Machine Learning of Significant Terms

Group Epsilon, SCC - 460

Group Members and Contributions:

1. Rivyesch Ranjan, 36330520 (120%)

   Data pre-processing, Presentation, Report (Section 1, Section 2 and Section 3)

2. Adam Matthews, 36282135 (120%)

   Data pre-processing, Model Implementation, Result Findings and Visualisation, Presentation, Report (Section 3)

3. Isha Patil, 36369395 (120%)

   Report (Section 2.1, Section 2.2 and Section 4), Report Formatting

4. Srirama Subrahmanyasai Moccapati, 36191658 (115%)

   Report (Section 1 and Section 2.1)

5. Junlin Lei, 36332146 (115%)

   Data pre-processing, Bug Fixing

6. Peirui Wang, 36280790 (10%)

   N/A

Project Owner: Informed Solutions

Main Contact Name: Tom Weeks

Main Contact Email: tom.weeks@informed.com

1. INTRODUCTION

Informed solution (IS) is a provider of digital transformation products and services. The digital solutions by IS delivers insights which leads to faster, better decisions. The solutions they provide help clients solve complex business problems across multiple sectors such as healthcare, sustainable energy and environment.

The research project involved identifying non-English terms that are significant from unstructured data and providing a comparison of some effective natural language processing (NLP) algorithms that could be used for this purpose.

The motivation behind this project is that unstructured data in documents can be sorted or categorized by identifying the significant terms within each particular document. However, in cases such as planning and sustainable land management, many of the documents involve both English terms and non-English terms. These non-English terms could be business specific terms, abbreviations of commonly used words, etc. Examples of non-English terms include OPEX which stands for operating expenditure and P60 which a tax form in the UK. While neither of these words are part of the English dictionary, they cannot be simply ignored as they may be significantly important in a particular document. This information can be used to automate labour intensive manual business processes, make faster, better-quality evidence-based decisions, and identify new and better business insights from data.

Our research team was provided data, consisting of numerous pdf files pertaining to applications to the Trafford Council. These applications were in relation to alterations to site layouts, internal refurbishment, and alterations, extensions and demolitions of houses and buildings.

NLP is a branch of artificial intelligence that can be used to identify significant terms and phrases in unstructured data. This is commonly referred to as "term extraction" or "keyword extraction." Appropriate keywords can serve as a highly concise summary of a document which could be used to easily organize documents and retrieve them based on their content [1].

The research objective was to conduct a comparison of NLP techniques to find the significance of non-English terms. This is centred around four key research questions: (1) What NLP techniques can be used to find significant terms? (2) Can a technique be used for terms not in a dictionary? Can they be adapted? (3) What are the advantages and disadvantages of each short-listed technique? (4) What is the performance of each technique on an unstructured dataset?

The remainder of this paper is structured in the following manner. Section 2 covers the pre-processing steps performed on the raw text data to remove noise and to allow for better processing and analysis. The theory and underlying principles of the three NLP techniques, namely Bag of Words, TextRank and Latent Dirichlet Allocation (LDA), used on the dataset are also presented in this section. Section 3 provides a summary of results and discussion. Finally, Section 4 closes the paper with the conclusion, future improvements and challenges encountered.

2. METHODOLOGY

2.1 . Pre-Processing

Pre-processing in NLP refers to a set of procedures used to clean, convert, and prepare text input for further analysis. Some common pre-processing steps include tokenization, stemming, lowercasing, and removal of stop words. The foundation for pre-processing is a corpus, which is a collection of texts. The effective use of these methods helps reduce noise and complexity from the main corpus of text and enables the NLP techniques applied on the text corpus to generate reliable results.

2.1.1. Extracting Raw Data

The pdf files from the dataset are unstructured data since it contains of a mixture of both text and images. Only text from each pdf was imported into Python. This was done by looping through each page of the document and extracting out the text within each page as a separate string. These strings are later combined such that there is one large string for a given document. This process is repeated separately for every document in the dataset directory. The text from the string is stored in a dictionary, where each key stores the document ID, and the value of the dictionary stores the raw text of that document. Thus, the final raw data extraction consists of a dictionary and a main word corpus list that had words from the complete dataset.

