Word Vectors for the Thoughtful Humanist Institute
Data Preparation Guide and Checklist
By Laura Johnson

During the first Word Vectors for the Thoughtful Humanist Institute at Northeastern this July, there were many engaging conversations about the role of data preparation in computational analytics, in terms of both technical concerns related to assessing the significance of results, testing, and data messiness or incomprehensibility. At this first research-oriented institute, participants came with a wide array of corpora. In order to share the aspects of data preparation that were most useful for this institute, this document contains a guide with detailed checklists and questions for consideration. Every corpus is different, yet there are some core features and elements of textual data that can usefully be considered when preparing data for word embedding models.

As we discussed throughout the institute, data preparation is something that is ongoing and just as much a part of experimentation with word embedding models as training the model is. How you prepare your data directly impacts what kind of results you get by training a model. I have split the process of data preparation into five stages: 1) performing initial data exploration, 2) analyzing the data and identifying "noise", 3) creating a data preparation plan, 4) cleaning or modifying the corpus, 5) reflecting and repeating (as necessary). As with testing a model, preparing data is an ongoing, evolving process; the more you do, the more you recognize what you could do next. In outlining data preparation in this way, I hope to emphasize two key concepts. First, time and computing power are important concerns to be realistic about. It is tempting to make all of your changes at once, but this process is iterative for a reason. Second, the best thing that anyone can do in this process is simply take careful notes. Any step you take to prepare your data should be written down because you likely will not remember everything if you are making big changes. It is harder to retrace your steps than you think. While there can be a great deal of variability with word embedding models, having a data preparation plan with detailed notes will let you reduce some degree of uncertainty about the data you are using to train your models.

Stage 1: Data Exploration

In order to properly prepare a corpus, you have to understand what it contains. In this context, exploring and understanding a corpus can be conceptualized as creating a "profile" with basic and advanced kinds of information. Consider the following questions:

- ❏ Where did your data come from and how is it organized? Did you organize it?
- ❏ What format is the data (plain text, OCR-generated data, XML/TEI data)? How familiar are you with this format?
- ❏ Will your data require any conversion to plain text data?

- ❏ How many files does your corpus contain? How large are these files, on average? What is the range of their size?
- ❏ What is the total word count for the entire corpus?
- ❏ What are the most frequent words in the corpus? In each document?
- ❏ How was your corpus created? