization process is useful for carrying out further research. In this paper, recent literature on automatic keyword extraction and text summarization are presented since text summarization process is highly depend on keyword extraction. This literature includes the discussion about different methodology used for keyword extraction and text summarization. It also discusses about different databases used for text summarization in several domains along with evaluation matrices. Finally, it discusses briefly about issues and research challenges faced by researchers along with future direction.

[5] M. Zhang, X. Li, S. Yue and L. Yang, "An empirical study of text rank for keyword extraction", IEEE Access, vol. 8, pp. 178849-178858, 2020.

As a typical keyword extraction technology, Text Rank has been used in a wide variety of commercial applications, including text classification, information retrieval and clustering. In these applications, the parameters of Text Rank, including the co-occurrence window size, iteration number and decay factor, are set roughly, which might affect the effectiveness of returned results. In this work, we conduct an empirical study on Text Rank, towards finding optimal parameter settings for keyword extraction. The experiments are done in Hulth2003 and Krapivin2009 datasets, which are two real datasets. We first remove the stop word by an open published English stop word list XPO6. And then, we extract the word stems by Porter Stemmer. Porter Stemmer is a tool which can find the stems of words with multiple variants, discard redundant information, strengthen the filtering effect, and extract the effective features of the text fully. We carry out extensive experiments to evaluate the effects of the parameters to keywords extraction, and evaluate the effectiveness of corresponding results by Precision, Recall and Accuracy. Experimental results show that Text Rank shows the best performance when setting co-occurrence window size w = 3, iteration number t = 20, decay factor c = 0.9 and rank k = 10 respectively, and the results are independent of the text length

[6] Y. Liu and M. Lapata, "Text Summarization with Pretrained Encoders," Proceedings of EMNLP, 2019.

This work introduces BERTSUM, a variant of BERT for abstractive and extractive summarization. The model leverages the pretrained contextual representations of BERT and fine-tunes them for use in summarization on benchmark data. It produced state-of-the-art results, setting the value in using transformer-based encoders for summarization. However, it is computationally expensive, something that may limit its use in light or real-time applications.

[7] J. Zhang, Y. Zhao, M. Saleh, and P. J. Liu, "PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization," Proceedings.of.ICML,.2020.

The authors present PEGASUS, a transformer model pre-trained on a novel objective—hiding complete sentences (gap-sentences) in a document and predicting them. PEGASUS outperforms BART and T5 on a range of summarization tasks like XSum and CNN/DailyMail. Its informative and coherent summaries are a great pick, although it remains computationally costly.

[8] I. H. Witten, G. W. Paynter, E. Frank, C. Gutwin, and C. G. Nevill-Manning, "KEA: Practical Automatic Keyphrase Extraction," DL'99:.Proceedings.of.the.ACM.Conference.on.Digital.Libraries,.1999.

KEA is a supervised learning algorithm utilizing a Naïve Bayes classifier to automatically extract keyphrases from text. It is feature-based on term frequency and first occurrence position. Extremely strong and widely referenced, it is not scalable as it is based on labeled training data while algorithms like TextRank or RAKE are unsupervised**.**

[9] F. Barrios, F. López, L. Argerich, and R. Wachenchauzer, "Variations of the Similarity Function of TextRank for Automated Summarization,".arXiv.preprint.arXiv:1602.03606,.2016.

In this paper, the authors explain how the sentence similarity function of TextRank can be adapted to improve the performance of extractive summarization. The authors perform experiments with different similarity measures like cosine similarity with TF-IDF or word embeddings to show extreme improvement in sentence relevance. As it's an extractive method, though, it does not generate new text or paraphrase ideas.

[10] E. Papagiannopoulou and G. Tsoumakas, "Local and Global Evaluation of Keyphrase Extraction," Information Processing & Management, 2020.

This work suggests a two-level evaluation framework—local (single document) and global (document-wise)—to quantify keyphrase extraction algorithms. It compares methods like TF-IDF, Text Rank, and Topic Rank, and shows that evaluation methods have a great influence on empirical algorithmic performance. This work is concerned only with unsupervised methods and not with deep learning methods.

