2019

Deploying Natural Language Processing to Extract Key Product Features of Crowdfunding Campaigns: The Case of 3D Printing Technologies on Kickstarter

Nina Chaichi
Portland State University

Tim R. Anderson
Portland State University, tim.anderson@pdx.edu

Citation Details

N. Chaichi and T. Anderson, "Deploying Natural Language Processing to Extract Key Product Features of Crowdfunding Campaigns: The Case of 3D Printing Technologies on Kickstarter," 2019 Portland International Conference on Management of Engineering and Technology (PICMET), Portland, OR, USA, 2019, pp. 1-9.

Deploying Natural Language Processing to Extract Key Product Features of Crowdfunding Campaigns

The Case of 3D Printing Technologies on Kickstarter

Nina Chaichi, Timothy Anderson
Dept. of Engineering and Technology Management, Portland State University, Portland, OR-USA

***Abstract*—In the technology management field, informal source of information such as social media has been used for technology mining, leader user detection, and etc. However, usually unstructured nature of the informal information introduces some challenges. This study is part of an effort to automate key product features extraction from crowdfunding campaign's textual information, in order to analyze the effect of them on the decision-making process of crowdfunding backers. This paper intends to evaluate the performance of UDPipe R package and six keyword extraction techniques on candidate features selection from textual data of 3D printer campaigns on Kickstarter. In the end, it will be discussed which technique captures the 3D printer features better.**

I. Introduction

It is stated that average people are unlikely to support technology projects to some degree, on the other hand, technology savvy people would back the technology project if they can appreciate and value [1]. In other words, technology product features have a strong impact on the decision of crowdfunding backers whether to support the project and consequently on the success of the project to raise their desired fund. However, to test the relation between the product features and success of crowdfunding projects, the initial step is to extract the product features from unstructured textual information of the project. There are similar efforts in multiple domains which aimed to extract targeted or key terms. Product feature extraction [2] or opinion mining from customer review [3] or keyword extraction [4] are amongst such efforts.

Product features extraction has three steps—preprocessing, candidate features extraction, and pruning step [5]. This study focus on the suitability of keyterm extraction techniques to extract candidate product features. So, this study will only perform preprocessing and candidate features extraction steps. The product feature extraction approaches are categorized into four main category—lexical terms frequency, syntactic relations, supervised learning, and topic models [6]. All keyterm extraction techniques except dependency parsing focus on lexical terms, so it is appropriate to compare their performance based on the topmost extracted candidate features.

In product feature extraction or opinion mining literature, a lexical term associated with the product features are nouns and with opinion toward the features are adjectives. So, the most common approach for customer review mining is to utilize part of speech (POS) analysis in order to extract noun and noun phrases [7]. Initially, a similar approach has been taken for the purpose of this study and nouns have been extracted as a candidate for 3d printer features.

Skimming through most frequent nouns shows promising results, for instance, it extracts relevant features such "material", "quality", "software", "resin", "extruder", "price". Though, these nouns alone are not completely meaningful. "Quality" on itself doesn't reveal if it is related to the printer itself or printed product.

For this reason, other approaches such as noun phrases extraction as well as other keyword extraction techniques including TextRank algorithm, Rapid Automatic Keyword Extraction (RAKE), and phrase dependency. UDPipe and TextRank R packages are used to implement these approaches on the data of 3d printer projects on Kickstarters.

The rest of paper is organized as follows. Section II explain the data gathering and pre-processing for the analysis. Section III briefly provides information about UDPipe package. Section IV explains the utilized keyword extraction techniques and approaches. Section V illustrates the results for each approach. Results are discussed in section VI and the conclusion is drawn in section VII.

II. Data

A. Extracting Data

In order to automatically extract 3D printer campaigns data from Kickstarter Platform, Selenium —a python package— has been used to develop an Application Programming Interface (API). In the first step, API uses the search word "3d printer" to find relevant projects through the Kickstarter website. In the second step, API gathers the URLs for all the relevant non-live projects—the ones that have already done with their campaign). In the last step, API uses the URL from the second step to reach project page and gather information. API has been written in Python and available on Github.

Data for 528 3d printer projects gathered in September 2017. However, only textual data including title, abstract description, and description of the campaign have been used here.

978-1-890843-39-7 © 2019 PICMET

B. Pre-processing Data

All the gathered projects are not suitable for the purpose of this study, for instance, some of the projects related to 3d printing term is a campaign to raise money to buy 3d printers for various purposes such as educational, vocational, and etc. From 519 projects which is initially gathered only 256 of them are directly related to the purpose of this study. Also, it is checked to see if there is any duplicate in data. 246 projects have been remained after removing the duplicated projects.

To build a corpus for analysis the textual information including title, abstract, description are aggregated for each project. To make the corpus ready for analysis two more steps need to be done —checking for misspelled words and replacing negations with antonyms.

For spellchecking, two Python packages have been tried—"textblob" and "enchant". However, randomly checking the output result for both packages aren't satisfying. For instance, "textbolb" changes the correctly spelled word "affordable" to "unfordable". Though the "enchant" result is more reliable than "textblob", the most detected erroneous words are the proper name. As it is demonstrated in [8] spelling errors only reduce the POS-tagging performance by 0.23% and skipping the spellchecking step won't have a significant effect on the integrity of POS-tagging analysis. So spellchecking step is skipped here.

