DESCRIPTION OF THE MIKE$_2$ ALGORITHM FOR PRESENTATION MINING

Vinothini Kasinathan[1] and Aida Mustapha[1,2]
[1]Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, UPM Serdang, Selangor Darul Ehsan, Malaysia
[2]Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, Malaysia
E-Mail: vinothini@apu.edu.my

ABSTRACT

This paper describes a keyphrase extraction algorithm in Presentation Mining called MiKe$_2$. The algorithm extracts keyphrases and keywords from a collection of presentation slides to be generated into a visual knowledge display looks like a mind map. MiKe$_2$ takes a statistical approach by combining the n-grams frequency count and weight from the C-Value approach. The algorithm is hoped to improve performance in Presentation Mining by automatically generating a high quality mind map that could improve teaching and learning in general.

Keywords: presentation mining, keyphrase, mining keyphrase.

INTRODUCTION

Slide presentationssuch as the PowerPoint are usually prepared by the subjectmatter expert or packaged as a book companion in a linear sequence (Kinchin *et al.*, 2008). Theintegrated knowledge structure of the subject matter expert is transformed intopresentation slides, which are in linear sequence. Nonetheless, post-presentation,the knowledge is actually reconstructed differently by the learners dependingon their understanding in Figure-1. Due to this,Kinchin (2009) proposed aconcept mapping to help learners visualize the content hence shifting the focus from linear structure to network of expert knowledge.

Presentation Mining (Kasinathan and Mustapha, 2015) is an approach to reduce the misinterpretation betweenthe original expert structure (original contents of the slides) by the instructorand the di erent audience (learners). The approach is to automatically generatea visual knowledge display such as mind map based on important keywords and key phrases extracted from the presentation slides. The objective of this paper is to introduce a new keyphrases extraction algorithm called MiKe$_2$ that willfurther improve the quality of mind maps produced.

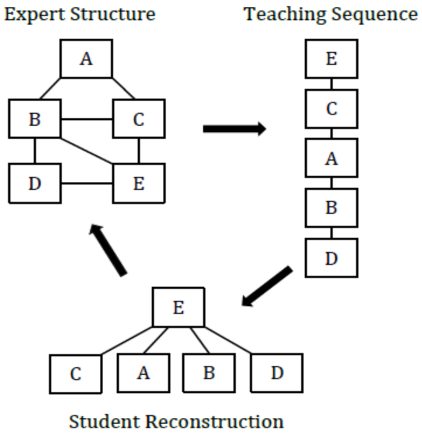


Figure-1. Original expert structure vs. student reconstruction (Kinchin *et al.*, 2008)

The remainder of this paper is organized as follows. Related work presents theworks related to keyphrase extractions algorithms. Description of MiKe2 Algorithm describes the pro-posed MiKe$_2$ algorithm followed by its application to Presentation Mining. Finally concludes with some direction for future works.

RELATED WORK

Keyphrases are sequence of words that provide a brief abstraction on the document content (Witten *et al.*, 1999), (El-Beltagy, 2006), (Kumar *et al.*, 2008). The number of words in a phrase usually ranges from one to three, while the sequence of four words in a phrase is usually rare, unless in a specic domain such as medicine. Research has shown that manually as signing keyphrases are costly and time consuming due to the demand of domainspecialist, which are individuals who have to read through all the content inorder to look up for the keyphrases (Lim *et al.,* 2013). While the task has now become nearly impossible to achieve due to present situation of document overloading, automatic keyphrase extraction systems are highly required. Automatic keyphraseextraction is far more cost effective, and the process of identifying key phrasesfound within a document that are most likely to be assigned by a human (El-Beltagy, 2006).