2.1 Key Technologies used:

1) **Several Algorithms are Classifled under Natural Language Processing (NLP):**
    - Classification and Signature Stripping BERT (Bidirectional Encoder Representations from Transformers)content removal system. BERT's understanding of language context helps sentences to be effectively classified into two categories and works on binary classification tasks.
    - **T5 (Text-To-Text Transfer Transformer):** Works with abstractive text summarization. T5 changes every task in NLP to producing text and summarizes long emails into meaningful briefs.
    - **TextRank:** It is an unsupervised algorithm for keyword extraction that is graph based. Words are ranked according to how often they appear together in the text and the level of relevance they possess using the PageRank algorithm are utilized.
    - **Named Entity Recognition (NER):** Automated identification and extraction of particular names, dates, organizations and other details within an email. This helps to improved keyword extraction and recognition of context.
2) **Data Preprocessing & Transformation**
    - **Regular Expression Clean Up**: Information and Body Email Cleansing are done using expressions in removing structured patterns such as greetings and sign-offs and new attributes such as type of mail are created.
    - **Libratom Library:** A Python library known as libratom that can read and write files is capable of parsing PST (Personal Storage Table) files and changing them to CSV format. It reads archives of Outlook email in a hierarchical manner.
    - **TFDistilBERT (through TensorFlow):** Signature Stripping email the model is trained in removing email signatures in the process of classifying emails as binary - some will be appended, and others will be tagged as signatures.
3) **Advanced programming techniques and technologies**
    - **Python:** The most widely used programming language includes Python as the language of choice for model development, execution of workflows, and data cleaning.
    - **TensorFlow:** This training toolkit incorporates model signature classifiers using BERT.
    - **Hugging Face Transformers:** Together with summarization, this includes several examples of model fine-tuning and classification as applied to T5, and BERT.
    - **NLTK & SpaCy:** Both are libraries of NLP methods - including tokenization and sentence and word boundaries - and filtering for keywords and tokens.
4) **Development environment for collaborative development:**
    - **Google Collaboratory (Collab):** This allows for hosted Jupyter notebooks for model testing, training, and for executing code in the cloud.
    - **Flask (web framework):** This is a module action for developing a web interface to input content, extract keywords, and summarize.
    - **HTML, CSS, Bootstrap:** The application design for User's Email Analyzer was to represent the best of what could be possible.

2.2 Research Methodologies Used

This project was developed using different methods that systematically address the issue of summarization and keyword extraction from bulk emails. The main methodologies applied are as follows:

1) **Problem Definition and Scope Identification :**

This research problem arose from the general reading and processing of the huge corporate emails for the keyword extraction and summarizing automation on the NLP platform.

2) **Data Collection and Preparation**

- Doing clean analyses of the ENRON Email Dataset which consists on averagely five hundred thousand emails in a form of PST (Personal Storage Table) file.

- There is also a stated scope for using The Libratom library on the parsing and restructuring the PST files into a CSV format.
- There were data cleaning procedures like invalid email deletion, address anonymization, and standardization of the format.

3) Data Preprocessing and Feature Engineering :

- Regular expressions were employed for extracting email subject and body content.
- Classifying an email as forwarded, replied and normal based on the subject line and other data indicators made the classification easier.
- Enhancing context-aware summarization adds the type_of_mail feature, broadening context comprehension.

4) Sentence-Based Classification Signature Removal:

- Class labels were assigned to eliminate signatures from the initial 100 emails. Each binary value indicates whether a given sentence is a signature or not.
- Footers and greetings were removed from the email body using TFDistilBERT model (in TensorFlow) trained as a binary classifier.

5) Key-Phrase and Keyword Extraction

- Relevant keywords in the cleaned text were attained using TextRank, a graph-based ranking algorithm, by determining patterns of co-occurrence.
- The N-Gram model was applied to merge neighboring keywords into meaningful key phrases.
- The most critical keywords and associated phrases were ranked and filtered using TF-IDF (Term Frequency-Inverse Document Frequency).

6) Text Summarization

- Abstractive summaries were generated using T5 (Text-to-Text Transfer Transformer) model from Hugging Face.
- Using extractive techniques, the system selects existing sentences and places them into a summary, but with T5, short contextually accurate summaries are generated instead.

7) Integration and System Development

- All modules including preprocessing, extraction, and summarization were merged into a single pipeline.
- Users are able to submit email content, and these keywords and summaries are displayed through a web application. The application was made using Flask.

8) Evaluation

- Before making judgments on the system, we first extracted keywords, generated summaries, and evaluated their respective precision and coherence.
- Because there is no definitive benchmark for large email datasets, inspection and case studies were used.