2.1.2. Finding All Terms

Tokenisation is applied here to split the raw text into individual words by spaces and sentence ends. Processing individual words instead of sentences allows for easier cleaning of the text, where: regex; removing numbers; and removing symbols are all made easier by only dealing with a single token, rather than sentences or paragraphs.

The presence of both uppercase and lowercase versions of the same word can cause noticeable amount of inconsistency during the implementation of the NLP techniques. When an NLP technique encounters two words, where capitalization is the only difference between them, it treats them as two distinct observations. The conversion of all characters to lowercase is a vital step in the NLP pre-processing procedure. This stage assures the homogeneity of the text data and promotes word matching, which makes the data easier to handle and analyse.

Punctuation and numbers often do not contribute much meaningful insight while generating a high priority list of keywords. This is because the pre-processing is done with the

intention to obtain a corpus or list of words, and not sentences in their full form. Hence, all standalone punctuation was removed from the corpus. However, in some instances, the punctuation does carry meaning for an individual word, such as hyphens. Thus, the hyphens in bigrams are not removed as they might refer to two terms joined together that can be interpreted as one single term.

2.1.3. Data Cleaning

Bigrams refer to the pairs of adjacent words present in the text. The inclusion of these words in the corpus depends on the problem statement that has been presented. In the context of this project, the removal of bigrams depends on whether the two words separated by a hyphen belong to the English language or not. If both words are English, then the bigram is split into two separate entries. If not, that is, if one of the two words are non-English, then the bigram is not split in two as this is a non-English term that may be of significance. The reason behind splitting bigrams that are considered English terms is that Python English dictionary libraries do not contain a vast majority of bigrams in them. Hence, these English bigrams will be treated as non-English terms due to there not being a match when compared to the dictionaries in subsequent pre-processing steps.

Lemmatization is a significant NLP pre-processing step [5]. This is the process of reducing a word to its root form. This is done so that variants of the same word are treated as just the root word and not multiple different words. Converting 'trees' into 'tree' is an example of this step. Lemmatization is essential since Python English dictionary libraries mainly contain only the root words and not words with prefixes and/or suffixes. Since this project is concerned with identifying significant non-English terms, this step ensures that the English words can be filtered out in subsequent pre-processing steps.

There are words that occur multiple times in a document though they have low semantic value [6]. These words are known as stop words and can be eliminated without altering the text's comprehension. Some common stop words are 'the', 'and', 'is'. Eliminating stop words can reduce the size of the corpus and enhance the analysis's effectiveness. This stage is especially beneficial for activities such as text summarization when it is essential to focus on the text's most significant terms.

The pre-processing steps done up till this stage removes extraneous information, lowers the dimensionality of the corpus and appropriately formats the text. These refine the performance of the NLP techniques and makes subsequent analysis more accurate and efficient.

2.1.4. Finding Non-English Terms

To identify non-English terms in the pre-processed corpus, each word in the corpus needs to be compared or checked with an English dictionary. If that particular word is in the English dictionary, then that word is deemed an English term whereas if it cannot be found then it is a non-English term. The fact that none of the Python English dictionaries are complete is a challenge as this would mean many English terms would be wrongly recognised as a non-English term. To overcome this problem, a larger, more comprehensive dictionary is created by taking the union of all words in the 'nltk' and 'english_words' dictionaries. These are two Python English dictionary libraries that contain a large majority of the English words, albeit in its root form. Nevertheless, some English words will still be mistaken for non-English terms due to regional or archaic spelling. This step separates the corpus into two, one containing all terms and the other containing only non-English terms.