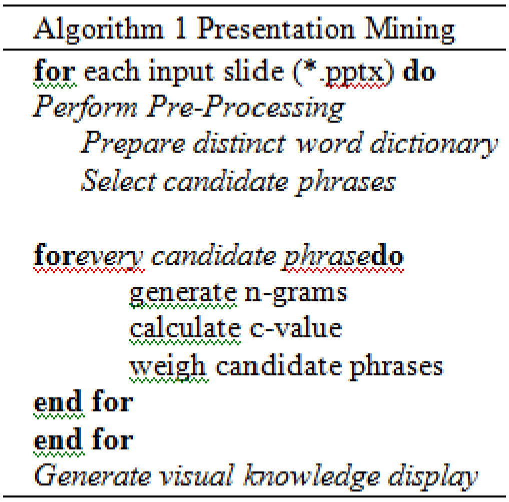
Implementation of automatic keyphrase extraction can be broadly categorized into two approaches; learning and non-learning (Kumar *et al.*, 2008). Among well-knownkeyphrase extraction systems are GenEx (Turney, 2000), KEA (Witten *et al.*, 1999) and KP-Miner (El-Beltagy, 2006). GenExand KEA both treated the task of extracting keyphrases as a supervised learning approach where training documents with known keyphrases are trained inorder to build a model for identifying the probabilities of identied candidatekeyphrases to be a keyphrase. KP-Miner, on the other hand, uses a non-learningapproach, whereby no training documents are required in order to identifykeyphrases within a given document (Lim *et al.*, 2013).

The main difference among the extraction algorithms lies in the calculation of weightage for each candidate phrases produced. GenEx uses the TF, positionof first occurrence, and number of words in a phrase. KEA uses TF-IDF and position of first occurrence. Meanwhile, KP-Miner uses TF-IDF, position of first occurrence, and two boosting factors which boost the weight of longer phrases, aswell as phrases which occur earlier. KP-Miner also proposed an N-gram filtration apporach that uses TF and position of firrst occurrence. KP-Miner outperformed

GenEx and KEA in terms of accuracy rate attributed by the N-gram filtration apporach as well as the processing time since no learning is involved.

DESCRIPTION OF MiKe$_2$ ALGORITHM

MiKe$_2$ keyphrase extraction algorithm takes a statistical approach to identify the most accurate and meaningful keyphrases based on the C-value (Frantzi and *et al.*, 2000) for thehighest n-gram generated for each candidate phrase. N-gram is very useful in keyphrase extractions because some words or terms are more probable to follow a word in certain contexts, hence forming a phrase of certain number of words. However, previous research has shown that n-gram is insufficient to differentiate meaningful phrases such as ‘no explicit loop’ vs. ‘explicit loop’. In summary, MiKe$_2$ is shown in Algorithm 1.



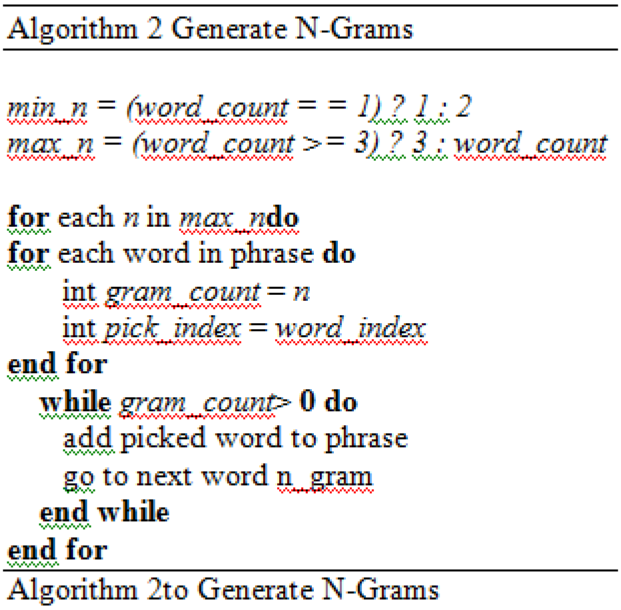
```
Algorithm 1 Presentation Mining
for each input slide (*.pptx) do
Perform Pre-Processing
    Prepare distinct word dictionary
    Select candidate phrases

forevery candidate phrasedo
        generate n-grams
        calculate c-value
        weigh candidate phrases
end for
end for
Generate visual knowledge display
```