2.3 Challenges

1) **PST Email Rich Layout**: ENRON files are contained in PST (Personal Storage Table) format which is untidy and non-human readable. Conversion to usable CSV data without corruption of data integrity required parsers like Libratom and a fair amount of manual effort.
2) **Disorganized and Noisy Data:** Blends of footers, reply signatures, used lists, and forwards threads all contribute to the overflowing clutter that make up emails. Data cleaning without any loss of information was the main challenge.
3) **Signature and Footer Recognition**: A signature detection model needs training on labelled data which requires building a sentence classifier on TFDistilBERT. Annotation is resource-intensive and yet indispensable for model design and evaluation.
4) **Accuracy of Keyword Extraction:** Informal emails have the tendency to be repetitive, and so the generally indiscrete TextRank falls prey to endorsing unimportant duplicates.
5) **Deficiencies of Abstractive Summarization:** Though T5 excels at performing tasks, sometimes in low contextual or non-datum accompanied input data, overgeneralized and outdated summaries arise.
6) **Evaluation Criteria:** There is a distinct lack of a sole measurement for qualitatively analysing keyword selection.

2.4 Gaps to be Addressed

1) **Absence of Common Datasets for Summarization of Email**: Most summarization datasets like CNN/DailyMail consist of news articles. ENRON is not summarized or keyword labelled, and it is hard to judge models impartially or use supervised learning.
2) **Limited Language Support and Diversity:** The models are given training on English emails only. Multilingual summarization or code-mixed content, which is common in multinational corporate settings, is not yet supported by the system.
3) **Semantic Relevance and Contextualization:** Algorithmic extraction-identified key phrases such as TextRank do not have semantic meaning. Any context-aware key phrase extraction based on deep learning such as BERT embeddings or PEGASUS could counteract this but was not thoroughly explored.
4) **No Personalization or Priority Ranking:** The system does not take into account the role, preference, or history of the user in prioritizing email or in personalizing the summary.
5) **Absence of Robust Evaluation Framework:** A robust evaluation framework including Precision, Recall, F1 Score, and human views can be developed for more seriously assessing the summarization and keyword extraction

3. Existing System

Systemic email systems nowadays don't offer automatic bulk email processing, no smart summarization, and no significant NLP advancements. Your system addresses these gaps with state-of-the-art transformer models, advanced preprocessing, and user-centric design.

• **Reading Manually and Prioritizing**: The majority of office staff read and screen manually hundreds of emails every day in order to manually retrieve valuable information. It is time-consuming, wasteful, and causes responses to be delayed or missed.

•**No Embedded Keyword or Summarization:** Earlier email clients such as Gmail and Outlook lack embedded summarization or keyword extraction. The recipients have to open every email to know what the email is about, even the irrelevant ones.

• **No Preprocessing for Email-Specific Noise:**

The existing tools do not support email-specific artifacts such as

o Threaded answers

o Quoted material

o Footers and signatures : Taking attention away from the body and breaking keyword consistency.

•**Insufficient Intelligence to Filter Content:** Current systems employ basic filters (e.g., sender, subject keywords) and do not employ context-aware models like BERT or T5 to identify the actual meaning or implication of content.

Semantic Unawareness: Email software keyword search does not take into account synonyms, entities, or word co-occurrences. Individuals might not see emails because different words were used.

•**Not for Summarization**: Email programs are not for summarization but for communication. Email programs are not intended to produce abstractive summaries, which would summarize long email conversations into a few informative sentences.

•**Lack of Adaptation to User Context**: Existing systems are incapable of learning from user experience or adjusting to specific priorities, company operations, or visited topics on a regular basis making the experience general and less valuable

Disadvantages of Existing Systems

1) **High Computational Requirements:** BERT and T5 are computationally intensive models and might be too expensive to utilize in real time on low-power equipment or even without full cloud facilities**.**
2) **English Data Dependency:** The model is primarily trained using English emails and is not equipped to process multilingual or code-mixed messages that are ubiquitous in global communication.
3) **No Ground Truth Available for Evaluation:** Since there are no keyword or keyword summary annotations available in the ENRON corpus, the quality of the output is being tested subjectively and without standard measures.
4) **Risk of Incomplete Summaries:** Abstractive summarization will produce misleading or false summaries when the input sentence does not have context or is extremely short, leading to misinterpretation of email messages.
5) **Limited Personalization:** The system handles messages uniformly and does not tailor them based on job titles, preferences, or email priority levels, which affects the quality of summaries.

