Simplification data README draft

This page consists of our **draft** README for the data release related to our paper,

For the sake of simplicity: Unsupervised extraction of lexical simplifications from Wikipedia
Mark Yatskar, Bo Pang, Cristian Danescu-Niculescu-Mizil and Lillian Lee
Proceedings of the NAACL, 2010 (short paper).

We plan to complete a final version of the README later, but wanted to quickly provide enough details here for interested parties to be able to start making use of the data beforehand.

All data files can be downloaded from this page ("Data for lexical simplification experiments").

- The most "processed" data can be found in **basefiles.tar.gz** (135MB)
- We also provide the following supplementary files:
  - enwiki.tar.gz (1.67GB); simplewiki.tar.gar (88.9MB)
  - fullwiki_files.tar.gz (1.66GB); simplewiki_files.tar.gz (88.9MB)
  - simple.ids.titles
  - full.ids.titles.sid
  - output.threshold
  - output.translation

Overview

Processing the Simple/English Wikipedia revision streams was done in three phases. Initially, articles of interest were identified and their corresponding revisions were extracted from wiki dumps. Each article has a unique id, ARTICLE_ID, and contains a set of revisions. Each revision has a unique id, REVISION_ID. Also, each revision has an optional comment representing the transition from the previous state to the current revision of the article (the first comment represents the creation of the article).

For each article (a set of revisions), a list of all unique sentences in the revision stream of this article was generated. Each sentence was given an id, and each revision of the article was then represented as a sequence of SENTENCE_IDs that belonged to it. This was a reasonably compact representation of the stream.

Afterward, sentences in adjacent revisions of an article were aligned using a TF-IDF weighting scheme where Document Frequency was computed in the sense that a revision in an article was considered a document and the corpus was all revisions of an article. This is used to generate an ALIGNMENT_SCORE.

Finally, phrases from aligned sentences were extracted by finding a single differing segment in each of the two sentences (PHRASE). If changes were large, the single difference could be each sentence; if the changes were small, the single differing segment would could be a single word from each sentence.

basefiles.tar.gz:

These files contain the *lexical edit instances* (PHRASE1 -> PHRASE2) used in our paper. We also provide the aligned sentence pairs from which these lexical edit instances were extracted. We include only sentence pairs that satisfy the following criteria:

- ALIGNMENT_SCORE > .3 using the TF-IDF-based alignment described above; and
- both PHRASE1 and PHRASE2 extracted from this sentence pair contain no more than 5 words.

There are three classes of files (all fields are tab-separated):

- *.extra: ARTICLE_ID, REVISION1_ID, REVISION2_ID, SENTENCE1_ID, SENTENCE2_ID, ALIGNMENT_SCORE, PHRASE1_LENGTH, PHRASE1, PHRASE2_LENGTH, PHRASE2, SENTENCE1, SENTENCE2, COMMENT
- *.sp: a subset of *.extra: an instance is filtered out if PHRASE1 and PHRASE2 have identical soundex.
- *.cut3: ARTICLE_ID, PHRASE1, PHRASE2

- *.simpl: a subset of *.extra containing instances associated with COMMENT that contains "simpl*"

These files are provided for three types of data:

- fullwiki: an extraction of articles from english wikipedia that were also found in simple wikipedia.
  This is not all shared articles, but the first 80% that were found sequentially searching the full wikipedia dump. Older articles will be earlier in the dump.
- simplewiki: an extraction of articles from simple wikipedia that were also found in the english wikipedia.
- simplewiki.all: an extraction of all articles that were found in the simple wikipedia.

simplewiki.tar.gz:

Intermediate files with basic pre-processing of revision data. Each "part" contains a subset of the data (due to Hadoop processing) and contains information about many pages.

There are five types of files (where * specifies part001,part002, ect):

- *.df: PAGE_ID, WORD, FREQ (the first line corresponding to an article lists its name). As noted earlier, for a given article, document frequency is computed over its revisions, where each revision is considered as a "document".

- *.sentid_sent: PAGE_ID, SENTENCE_ID, SENTENCE
- *.revid_sentid: PAGE_ID, REVISION_ID, UNSIMPLE_FLAG, SENTENCE_COUNT, SENTENCE_STREAM, COMMENT
- *.directed_sentid_sentid_w: PAGE_ID, REVISION1_ID, REVISION2_ID, SENTENCE1_ID, SENTENCE2_ID, ALIGNMENT_SCORE, PHRASE1_LENGTH, PHRASE1, PHRASE2_LENGTH2, PHRASE2
- *.index: This is an index for looking up a particular page's info within the part files. There is one of these for each of the types listed above, but it takes the same format: PAGE_ID, END_LINE_NUMBER, NUMBER_OF_LINES.

Two fields need additional explanation:
SENTENCE_STREAM: This is a comma separated string of SENTENCE_IDs in the order they appear in the revision, with the character "P" used to indicate paragraph breaks.
UNSIMPLE_FLAG: This a flag for whether or not the revision was tagged with the "UNSIMPLE" group. Editors had the option of marking a file with this tag to indicate that it needed further simplification.

enwiki.tar.gz

Similar to simplewiki.tar.gz, intermediate files generated when processing the english wikipedia.

simplewiki_files.tar.gz

This is the same information as simplewiki.tar.gz but where each article is in its own folder and the index files thrown away.
That folder contains df,sentid_sent, revid_sentid, direction_sentid_sentid_w which only correspond to that one article.
This is convenient for browsing documents.

enwiki_files.tar.gz

The same as simplewiki_files.tar.gz except for the english wikipedia.

simple.ids.titles

A map for simple article titles
PAGE_ID, SIMPLE_TITLE

full.ids.titles.sid

A map from english article ids to titles to simple article ids.
ENGLISH_PAGE_ID, TITLE, SIMPLE_PAGE_ID

output.threshold

Full output of the simplification system used for evaluation. Here we report the top simplification from a source word given that there was some simplification that surpassed a threshold of being at least 45% likely. It is sorted by the probability that the source word needs simplification.

SOURCE_WORD TARGET_WORD

output.translation

The full translation table derived from our simplification model. As a note, in the paper we reported 1079 pairs, in fact there are 1078 because of a spuriously printed line.

PROBABILITY_OF_SIMPLIFICATION SOURCE_WORD NUM_TARGET_WORDS (TARGET_WORD TARGET_PROBABILITY)+