2.2. Techniques

To obtain a list of high priority non-English words found in the PDF files, several NLP techniques were implemented. The main techniques used are – Bag of Words, LDA and Text Rank. These techniques were each applied on the pre-processed corpus and a list of the top 20 high priority keywords was extracted.

Bag of words is used to analyse text by overlooking its grammar and structure but considering the recurrence count of every word. LDA follows the laws of topic modelling and prioritises words based on the probability of their occurrence throughout all documents. On the contrary, TextRank aims to create a weighted graph using words as nodes and deducing similarity scores by calculating the weights of the links between the nodes.

2.2.1. Bag of Words

Bag of words is used to understand the structure of the text better. It does this through the process of conversion of words into vectors. This technique does not understand the sentiment of the word or sentence that is being processed. The sentiment of a word or sentence can be determined by analysing if its conduct is positive or negative. Bag of words associates text with a vector indicating the number of occurrences of each chosen word in the training corpus [7].

The list of non-English words from each document that had been extracted were vectorized, where each word was marked according to the frequency of its occurrence in all documents. Every document had its own list of non-English words that had been extracted while pre-processing. Every list of each document was considered as a separate document input to the technique. Thus, each document that was pre-processed by the bag of words technique had its own vector representation. This led to the formation of a matrix that conveniently conveyed whether a particular non-English word occurred in a document. Based on the above matrix, a list of keywords with the highest number of occurrence count was considered as significant and of high priority.

2.2.2. Latent Dirichlet Allocation (LDA)

Sentences can be classified into several topics. Every document is considered as an aggregate of topics and these

topics can be seen as an aggregate of keywords with their corresponding coherence score.

Every word that belongs to a document is represented by the probability of it belonging to a particular topic. In other words, word probabilities can be viewed across the *K* topics as a *K* -dimensional vector representation for each word [8]. To calculate this, each word in the document is assigned to one of the topics that have already been generated. The probability that a word *w* belongs to a topic *t* is calculated as follows:

$$p(\text{word belongs to topic } t) = p(\text{topic } t \mid \text{document } d) * p(\text{word } w \mid \text{topic } t)$$

In the above equation, the probability that a word belongs to a particular topic is expressed as the multiple of the probability of a document being under a topic *t* and probability of a topic *t* through all documents belonging to a particular word. So, it is assumed that if a word has a good count of appearance in one document that belongs to a topic *t*, then that word might be a part of *t.*

The inputs taken in by LDA include words and their corresponding frequency. This eradicates the use or presence of sentimentality in the document or sentence under consideration. This makes the model perfect to evaluate non-English words that are extracted from every document. The number of topics that need to be generated needs to be decided beforehand and this influences the accuracy of the results produced.

The initial inputs to the program are the list of documents, where each document is a list of non-English words, the frequency of each word in a document, and the number of topics that need to be extracted from a given set of documents. Using the above information, a list of top n keywords is generated with their respective coherence score. A higher coherence score is used to determine the interpretability of a particular topic or keyword.

Each document is converted into a dictionary to represent the list of non-English words as a bag of words. This conversion into a dictionary is done using Gensim. When a dictionary using Gensim is created, it creates a bag of words corpus which can be further used to model or generate topics from the document inputs.

The list of top keywords and their corresponding probability or coherence scores are generated by first retrieving all topics from the model along with the specified number of keywords. Then the document is divided into several keywords by first cutting off all keywords that have a probability score less than 0. Further, the average coherence of each keyword is calculated. This is done by summating the coherence scores for each word and then averaging them. This list is sorted by average coherence and returned along with the count data.

2.2.3. TextRank

TextRank is a technique that can be used to rank keywords present in the form of text. This technique is an adaptation of a method used to rank various pages called “PageRank”, where each web page is represented as a node in a graph. If two nodes are connected by a directed edge, then the corresponding pages are linked. Once a graph is constructed, each node (page) is assigned with a weight. TextRank uses this concept to build weighted graphs where every word can be represented as a node. It identifies connections between various entities in a text and implements the concept of recommendation [9].

The approach followed by TextRank after pre-processing can be explained in three phases – constructing a graph, computing scores, and selecting relevant keywords. A weighted graph is constructed, where each word is treated as a node to establish relationships between them. Using this, the PageRank algorithm assigns each node with a score based on the number of links associated with it. So, the nodes (words) with the highest scores are considered significant.

The structure of the algorithm depends on the raw list of non-English words and the number of keywords that need to be generated.

Generally, the text from which keywords are to be extracted is broken down into segments of sentences using spacy. These segments are further broken down into separate words that are unique throughout. The non-English words in this case that had already been tokenized while pre-processing, are used as nodes for the construction of the weighted graphs.

The weights of the words that are represented as nodes in the graph are deduced by the relationship links between them. For each word, its similarity with every other word can be explained using the weights between them. This is done using a window-based approach. The similarity is computed by setting a fixed window size for each word. This similarity score between each word is used as the weight between two nodes.

On calculating the weights, a weighted graph is constructed that represents a snap of the words and their relationship. From the constructed graph, a score is generated for each node by the PageRank algorithm. This score represents the importance of every node, after considering its relationship with every other node. The non-English words that are assigned by the highest scores are extracted as the top or high priority keywords.

3. RESULTS

Whilst each technique had its own unique way of finding the top keywords and ranking them in order of importance, the main metrics that provide insights to the project owner are the count, number of documents and percentage of documents. The count simply refers to the number of times a term appeared in the corpus. Meanwhile, the number of documents indicates if the term occurred over a spread of documents or only a few

documents. This metric provides context to the significance of a term as if it occurs frequently and over a large number of documents that would mean that term is highly significant. The percentage of documents is similar to the previous metric except it is shown as a percentage.

The results shown in Section 3.1 is relevant to a subset of the data which contained 139 files. These files consist of PDFs, word documents, JPEG images and TIFF images. Of these files, only text from the PDFs were extracted and used in the corpus. Hence, the valid number of documents for this subset of the dataset is 96 files.

3.1. Top Keywords

The top 20 most significant non-English terms determined using the Bag of Words technique is shown in Table 1. It can be easily inferred from the table that the criterion for determining the most significant terms is solely based on the frequency or count of a word appearing in the text corpus. From the table below it is clear the word 'Altrincham' which is a place in Greater Manchester is the most significant term with almost twice the frequency of the next most significant term 'hha' which is a code used by the Trafford Council for house holder applications. Furthermore, 'Altrincham' appears in 63 documents or 66% of the documents used as the dataset. Knowing that the word appears in a large proportion of the documents is a solid indicator that the word is significant, and not one that appears multiple times across just a few documents. A good example of this would be the word 'cil' which is the fourth most significant term appearing 199 times across the corpus. However, all these occurrences of 'cil' were limited to just 7% of the entire dataset which suggest that this word is repeated many times across the few documents that it is found in.

Analysing the top 20 keywords produced by Bag of Words shows that many of the keywords identified are places, postcodes or motorways. The words 'Altrincham', 'Trafford', 'Bowdon', 'Stretford', 'Timperley', 'Davyhulme' and 'Urmston' are all places in Greater Manchester. Postcodes such as 'WA14' and 'WA15' are for Warrington and Altrincham respectively. The motorways 'M33' and 'M41' also show up as keywords.

Beyond that, there are a few notable observations that need to be mentioned. The term '©' which is a symbol for copyright was not successfully filtered out during the pre-processing stage. Whilst numbers, punctuations and many common symbols were removed from the corpus, the filtering process is not entirely perfect and does not remove all symbols, especially less common ones. The terms 'centre' and 'mm' are wrongly classified as non-English terms. The dictionaries used to compare the corpus to contained only the American vocabulary of English words. This explains why 'centre' is considered as a non-English term as it is spelt in British English. In the case of 'mm', the dictionaries did not contain abbreviations. This is a major limitation since there is no dictionary or combination of dictionaries that contain every single word found in the English dictionary. Overall, these three terms are clearly not significant terms, however, all three appears quite frequently in the corpus. The drawback of this technique is that it ignores context by discarding the meaning of the words and focusing on the frequency of occurrence.

Table 1: Bag of Words Top 20 Keywords

Words	Count	No. of Documents	Percentage of Documents
altrincham	472	63	66
hha	238	11	11
centre	209	22	23
cil	199	7	7
wa14	187	43	45
trafford	165	29	30
bowdon	153	22	23
ltd	145	35	36
©	124	11	11
ful	123	20	21
m33	110	9	9
rwp	105	14	15
wa15	101	15	16
mh	86	9	9
timperley	81	16	17
mm	76	11	11
davyhulme	72	9	9
urmston	71	9	9
m41	71	9	9
stretford	66	11	11

Using Gensim LDA the top 20 keywords can be viewed in Table 2. This technique identifies the term 'hha' as the most significant term over 'Altrincham' despite it occurring fewer times. The results show that the keywords found are not dependent on the frequency of the terms. Instead, the ranking of the keywords is based on the coherence score calculated by the algorithm when it performs topic modelling which for the purposes of this project is used synonymous with keyword extraction. It should be noted that while the top keywords have a high coherence score, it does not necessarily appear in a large percentage of the documents.

A closer look at the keywords in the table shows that the list of keywords produced by this technique is significantly different to the results of the previous technique. There are fewer places that show up as keywords, with only 'Altrincham' and 'Bowdon' considered as significant. There are more abbreviations commonly used in construction such as 'C16', 'rwp', 'dwg' and 'upvc'. There are also two application form numbers that make it into the top 20 keywords.

The words 'floorspace' and 'relates' appear in the list as non-English keywords. The first is due to the word being spelt as one word instead of two separate words 'floor' and 'space'. As a result, it is not identified as an English word rightfully so. On the other hand, 'relates' was wrongly recognised as a non-English word. This can be attributed to the lemmatisation pre-processing step not reducing the word to its root form 'relate'.

Table 2: Gensim LDA Top 20 Keywords

Words	Count	No. of Documents	Percentage of Documents
hha	238	11	11
altrincham	472	63	66
regs	20	10	10
dwg	26	19	20
cil	199	7	7
floorspace	31	9	9
wa14	187	43	45
pp-11471240	8	2	2
relates	47	12	12
upvc	43	16	17
rwp	105	14	15
e3	13	9	9
mh	86	9	9
ltd	145	35	36
bowdon	153	22	23
lcd	20	1	1
centre	209	22	23
m2	33	4	4
pp-11298341	10	1	1
c16	11	1	1

The TextRank technique results are displayed in table 3. The list of keywords is similar to the list of keywords identified by Bag of Words. 'Altrincham' is the top keyword and there are numerous places, postcodes, and motorways in the list. Unlike Bag of Words, the top keywords are not ordered by the count. It can also be seen that the top keywords appear in a high percentage of documents.

Similar to the other techniques some terms that refer to units such as 'm^{2}' and 'mm' are considered as key terms although these terms provide very little information about the data in the pdf files. Some abbreviations such as 'cil' and 'ppe' which stands for community infrastructure levy and personal protective equipment respectively were also a part of the top 20 list.

Table 3: TextRank Top 20 Keywords

Words	Count	No. of Documents	Percentage of Documents
altrincham	472	63	66
mm	76	11	11
trafford	165	29	30
wa14	187	43	45
bowdon	153	22	23
ltd	145	35	36
m33	110	9	9
groby	59	15	16
cil	199	7	7
timperley	81	16	17
wa15	101	15	16
urmston	71	9	9
mh	86	9	9
hons	19	11	11
ppe	16	2	2
rooflight	29	6	6
stretford	66	11	11
m41	71	9	9
davyhulme	72	9	9
m2	33	4	4