The last step of preparing data for POS-tagging analysis is replacing negation with antonyms. In order to replace negation with antonym, first, all the contraction form of "\'nt" should be replaced by "not", and then WordNet will be used to replace negation with antonyms. RegexReplacer and AntonymReplacer class from replacers module developed by [9] have been used to accomplish these two tasks. More information on how these two classes work can be found at [10].

III. UDPipe

UDPipe is an end-to-end "open-source tool which automatically generates sentence segmentation, tokenization, POS tagging, lemmatization and dependency parsing tree"[11]. It is released under Mozilla Public License (MPL) and the UDPipe library is available in Java, Python, Perl, C#, and R platform. Deploying learning algorithm such as Gradient Recurrent Unit (GRU) network, feature averaged perceptron, neural network classifier make the pipeline trainable. UDPipe R package is utilized in this study. Training and use case details for UDPipe R is available on Github[1]. Also, UDPipe baseline system is elaborated in [11], [12].

IV. Keyword Extraction Techniques

This section focus on six keyword extraction techniques that mentioned as use cases for UDPipe R package. These techniques, also, utilized for product feature extraction and opinion mining which partly requires product features extraction in case of customer review of products. Followings are the explanation of each technique.

A. Extracting Noun using Part of Speech Tagging

Part of Speech (POS) technique tags word as a Noun, adjectives, a verb, and so on. Mostly, noun and noun phrases associated with product features in the text [2], [7], however, other constructs such as verb phrases have been used as well [13]. Since, UDPipe automatically does the POS tagging, extracting nouns is just subsetting the results.

B. Collocations and Co-occurrences

UDPipe extract multi-word expressions from the words that tend to be close to each other. The multi-word expression can be the combination of words that are in each other's neighborhood. Collocation looks at words following one another, while, co-occurrence emphasis on the frequency of words collocated, appeared within the sentence or in the close neighborhood of one another. Though more complex approaches have been taken for a term or opinion extraction, word location and co-occurrence remain as the most important elements of these complex approaches [14]. In use case example which applied here, only the nouns and adjectives are selected for collocation and co-occurrence analysis which resembles the opinion mining approaches where noun associated with product features and adjective with an opinion.

C. TextRank Algorithm

Graph-based methods are categorized as unsupervised keyword extraction methods [14]. Graph-based methods mostly construct co-occurrence network and then use network measures such as centrality, betweenness, and so on to extract keywords. TextRank is considered one of the state-of-the-art graph-based approaches [14]. TextRank is derived from the Google PageRank algorithm which utilized for keyword extraction and text summarization [15]. In the TextRank approach, the text graph consists of vertices and edges. Vertices are the sequence of one or more lexical units co-occur within a window of 2-10 words. Vertices can be restricted to lexical units of a certain part of speech including all open classes (noun, adjective, verb, and so on), nouns and verbs only, nouns and adjectives only, and etc. However, the best result observed for nouns and adjectives only which applied here. Edges represent the potentially useful connection between two vertices. In order to select key terms, TextRank goes beyond simple graph connectivity measure by considering the importance of other text units that are linked to. The extracted keywords are the most influential ones, the ones that are highly recommended by other related units. A graph-based algorithm such as PageRank is used to extract product features from text [16]. TextRank R package has been used to run the analysis in this study.

D. Rapid Automatic Keyword Extraction

Rapid Automatic Keyword Extraction or RAKE is an unsupervised, domain and language independent technique to extract key phrases [17]. In this technique, candidate keywords are the sequence of contiguous words split at phrase delimiter and stop word depositions. Then, candidate

[1] https://bnosac.github.io/udpipe/docs/doc2.html

keywords would rank based on their final score which is the sum of the score of its member. A score of each word is the ratio of word degree (how many times it co-occurs with other words calculated based on co-occurrence graph) to word frequency.

E. Noun Phrases

As discussed before, noun phrases are considered an unsupervised approach which can be used for keyword extraction [4] and also as product feature extractions [2], [7], [13]. This approach is considered a rule-based linguistic approach. In UDPipe expression pattern or rule can be determined to extract keywords.

F. Dependency Parsing

"Universal Dependencies (UD) is a project that is developing cross-linguistically consistent treebank annotation for many languages"[2], and is based on Stanford dependencies parser—to recognize textual entailment system—[18], Google universal part of speech tags[19], and interset interlingua [20]. UDPipe captures how word linked to one another based on relations defined in UD. "The UD annotation assumes the nominal, or noun phrase, as one of the basic structures that we expect to find in all languages. A nominal minimally consists of a noun, proper noun or pronoun"[3]. So to extract keywords, UDPipe casework starts from this nominal subject or nsubj and add adjectives to it.

V. RESULTS

A. Noun Extraction Results

Figure 1. shows thirty nouns with the highest frequency. Among those, words such as "part", "material", "quality", "time", "filament", "software", "resin", "source", "extruder", "resolution", "cost" and etc point out to the different characteristics of 3d printers. As illustrated, a good portion of most frequent words related to 3d printers characteristics. However, there are two major problems with this approach. The first problem is the full automation of product feature extraction based on frequency. Not all the frequent nouns are the product features and also there are low-frequency words that can represent the novel characteristics of the technology as well, though clustering words can reduce the problem can't resolve the issue completely. The second problem is the ambiguity of the meaning of the words. For instance, "quality" is word refer to the quality of technology, the printed product, or even project. This can be improved by extracting phrases or ngrams as the rest of approaches intended to do so.

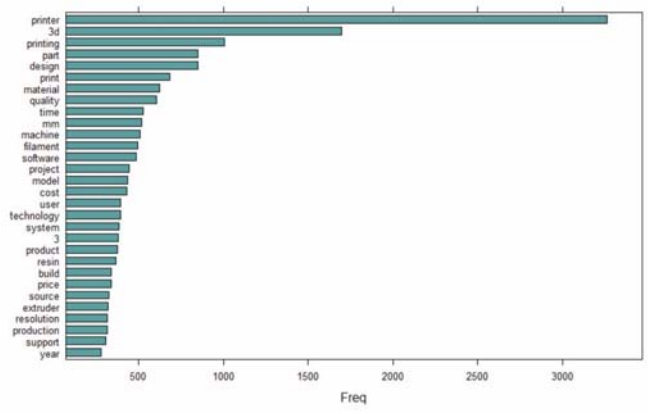

Figure 1. Thirty most occurring nouns.

B. Collocations and Co-occurrences Extraction Results

Collocations are a sequence of lexical terms which follow each other such as nouns followed by nouns, adjectives followed by nouns, and so on. Three indicators calculated for collocations in UDPipe package including—Pointwise Mutual Information (PMI), Mutual Dependency (MD), and Log-frequency Biased Mutual Dependency (LFMD). These indicators calculate how likely two terms are collocated in comparison to being independent as follows:

- PMI: log2(P(w1w2)/P(w1)P(w2))
- MD: log2(P(w1w2)^2/P(w1)P(w2))
- LFMD: MD + log2(P(w1w2))

Figure 2., Figure 3., and Figure 4. illustrate the top thirty terms when the results of collocations are sorted based on PMI, MD, and LFMD respectively. As shown, ordering the text based on these indicators neither improve the noun extraction approach nor related product features.

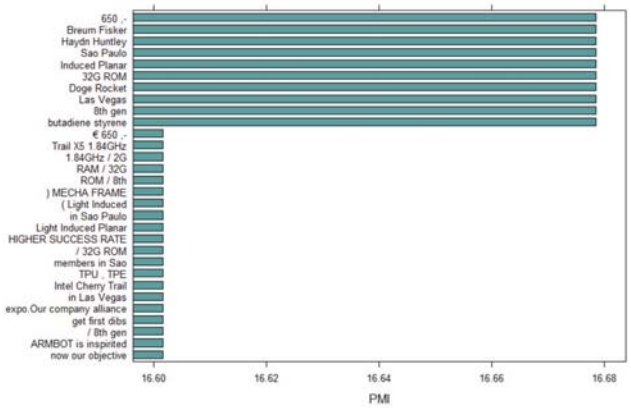

Figure 2. Thirty collocated terms with highest PMI.

[2] https://universaldependencies.org/introduction.html
[3] https://universaldependencies.org/u/overview/nominal-syntax.html

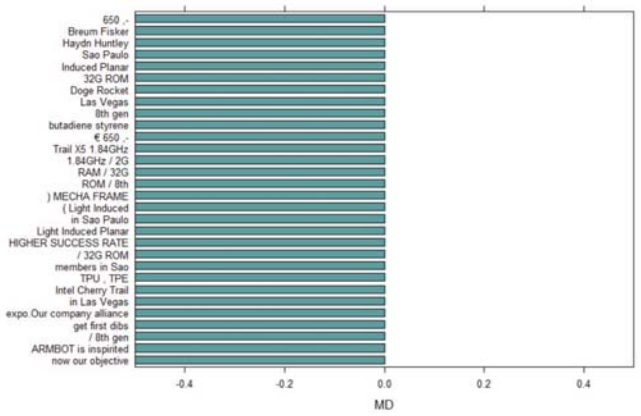

Figure 3. Thirty collocated terms with highest MD.

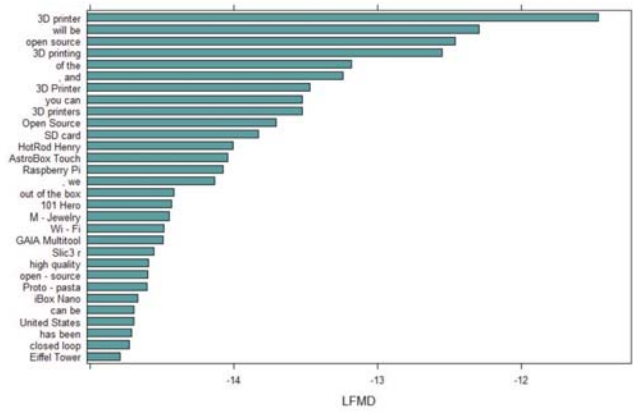

Figure 4. Thirty collocated terms with highest LFMD.

Co-occurrence analysis can be applied in various ways. Here, we analyze the co-occurrence in three ways, the co-occurrence of the words within a sentence, follow one another, follow one another in a certain distance. Figure 5, Figure 6, and Figure 7 illustrate the topmost frequent terms resulted from these co-occurrences analysis respectively.

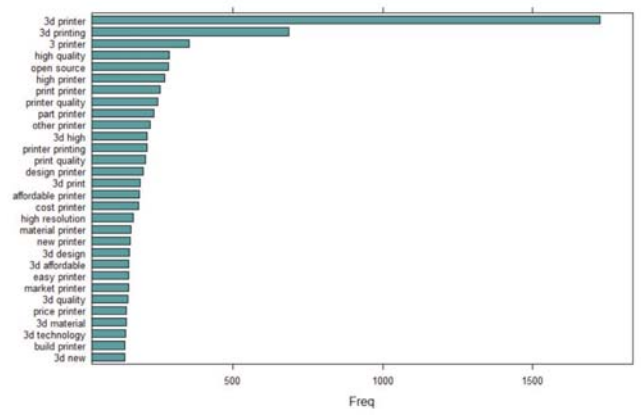

Figure 5. Thirty most co-occurring noun and adjective within a sentence.

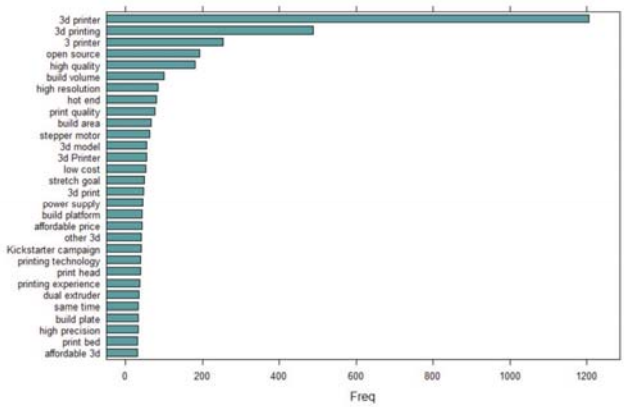

Figure 6. Thirty most co-occurring consecutive noun and adjective.

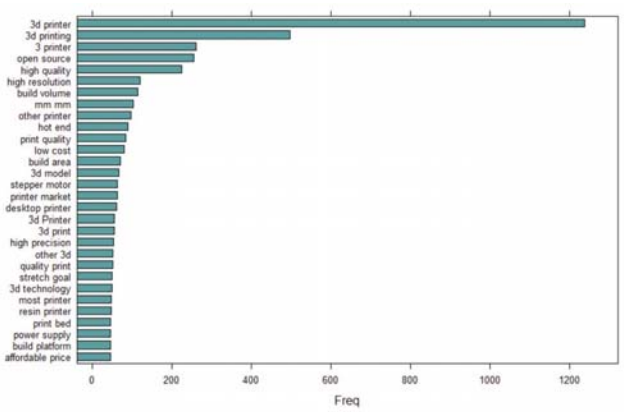

Figure 7. Thirty most co-occurring noun and adjective within the window of two words.

The result of three approaches are similar, however, the co-occurrence of following nouns and adjectives right after each other—second approach— contain more relevant product features in most frequent terms rather than two other approaches. However, all three approaches provide better and unambiguous result compared to noun extraction approach. So, the result of the second approach can be considered as a base and other approaches result compared to this to measure the improvement for extracting product feature extraction.

Network visualization the result of second co-occurrence approach shown in Figure 8 provides a better understanding of the relation between co-occurred words with frequency more than 6. For instance, we expect high precision, resolution, and quality of the print. Besides, build volume, build platform, build area, and build plate are all related to printing chambers. The big chunk of the graph shows the features and nouns connected with 3d printer technology within a certain distance. Most of the isolated terms are frequent nouns in Kickstarter platform which are general phrases and dependent of promoted technology such as "team member", "early bird", "great deal", and so on.

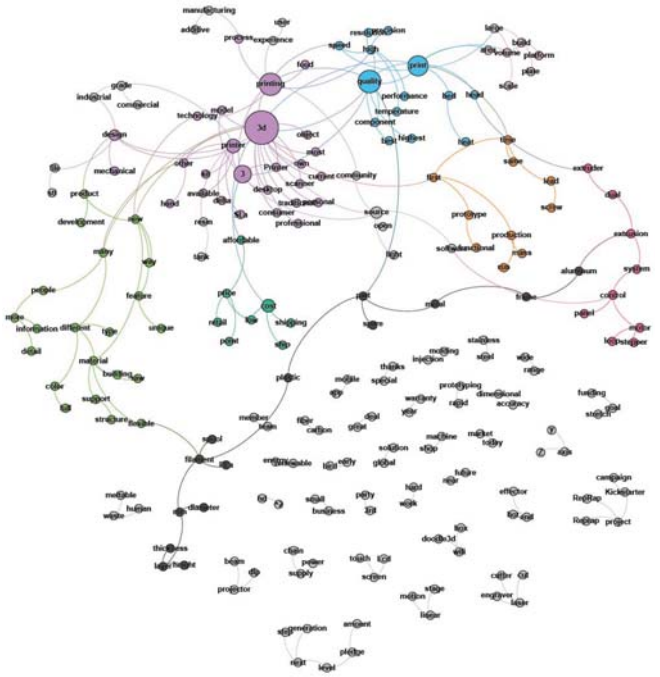

Figure 8. Co-occurrence network of consecutive noun and adjective.

C. *TextRank Algorithm Results*

TextRank algorithm is used here to extract keywords including either nouns or adjectives following one another based on most important nouns and adjectives.

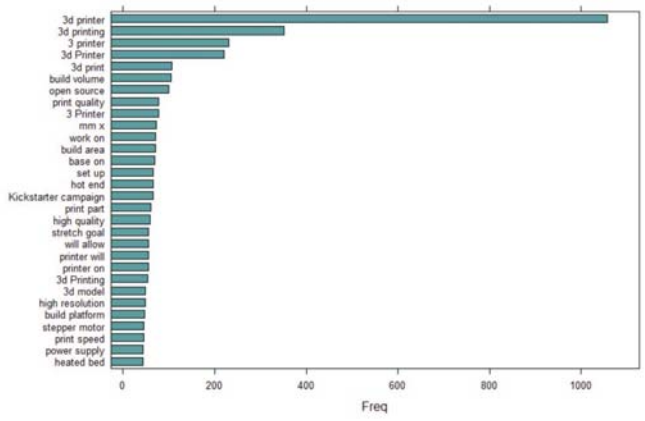

Figure 9. Thirty most frequent 2gram extracted keywords using the TextRank algorithm.

Comparing the top most frequent 2gram extracted keywords using TextRank—illustrated in Figure 9— with the top most frequent co-occurred consecutive noun and adjective—illustrated in

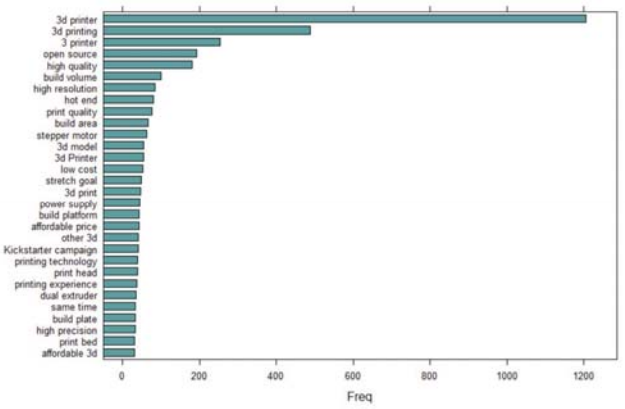

Figure 6— shows that the result of the later is richer than former. Also, the textual information of Kickstarter sometime has a mixed language which causes a problem with lemmatization. For instance, wil (noun) lemmatized as will (aux) and TextRank make this mistake bolder as "printer will" and "will allow" are appeared among the top most frequent keywords.

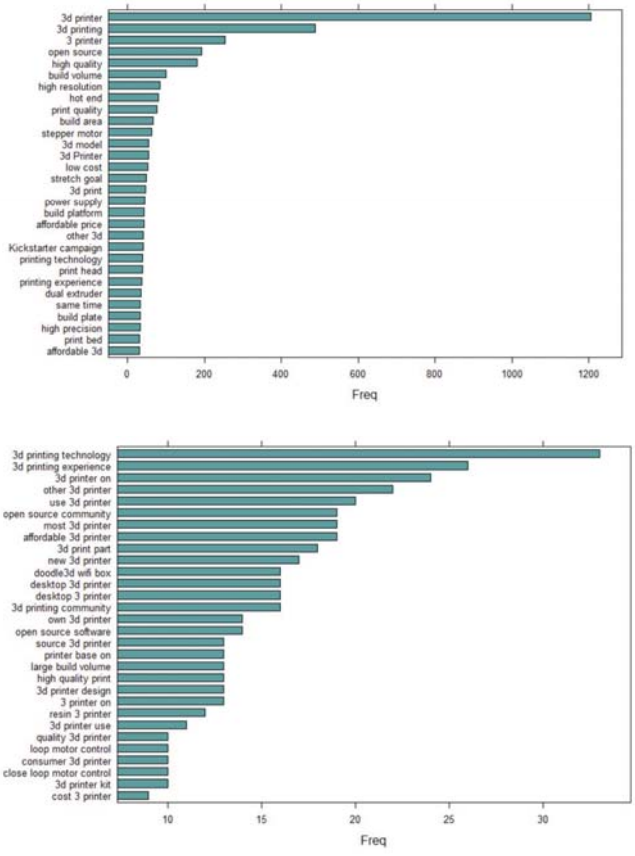

Figure 10. Thirty most frequent 3&4gram extracted keywords using the TextRank algorithm.

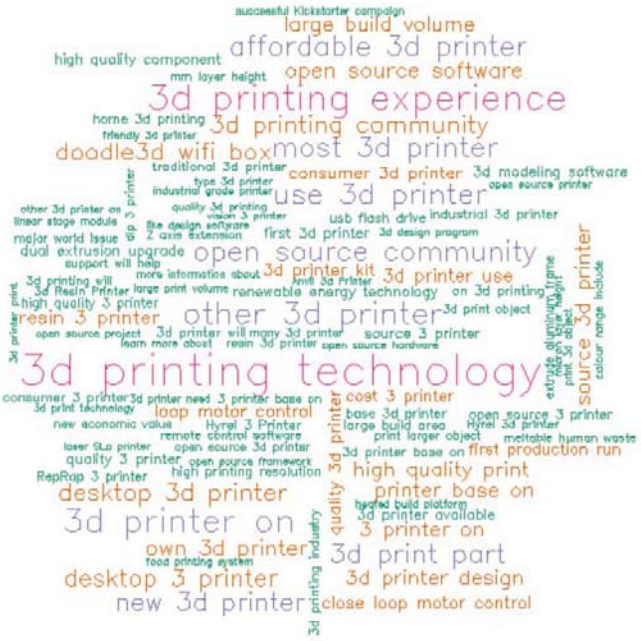

Figure 11. WordCloud of most frequent 3&4gram extracted keywords using TextRank algorithm including 100 words.

Since the co-occurrence approach just produces 2gram keywords, the result of 3&4gram extracted keywords using TextRank algorithm —illustrated in Figure 10 and Figure 11— are examined to see whether they can add value to co-occurrence results. Though the result provides some good insights like clarifying on what open source linked to, topmost frequent terms are including lots of useless terms such as "other 3/3d printer", "new 3d printer", "own 3d printer", and so on. So, the results as it is can't further enrich the noun extraction approach results.

D. Rapid Automatic Keyword Extraction Results

This section discusses the RAKE analysis results for co-occurred nouns and adjectives. Figure 12. illustrates the top thirty most frequent 2gram extracted keywords for RAKE analysis. In comparison with 2gram keywords extracted using TextRank, RAKE approach provides more relevant results in the top most frequent keywords. However, the RAKE results still are not as good as co-occurrence consecutive nouns and adjectives.

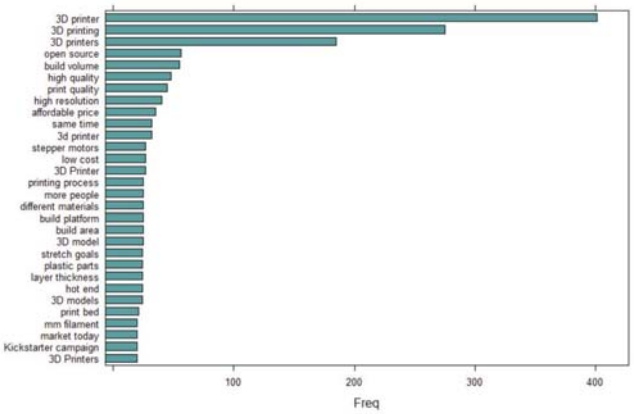

Figure 12. Thirty most frequent 2gram extracted keywords using RAKE.

Comparing the 3&4gram extracted keywords using RAKE—illustrated in Figure 13 and Figure 14— with TextRank results shows that the quality of results are similar though the top most frequent keyterms aren't.

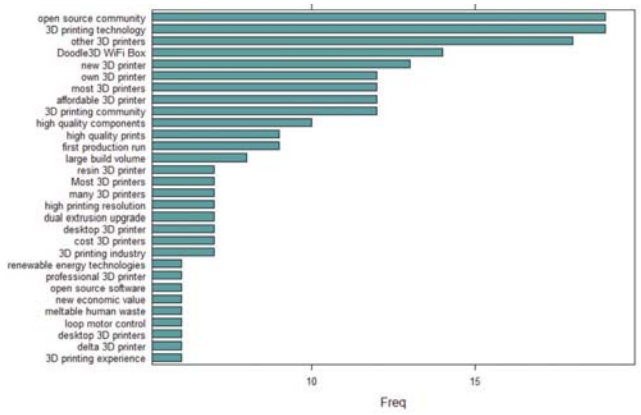

Figure 13. Thirty most frequent 3&4gram extracted keywords using RAKE.

Going above just top most frequent 3&4gram extracted keywords illustrated as WordCloud in Figure 14 shows that results are consist of useful and not very meaningful terms. So, the results as it is can't further enrich the noun extraction approach results.

Figure 14. WordCloud of most frequent 3&4gram extracted keywords using RAKE including 100 words.

E. Noun Phrases Extraction Results

An adjective+noun, pre/postposition, optional determiner, and another adjective+noun is the pattern used to extract the simple noun phrases from the corpus. Figure 15. illustrates the top thirty most frequent 2gram extracted noun phrases. As shown, the result of this approach is the poorest compared among the result of approaches discussed so far. Many phrases in most frequent keyterms start with determiner such as "your printer", "our printer", "that we", "our website", and etc. Even looking into more frequent results in Figure 16. does not indicate rich results. The poor result may be because of how simple noun phrases pattern are determined. Changing noun phrases construction can improve the quality of results, however, finding the pattern which yields desirable keyterms beyond the scope of this paper and doesn't worth extra time since co-occurrence analysis already provides decent results.

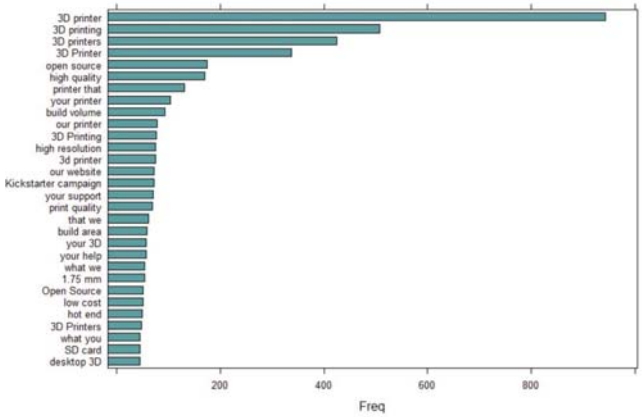

Figure 15. Thirty most frequent 2gram extracted noun phrases.

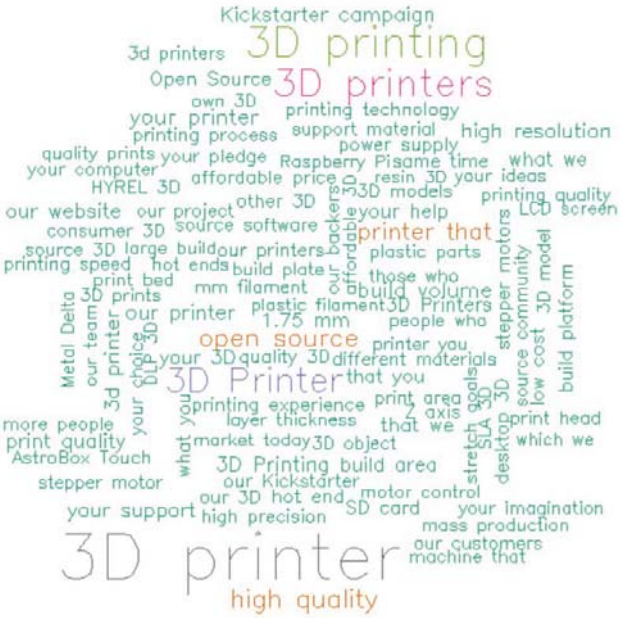

Figure 16. WordCloud of most frequent 2gram extracted noun phrases including 100 words.

Figure 17. illustrates the top thirty most frequent 3&4gram extracted noun phrases. Unlike the 2gram results, noun phrases extraction approach performance is comparable with the performance of TextRank and RAKE approach. And it is the first approach that captures the type of 3d printer technology including SLA and DLP in the most frequent keyterms. Also, it points out that the most popular filament size which is 1.75 mm filament. Though, the results aren't a good addition to the co-occurrence approach as it is. The results include many meaningless terms such as "3d printer that", "other 3d printer", "your 3d printer", and so on.

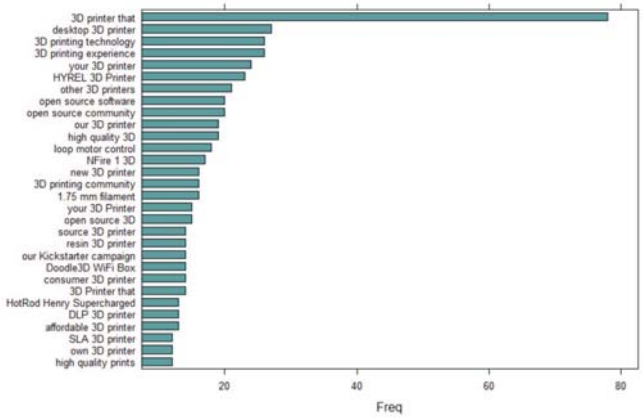

Figure 17. Thirty most frequent 3&4gram extracted noun phrases.

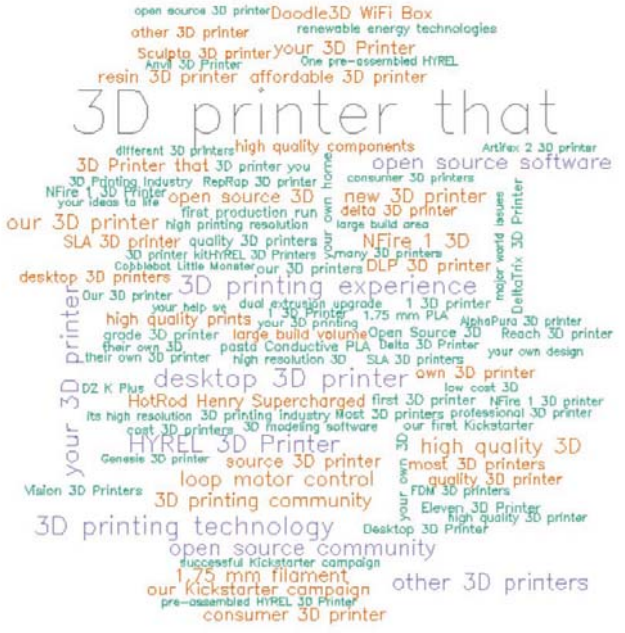

Figure 18. WordCloud of most frequent 3&4gram extracted noun phrases including 100 words.

F. Dependency Parsing Results

Figure 19 illustrates the result of dependency parsing approach. The results are very limited and have zero value compared to other approaches.

Figure 19. Wordcloud extracted keywords using Dependency Parsing.

VI. Discussion

Six keyword extraction techniques—nouns extraction, collocations and co-occurrences of nouns and adjectives, TextRank algorithm with vertices restricted to nouns and adjectives, Rapid Automatic Keyword Extraction (RAKE) based on co-occurrence of nouns and adjectives, simple noun phrase, dependency parser of the nominal subject— have been applied to 3d printers textual information.

These six techniques are carried out using UDPipe R package. Comparing the top thirty frequent unigram and bigram terms in six techniques show that co-occurred nouns and adjectives technique includes more meaningful and relevant 3d printer technologies features and associated opinions.

Part of speech (POS) tagging—lexical terms—, collocations, and co-occurrences are almost the base of all the approaches used in this study. Based on opinion mining literature, nouns and adjectives are the base lexical terms for all the approaches. Though, extracting nouns result in good yet ambiguous terms.

The second approach in which co-occurrence of nouns and adjectives explored improves the only noun extraction approach by reducing the features ambiguity. Co-occurrence of nouns and adjectives techniques resembles to Hu and Liu [3] approach to extract features and opinions. However, Hu and Liu [3] believe that product features can be unigram, bigram, and trigram.

Though, implemented co-occurrence of nouns and adjective techniques using UDPipe only produce bigram terms which can be a co-occurrence of two nouns, two adjectives, or noun and adjective. The co-occurrence of two adjectives is rare compared to two nouns—feature phrase— or noun and adjective—feature and associated opinion.

Other techniques such as TextRank and RAKE—based on co-occurrence graph of nouns and adjectives— can produce n-grams terms with n higher than 2, however, the results aren't decent for the feature and opinion extraction. Among the chosen approaches for this study, dependency parsing approach yields the least useful result.

The performance of approaches including TextRank, RAKE, and noun phrase extraction compared based on two criteria. Direct comparison of the top most frequent 2gram result of these approaches with the top most frequent co-occurred nouns and adjectives to find out which approach capture more meaningful aspects of technology. The second criteria are to find if the approaches mentioned above can provide useful 3 and 4-gram keywords.

The result did show that no approach can surpass the performance of co-occurrence consecutive nouns and adjectives in extracting 2gram product features. The graph-based approach shows the promising result to extract higher than 2gram keywords, though, not in the current format. Also, the phrase extraction approach can be useful to extract 3 & 4gram keywords if the right pattern can be found.

To overcome the issue, the same approach as Hu and Liu [3] could be taken to extract noun phrases as well as nouns. However, UDPipe package is not suitable for extracting noun phrases since it requires the noun phrase to be defined and

noun phrase pattern are numerous which makes the approach time consuming and not thorough enough.

VII. Conclusion

This study is the first step to automate the product feature extraction in crowdfunding—specifically Kickstarter— to facilitate studies such as decision making, new product analysis, lead user identification, technology forecast, and so forth. The first step involves finding ready to use tools and approaches such as keyterm extraction techniques to use them as it is or use them in modifying and establishing domain suitable approach.

This study shows that UDPipe R packages which provides end to end tool to perform some of preprocessing steps and includes technique for extracting keyterms cannot fill the gap for comprehensive and ready to use tool for extracting product features in popular environments such Python and R. However, package is still very useful and easy to use for performing required preprocessing steps such as tokenization, POS, and dependency parsing.

Moreover, co-concurrent consecutive nouns and adjectives yield the most meaningful and frequent 3d printer features and these results are useful to establish a new approach. For instance, a co-occurrence graph built based on co-occurrence analysis can be used to modify the rules in a rule-based approach [21] to extract candidate product features. The main focus here is on extracting candidate features, however, the co-occurrence graph is also useful for pruning the noise to filter the results.

On the other hand, environment such as R and Python are rich in tools to handle the post-extraction to prepare the features for further analysis. Clustering the semantically similar terms into the unique category and quantification is an example of these post-processing steps.

References

[1] J. Riedl, "Crowdfunding technology innovation," *Computer*, vol. 46, no. 3, pp. 100–103, 2013.

[2] R. Ghani, K. Probst, Y. Liu, M. Krema, and A. Fano, "Text mining for product attribute extraction," *ACM SIGKDD Explorations Newsletter*, vol. 8, no. 1, pp. 41–48, 2006.

[3] M. Hu and B. Liu, "Mining opinion features in customer reviews," in *AAAI*, 2004, vol. 4, pp. 755–760.

[4] S. Siddiqi and A. Sharan, "Keyword and keyphrase extraction techniques: a literature review," *International Journal of Computer Applications*, vol. 109, no. 2, 2015.

[5] M. Z. Asghar, A. Khan, S. Ahmad, and F. M. Kundi, "A review of feature extraction in sentiment analysis," *Journal of Basic and Applied Scientific Research*, vol. 4, no. 3, pp. 181–186, 2014.

[6] B. Liu, *Sentiment analysis: Mining opinions, sentiments, and emotions*. Cambridge University Press, 2015.

[7] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004, pp. 168–177.

[8] T. Mizumoto and R. Nagata, "Analyzing the Impact of Spelling Errors on POS-Tagging and Chunking in Learner English," in *Proceedings of the 4th Workshop on Natural Language Processing Techniques for Educational Applications (NLPTEA 2017)*, 2017, pp. 54–58.

[9] J. Perkins, "nltk3-cookbook," *GitHub repository*, 2014. [Online]. Available: https://github.com/japerk/nltk3-cookbook.

[10] J. Perkins, *Python 3 text processing with NLTK 3 cookbook*. Packt Publishing Ltd, 2014.

[11] M. Straka and J. Straková, "Tokenizing, pos tagging, lemmatizing and parsing and 2.0 with udpipe," *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pp. 88–99, 2017.

[12] M. Straka, J. Hajic, and J. Straková, "UDPipe: Trainable Pipeline for Processing CoNLL-U Files Performing Tokenization, Morphological Analysis, POS Tagging and Parsing.," in *LREC*, 2016.

[13] N. Archak, A. Ghose, and P. G. Ipeirotis, "Deriving the pricing power of product features by mining consumer reviews," *Management science*, vol. 57, no. 8, pp. 1485–1509, 2011.

[14] S. Beliga, A. Meštrović, and S. Martinčić-Ipšić, "An overview of graph-based keyword extraction methods and approaches," *Journal of information and organizational sciences*, vol. 39, no. 1, pp. 1–20, 2015.

[15] R. Mihalcea and P. Tarau, "Textrank: Bringing order into text," in *Proceedings of the 2004 conference on empirical methods in natural language processing*, 2004.

[16] Z. Yan, M. Xing, D. Zhang, and B. Ma, "EXPRS: An extended pagerank method for product feature extraction from online consumer reviews," *Information & Management*, vol. 52, no. 7, pp. 850–858, 2015.

[17] S. Rose, D. Engel, N. Cramer, and W. Cowley, "Automatic keyword extraction from individual documents," *Text Mining: Applications and Theory*, pp. 1–20, 2010.

[18] M.-C. De Marneffe *et al.*, "Universal Stanford dependencies: A cross-linguistic typology.," in *LREC*, 2014, vol. 14, pp. 4585–4592.

[19] S. Petrov, D. Das, and R. McDonald, "A universal part-of-speech tagset," *arXiv preprint arXiv:1104.2086*, 2011.

[20] D. Zeman, "Reusable Tagset Conversion Using Tagset Drivers.," in *LREC*, 2008, vol. 2008, pp. 28–30.

[21] G. Qiu, B. Liu, J. Bu, and C. Chen, "Opinion word expansion and target extraction through double propagation," *Computational linguistics*, vol. 37, no. 1, pp. 9–27, 2011.