6) **Privacy and Legal Concerns:** Using it in real business emails raises privacy and security concerns, especially for sectors with legal obligations like finance or healthcare.

4. Conclusion

Summarization and key-phrase extraction are two complementary techniques which, given a natural language text, extract the most important sentences and the most important words or phrases. Therefore, when applied to an email or an email thread, it is expected that these two procedures, when suitably implemented, would give us the most important sentences and words/phrases contained in them, thereby effectively giving us a "snippet" of the text and mostly reducing the time needed to read an entire email. Once a user reads the snippet, they can either choose to read the complete email/email thread, or they can choose to read it later. This is similar to the current techniques for Web search, where we read the snippets to determine the purported "value/relevance" of a particular web page rather than going inside each and every web page to determine its value. This technology will make the task of processing email more efficient, by assisting people prioritize email and by giving people a choice between reading an email right away, removing it, or postponing it for future. Several datasets have been released for general-purpose summarization and keyword extraction, but very few of them deal specifically with emails.

4.1 Future Scope

These approaches and models were built using the email data of "Enron" email dataset. As part of future work, a better and much more precise models could be generated using different kinds of data like Wikipedia data, documents, long paragraphs which are much cleaner. In this research the primary aim was to count only precision value based on the context of our research problem and this value was used for the answering the research question. As part of future work, either recall or f1 measure can also be focused and used to answer different contexts of research questions. As it is known that, this email dataset used in this research is a massive data with over 5,00,000 email records.

References

[1] Abuobieda A., Salim N., Albaham A.T., Osman A.H., Kumar Y.J. (2012). Text summarization features selection method using pseudo genetic-based model. International Conference on Information Retrieval Knowledge Management, pp. 193–197.

[2] Aliguliyev R.M. (2009). A new sentence similarity measure and sentence-based extractive technique for automatic text summarization. Expert Systems with Applications, 36(4): 7764–7772.

[3] Abilhoa, W.D., & de Castro, L.N. (2014). A keyword extraction method from Twitter messages represented as graphs. Applied Mathematics and Computation, 240, 308–325.

[4] Augenstein, I., Das, M., Riedel, S., Vikraman, L., & McCallum, A. (2017). SemEval 2017 Task 10: ScienceIE - Extracting keyphrases and relations from scientific publications. In Proceedings of SemEval, Vancouver, Canada, pp. 546–555.

[5] Hernández-Castañeda Á., García-Hernández R.A., Ledeneva Y., & Millán-Hernández C.E. (2020). Extractive automatic text summarization based on lexical-semantic keywords. IEEE Access, vol. 8, pp. 49896–49907.

[6] Bharti S.K., & Babu K.S. (2017). Automatic keyword extraction for text summarization: A survey. arXiv preprint arXiv:1704.03242.

[7] Zhang M., Li X., Yue S., & Yang L. (2020). An empirical study of TextRank for keyword extraction. IEEE Access, vol. 8, pp. 178849–178858.

[8] Thushara M.G., Mownika T., & Mangamuru R. (2019). A Comparative Study on different Keyword Extraction Algorithms. ICCMC.

[9] Keyword Extraction Based on Semantic Similarity Metric and Multi-Feature Computing. (2020).

[10] Enhancing unsupervised neural networks based text summarization with word embedding and ensemble learning. (2019).

[11] EmailSum: Abstractive Email Thread Summarization using BERT. (2021).

[12] Turney, P.D. (2000). Learning Algorithms for Keyphrase Extraction. Information Retrieval, 2(4), 303–336.

[13] Ulrich, J., Murray, G., & Carenini, G. (2008). A publicly available annotated corpus for supervised email summarization. In Proc. of AAAI Workshop.

[14] Mihalcea, R., & Tarau, P. (2004). TextRank: Bringing Order into Texts. In Proceedings of EMNLP, pp. 404–411.

[15] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In NAACL-HLT

[16] Raffel, C., Shazeer, N., Roberts, A., et al. (2020). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. JMLR.

[17] Pennington, J., Socher, R., & Manning, C.D. (2014). GloVe: Global Vectors for Word Representation. EMNLP.

[18] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. arXiv preprint arXiv:1301.3781.

[19] Page, L., Brin, S., Motwani, R., & Winograd, T. (1999). The PageRank citation ranking: Bringing order to the web. Technical Report, Stanford InfoLab.

[20] Kleinberg, J. (1999). Authoritative Sources in a Hyperlinked Environment. Journal of the ACM, 46(5), 604–632