3.2. Influence of Input Documents on Mean Document Coverage

Figure 1 below displays the mean document coverage of the top 20 keywords produced by each of the three techniques used for varying number of input documents. The graph reveals that the TextRank technique performs the best for cases when less than 30 documents are provided as input. When more documents are used the Bag of Words technique slightly outperforms the TextRank algorithm. For the results shown in Section 3.1, the mean document coverage is 19.4, 16.05, 17.30 for Bag of Words, LDA and TextRank respectively. It is also clear that the Bag of Words and TextRank techniques are superior to LDA with respect to the objectives of this project. The poor performance of LDA could be potentially due to it being a model meant specifically for topic modelling and not keyword extraction.

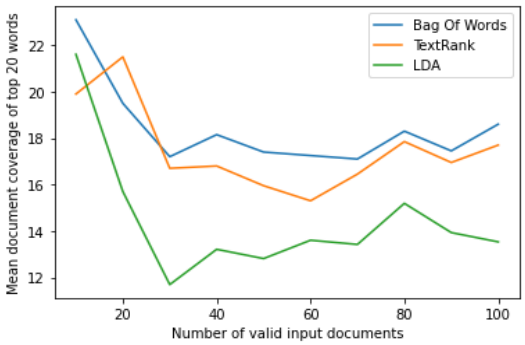

Figure 1: Comparison of all three Techniques' Mean Document Coverage vs Number of Input Documents

3.3. Influence of Input Documents on Execution Time

Figure 2 below depicts the time taken to execute each technique for varying number of input documents. From the figure below it is evident that the TextRank technique takes a significantly longer time to produce the top keywords compared to the LDA technique, especially for a larger input of documents. As the number of input documents to the model increases, the time taken to execute TextRank grows exponentially. On the other hand, there is very little influence on the LDA model. This infers that the LDA model is more efficient and suitable when attempting to generate the top keywords from an extremely large collection of documents. Bag Of Words is not included here, as it is done as part of pre-processing. Thus, would be 'instant' compared to the other two techniques.

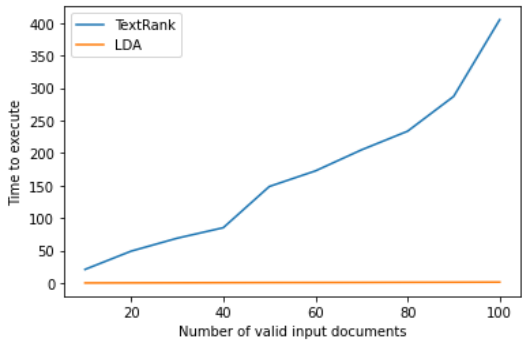

Figure 2: Comparison of Execution Time vs Number of Input Documents

3.4. Validity

The data used for the project consisted of unstructured pdf files, which led to the formation of a word corpus on which the techniques were applied. Thus, the structure of the dataset liquidates the comprehension of internal and external validity in the data. There are no independent and dependent variables that can be used to find causal relationships between them. The metric to validate the keywords is suggested by IS. There is no evidence to support the statement that the percentage of occurrence of top-priority non-English words through all documents validates their significance. This fails the construct validity test which suggests that the test or metric used to demonstrate the validity must exist.

4. CONCLUSION

While LDA is successfully able to extract meaningful keywords better than Bag of Words and TextRank, its performance sees a steep decline in the coverage of documents, as the number of inputs is increased. However, after looking at the time taken to execute LDA and TextRank, LDA is more efficient, while TextRank takes longer to execute as the number of input documents increase. The performance of both Bag of Words and TextRank can be concluded as best when compared to LDA. This is due to the fact their mean document coverage is significantly higher than LDA. However, for large datasets, Bag of words performs better than the other two in terms of computational time and performance.

The project covers the extraction of non-English words of significance and the percentage of their occurrence through all documents or text files. The scope of this project can be expanded by identifying the topics that these words belong to and further classifying them into business sectors. This will allow the non-English words to be categorized based on their importance in a particular field. The classification can be done using topic modelling techniques, where the generated topics can be classified. This can also be done by building up on the LDA model that has been implemented to generate a list of high priority non-English words.

The main challenges incurred while implementing this project revolved around validity and NLP being a topic that is not included in the curriculum of this course. Once the lists from all three techniques had been generated, there was no method, metric, or test to validate the non-English words. Consequently, IS suggested that the models must generate a percentage of occurrence score which would help them understand the importance and significance of every word that had been prioritized. NLP is a data science topic that was new to all the members who had been assigned with this project. The initial task of understanding the concepts and researching and analysing all suitable techniques before starting the implementation can be considered as one of the challenges that were faced.

ACKNOWLEDGEMENTS

1. https://towardsdatascience.com/latent-dirichlet-allocation-lda-9d1cd064ffa2
2. https://www.tutorialspoint.com/gensim/gensim_creating_a_dictionary.htm
3. https://www.machinelearningplus.com/nlp/topic-modeling-gensim-python/#20topicdistributionacrossdocuments
4. https://towardsdatascience.com/evaluate-topic-model-in-python-latent-dirichlet-allocation-lda-7d57484bb5d0
5. https://www.machinelearningplus.com/nlp/topic-modeling-gensim-python/
6. https://towardsdatascience.com/textrank-for-keyword-extraction-by-python-c0bae21bcec0
7. https://www.simplypsychology.org/validity.html

REFERENCES

[1] Wael Etaiwi, Ghazi Naymat, The Impact of applying Different Preprocessing Steps on Review Spam Detection, Procedia Computer Science, Volume 113, 2017, Pages 273-279, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2017.08.368.

[2] Pilehvar, M. T. (2017). On the Role of Text Preprocessing in Neural Network Architectures: An Evaluation Study on Text Categorization and Sentiment Analysis. *ArXiv*. https://doi.org/10.48550/arXiv.1707.01780

[3] F. S. Gharehchopogh and Z. A. Khalifelu, "Analysis and evaluation of unstructured data: text mining versus natural language processing," 2011 5th International Conference on Application of Information and Communication Technologies (AICT), Baku, Azerbaijan, 2011, pp. 1-4, doi: 10.1109/ICAICT.2011.6111017.

[4] Maslej-Krešňáková V, Sarnovský M, Butka P, Machová K. Comparison of deep learning models and various text pre-processing techniques for the toxic comments classification. Applied Sciences. 2020 Dec 2;10(23):8631.

[5] K. Smelyakov, D. Karachevtsev, D. Kulemza, Y. Samoilenko, O. Patlan and A. Chupryna, "Effectiveness of Preprocessing Algorithms for Natural Language Processing Applications," 2020 IEEE International Conference on Problems of Infocommunications. Science and Technology (PIC S&T), Kharkiv, Ukraine, 2020, pp. 187-191, doi: 10.1109/PICST51311.2020.9467919.

[6] K. S. Dar, A. B. Shafat and M. U. Hassan, "An efficient stop word elimination algorithm for Urdu language," 2017 14th International Conference on Electrical Engineering/Electronics, Computer,

Telecommunications and Information Technology (ECTI-CON), Phuket, Thailand, 2017, pp. 911-914, doi: 10.1109/ECTICon.2017.8096386.

[7] HaCohen-Kerner, Y., Miller, D., & Yigal, Y. (2020). The influence of preprocessing on text classification using a bag-of-words representation. *PLOS ONE*, *15*(5), e0232525. https://doi.org/10.1371/journal.pone.0232525

[8] Cai, Zhiqiang; Li, Hiyiang; Hu, Xiangen; Graesser, Art Grantee Submission, Paper presented at the International Conference on Educational Data Mining (9th, Raleigh, NC, Jun 29-Jul 2, 2016, (p577-578))

[9] Mihalcea, Rada, and Paul Tarau. "Textrank: Bringing order into text." In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pp. 404-411. 2004.