From the algorithm, pre-processing in MiKe$_2$ involves standardization, sentence segmentation, tokenization, lemmatization, part-of-speech tagging, words removal, phrase recognition and chunking. During standardization, the collection of input slides will be tranformed into ASCII-English. Single and double quotations as well as hyphens are converted into a readable form. Newlines are replaced with tab and whitespaces are trimmed. Next, in sentence segmentation, the sentences are split into newline, period, exclamation marks and question marks using API. During tokenization, digits and letters in on-alphanumeric characters are separated, hyphens are joined with words, whitespaces and continuous symbol are removed.

MiKe$_2$ uses API to lemmatize tokens into their base forms and refine the results by taking characters after the plus symbol. After lemmatization, API part-of-speech tagging is performed and the results are refined by taking the characters after the underscore symbol. During words removal, symbols, words less than 4 characters (except thos capitalized and tagged with ‘CD’ during POStagging), as well as stop words are removed. Finally, during phrase recognition and chunking, full sentences are sent to perform API chunking and the chunked tags are modified to cross marked tokens to “0”. Based on the chunk tags, tokens are also joined to form a phrase.

Once pre-processing is completed, words are gathered from all slides to form a list of distinct words. Finally, candidate phrases are selected based on the three conditions; the candidate phrase cannot be a substring ot duplicated, it has to be a noun, and it must not contain words with ‘CD’ POS-tag. MiKe$_2$ then uses the candidate’s phrases to generate the n-grams as shown in Algorithm 2.



```
Algorithm 2 Generate N-Grams

min_n = (word_count = = 1) ? 1 : 2
max_n = (word_count >= 3) ? 3 : word_count

for each n in max_ndo
for each word in phrase do
    int gram_count = n
    int pick_index = word_index
end for
  while gram_count> 0 do
    add picked word to phrase
    go to next word n_gram
  end while
end for
Algorithm 2to Generate N-Grams
```

From the set of n-grams generated, MiKe$_2$ will return one n-gram with the highest C-value that is calculated using Equation 1.

$$C - value(a) = \begin{cases} \log_2|a| \cdot f(a)\, if\ a\ is\ not\ nested \\ \log_2|a|(f(a) - \frac{1}{P(T_\alpha)\sum_{b\in T_\alpha} f(b)}\ otherwise \end{cases} \quad (1)$$

where a is the n-gram, f(·) is the frequency of occurrence in slides, T_α is the set of extracted candidate keyphrases that contain a, and $P(T_\alpha)$ is the number of the candidate keyphrases. With the C-values serve as weights to the candidate phrases, the selected phrases will be used in generating a visual knowledge display (i.e. mind map)

APPLICATION TO PRESENTATION MINING

Presentation mining in an approach to extract keyphrases from presentationslides such as PowerPoint and generate a visual knowledge display using theextracted keywords. Figure-2 shows the steps in Presentation Mining which isundertaken in this research. To illustrate the application to MiKe2 in a Presentation Mining approach, a collection of presentation slides for Articial Intelligence course at introductory level across di erent universities worldwide are used as the input slides. The scope will be using the Artificial Intelligence: Modern Approach text book which is written in American English which will be the lexicon used.

After the process of selecting slides and keyphrases, the system will visualize the selected keyphrases into a SmartArt diagram in Microsoft PowerPoint, which is similar to mind map.

