Using the following Unix command we can get the top 40 words. I've split them into groups of 10 manually for readability. I am removing capitalization for the first letter of words so there is a more accurate count (handling beginning of sentences). I am delimiting solely on spaces, so punctuation may be attached to some words.

```
cat uncorpus.eng.txt | perl -pe 'tr/A-Z/a-z/;' | perl -pe 's/\s/\n/g;'
| sort | uniq -c | sort -nr | head -40
```

```
578773
272274 the
175498 of
136608 and
101442 to
67237 in
35911 on
32424 for
22567 that
21288 its

20916 a
20383 with
20010 united
19994 as
17748 international
17262 by
17189 nations
14018 at
13408 all
12037 states

11418 their
9379 human
9292 december
9039 general
8966 development
8832 including
8795 or
8535 resolution
8240 rights
7344 other

7184 secretary-general
7064 report
6852 implementation
6592 also
6481 be
6347 committee
5980 from
5956 which
```

5908 requests
5488 relevant

Comparing these four groups, we see that the top words are articles, prepositions, etc. These are all typically considered filler words as they provide little help to what the text is about. What is interesting is the most common word is empty. This is an error with the script and I'm not sure what is causing this. I assume if there are two spaces next to eachother the script is putting in 2 \n which leads to empty lines when finding unique words. The second group still contains some articles and filler words (a, as, at) but we start to get an idea of what this document may be about. united, nations, international, states are all present, showing that the text most likely has something to do with foreign policy. On the third and fourth level, we have no more filler words and are left with adjectives and nouns. These words give us a better idea what the writing is about, but it is interesting to note that there are very few verbs in the top 40 list.

Using the following Unix command we can get the bottom 40 words. I've split them into groups of 10 manually for readability. I am removing capitalization for the first letter of words so there is a more accurate count (handling beginning of sentences). I am delimiting solely on spaces, so punctuation may be attached to some words.

```
cat uncorpus.eng.txt | perl -pe 'tr/A-Z/a-z/;' | perl -pe 's/\s/\n/g;'
| sort | uniq -c | sort | head -40
```

```
   1 "
   1 "(b)"
   1 "(conclusion
   1 "(d)"
   1 "(d)";
   1 "(e.g.,
   1 "(i)".
   1 "(ii)"
   1 "(ix)

   1 "(k)
   1 "(l)
   1 "(m)
   1 "(n)
   1 "(o)
   1 "(p)
   1 "(v)
   1 "10
   1 "100
   1 "200"
```

```
   1 "22.1
   1 "22.2
   1 "22.3
   1 "23.
   1 "27c.5.
   1 "4
   1 "4.5
   1 "6
   1 "600".
   1 "8

   1 "80
   1 "88
   1 "a.
   1 "academic
   1 "action
   1 "adapting
   1 "addressee"
   1 "addressing
   1 "administration
   1 "adult"
   1 "advisory
```

The bottom 40 words all have a frequency of 1, so it doesn't necessarily matter what order these groups are in. For that reason it is less helpful to compare these groups to each other and more helpful to discuss these 40 words as a whole. To start, it is clear that the reason these tokens are showing up are due to poor tokenization. Since I was only delimiting on white space I was not separating away quotes and parentheses from our tokens. This makes "academic unique from academic, yielding scewed results. I'm not surprised by numbers, roman numerals, and letters in parentheses being located in the bottom 40 tokens, but I would have expected some different results if I tokenized using a better algorithm.