Generating the MEDLINE N-Gram Set

Chris J. Lu, Ph.D.[1, 2], Destinee Tormey[1], Lynn McCreedy, Ph.D.[1] and Allen C. Browne [1]
[1]National Library of Medicine, Bethesda, MD; [2]Medical Science & Computing, Inc, Rockville, MD

Abstract
The MEDLINE n-gram set is a very useful resource in Natural Language Processing (NLP) and Medical Language Processing (MLP). Currently, there is no MEDLINE n-gram set available in the public domain. Due to the large scale of data, it is a challenge to generate MEDLINE n-grams to fit into a research schedule with limited computer resources. The Lexical System Group (LSG) developed an algorithm to generate the MEDLINE n-gram set for adding multiwords into the SPECIALIST Lexicon. We believe the NLP community can benefit from access to this big data. We processed 2.6 billion single words from 22.4 million MEDLINE documents (titles and abstracts) to generate MEDLINE n-grams (n = 1 to 5) with terms appearing at least 30 times and having less than 50 characters for the 2014 release.

1. Background, Motivation, and Challenge

The LSG used the MEDLINE n-gram model to generate multiwords in 2014. A sophisticated algorithm has been developed to generate a comprehensive MEDLINE n-gram set to replace the approximated n-gram model by prediction filter [1]. N = 1 to 5 were chosen because these terms cover up to 99.47% of the terms in the Lexicon, as shown in Table-1. Also, 99.55% of terms in the Lexicon have less than 50 characters and thus terms with more than 50 characters are filtered out. Both word count (WC) and document count (DC) are calculated. The approach of deriving n-grams by bigrams doesn't provide accurate WC|DC. Titles and abstracts from 22,356,869 MEDLINE documents are collected and then tokenized to 126,612,705 sentences and 2,610,209,406 words (tokens) in the 2014 release. Our sentence tokenizer found very few unrecognized sentence patterns (0.01% - 14,314). Java class (HashMap) is used for the key-value mechanism to store n-grams as keys and their associated information (WC, DC, and PMID) as values in the process. This system works perfectly for unigrams and bigrams. However, this approach does not work for n-grams where n > 3 because the number of terms (keys) exceeds the limit of maximum keys in the Java HashMap class (2^{30}-1). The processing time is not feasible (it can take months) when using persistent key-value DB or other database (embedded or server) approaches to replace the Java HashMap, even with a powerful computer that has 192 GB memory.

2. Approach – Split, Group, Filter, Combine, and Sort

An algorithm was developed to resolve the issues stated in the previous section. First, it splits the MEDLINE documents into S sections so that n-grams of each section can be generated within Java limitations. Note that duplicated terms are distributed across S sections. Second, it retrieves unique terms (keys) and calculates WC and DC by grouping terms from a range of alphabetic characters, such as terms from 'c' to 'f', through generated S sections. Each range of grouped (unique) terms, which are within Java limitation, is saved into G groups. Third, it applies WC and DC filters through all groups and then combines all filtered terms to form the n-gram set. Fourth, it sorts the n-gram set by DC, WC, and the alphabetic order of terms. For example, 20 sections of 5-grams were generated from MEDLINE titles and abstracts. Then, 14 groups with different ranges of alphabetic characters of unique 5-gram are retrieved from these 20 sections. The program combines the 5-grams by filtering out the low frequency terms from these 14 groups. Table-2 shows 1.54M 5-grams with WC >= 30 are retrieved from 1.67G 5-grams in this step. Finally, these filtered 5-grams are sorted and saved in a file. The idea is to keep those large numbers of invalid n-gram terms (N = 3 to 5) in different sections and groups, to resolve the Java limitation issue. The WC and DC filter is then applied to remove these low frequency (invalid) n-gram terms before the final combination.

N	Word Count	Cumulative Word Count
1	457,335 (52.26%)	457,335 (52.26%)
2	281,857 (32.21%)	739,192 (84.47%)
3	93,011 (10.63%)	832,203 (95.10%)
4	29,905 (3.42%)	862,108 (98.52%)
5	8,358 (0.96%)	870,466 (99.47%)

Table-1. (Multi)Word Distribution in the Lexicon

N	N-gram	N-gram, WC >= 30
1	21,530,469	804,382
2	205,868,398	4,587,349
3	703,148,136	6,287,536
4	1,295,096,308	3,799,377
5	1,665,248,566	1,545,175

Table-2. 2014 MEDLINE N-grams

3. Conclusion

The 2014 MEDLINE n-gram set is successfully generated by the above method. Table-2 shows the number of each n-gram. Three fields (document count, word count, and n-gram) are included. The MEDLINE n-gram set is distributed annually by the National Library of Medicine (NLM) with the SPECIALIST Lexicon annual release via an Open Source License agreement [2].

References
1. C.J. Lu, D. Tormey, L. McCreedy, A.C. Browne, "Using Element Words to Generate (Multi)Words for the SPECIALIST Lexicon", AMIA 2014 Annual Symposium, Wash., DC, Nov. 15-19, 2014, P. 1499.
2. Lexical Systems Group, n-gram in multiwords projects: http://umlslex.nlm.nih.gov/nGram