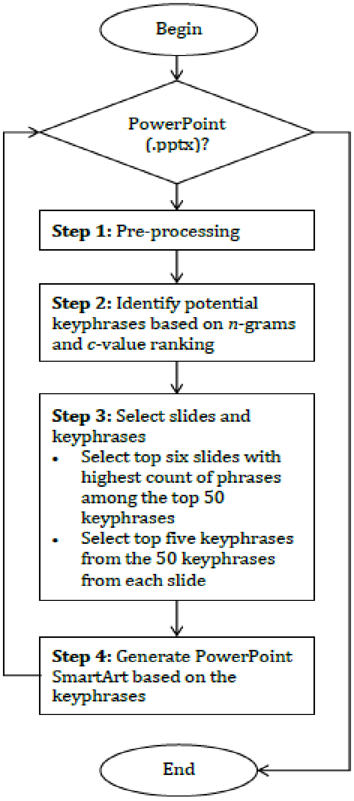

Begin
PowerPoint (.pptx)?
Step 1: Pre-processing
Step 2: Identify potential keyphrases based on n-grams and c-value ranking
Step 3: Select slides and keyphrases
Select top six slides with highest count of phrases among the top 50 keyphrases
Select top five keyphrases from the 50 keyphrases from each slide
Step 4: Generate PowerPoint SmartArt based on the keyphrases
End

Figure-2. Steps in presentation mining.

Output generated mind map using the PowerPoint SmartArt is shown in Figure-3.

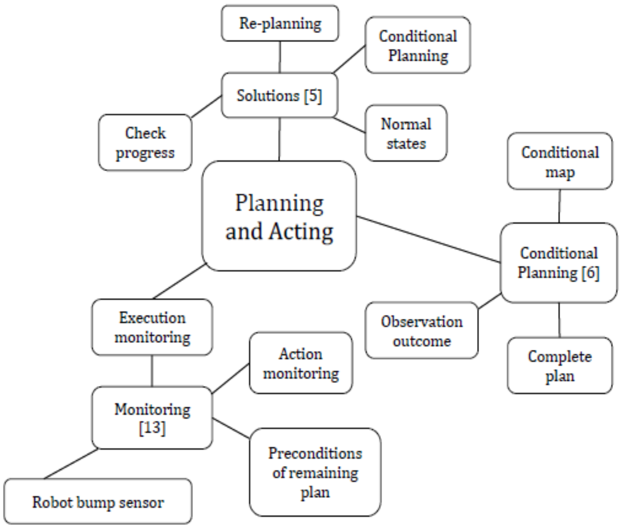


Figure-3. Output of the powerpoint SmartArt.

The list of keyphrases extracted by MiKe$_2$ will then be compared with the output from KP-Miner to see the difference. Table-1 shows the comparison of keyphrases extracted by the KP-Miner and the proposed MiKe$_2$algorithm on Chapter 13 of the Artifcial Intelligence: A Modern Approach textbook (Russel and Norvig, 2003). This Table shows that the MiKe$_2$ algoritm overcomes some issues of KP-Miner as anon-learning algorithm by being able to extract more meaningful keyphrases. MiKe$_2$ has used the strength of C-value algorithm that brings up keyphrase ranking to the top after re-ranking it. It also has the strengths of N-gram which avoids the bias extraction process in which KP-miner uses. Therefore the outputs from MiKe$_2$ are much more meaningful then KP-Miner algorithm.

Table-1. Comparison of keyword and keyphrases between KP-miner and MiKe$_2$.

Slide	KP-miner	MIKE$_2$
0	Planning and Acting	Planning and Acting
5	Solutions	Solutions
	Conditional planning	Observation actions
	Assume normal states	Failure
	Observation actions	Conditional planning
	Check progress	Unanticipated outcomes
	Re-planning	Check progress
4	Things go wrong	Things go wrong
	Incomplete informatiom	Incorrect information
	Unknow preconditions	Unknow preconditions

	Disjunctive effects	Required preconditions
	Incorrect information	Incorrect postconditions
	Qualification problem	Current state
6	Conditional planning	Conditional planning
	Insert conditional step	Observation outcome
	Complete plan	Conditional step
	Observation outcome	Current KB
13	Monitoring	Monitoring
	Executive monitoring	Action monitoring
	Action monitoring	Execution monitoring
	Preconditions of remaining plan	remaining plan
	Robot bump sensor	Robot bump sensor
2	Outline	Outline
	Real world	Real world
15	Re-planning	Re-planning
	No explicit loop	Explicit loop
	Simplest	Best continuation
	Scratch scratch	

CONCLUSIONS

Slide presentations have been widely used in current teaching and learning process. While text-laden slides might give a comprehensive feel over the materials, the slides full of key points are not useful without the presenter (Kasinathan *et al.*, 2013). The objective of Presentation Mining is to improve the teaching and learning process by transforming the slide contents into a visual knowledge display because the main challenge lies in the fact that slides already contains keywords and keyphrases. Visual knowledge display such as the mind map reorganizes the keywords/keyphrasesin the slides from sequential to network-based while keeping the relationships from the slides intact.

This paper presents a new keyphrase extraction algorithm called MiKe$_2$thatcapitalized on the statistical information in the words. MiKe$_2$ was applied to Presentation Mining and its outputs are compared with the output of KP-Miner. Based on the comparisons, MiKe$_2$was at par to KP-Miner with more meaningful keyphrases such as ‘conditional step’ as opposed to ‘insert comditional step’by KP-Miner. In the future, this research will strive to improve the keyphraseextraction algorithm in Presentation Mining approach by considering contextual knowledge within the slides.

ACKNOWLEDGEMENT

This project is sponsored by University Putra Malaysia under Geran Putra-Inisiatif Putra Siswazah (GP/IPS/2013/9399844).

REFERENCES

El-Beltagy, S. R. 2006. KP-Miner: A Simple System for Effective Keyphrase Extraction. In Innovations in Information Technology, 2006 (pp. 1-5). IEEE.

Frantzi, K., Ananiadou, S., and Mima, H. 2000. Automatic recognition of multi-word terms:. The c-value/nc-value method. International Journal on Digital Libraries, 3(2), 115-130.

Kasinathan, V., Mustapha, A. 2015. Ontology Support for Web-based Presentation Mining. In Proceedings of the the Second International Conference on Advanced Data and Information Engineering, 25 - 26 April 2015, Bali, Indonesia.

Kasinathan, V., Mustapha, A., and Rani, M. F. C. A. 2013. Structure-Based Algorithm for Presentation Mapping in Graphical Knowledge Display. International Journal of Information and Education Technology, 3(2), 196.

Kinchin, I. 2009. A knowledge structures perspective on the scholarship of teaching & learning. International Journal for the Scholarship of Teaching and Learning, 3(2), 5.

Kinchin, I. M., Chadha, D. and Kokotailo, P. 2008. Using PowerPoint as a lens to focus on linearity in teaching. Journal of Further and Higher Education, 32(4), 333-346.

Kumar, N., and Srinathan, K. 2008, September. Automatic keyphrase extraction from scientific documents using N-gram filtration technique. In Proceedings of the eighth ACM symposium on Document engineering (pp. 199-208). ACM.

Lim, V. H., Wong, S. F. and Lim, T. M. 2013, April. Automatic keyphrase extraction techniques: A review. In Computers and Informatics (ISCI), 2013 IEEE Symposium on (pp. 196-200). IEEE.

Russel, S., and Norvig, P. Artificial Intelligence: A Modern Approach, 2003. EUA: Prentice Hall.

Turney, P. D. 2000. Learning algorithms for keyphrase extraction. Information Retrieval, 2(4), 303-336.

Witten, I. H., Paynter, G. W., Frank, E., Gutwin, C., and Nevill-Manning, C. G. 1999, August. KEA: Practical automatic keyphrase extraction. In Proceedings of the fourth ACM conference on Digital libraries (pp. 254-255). ACM.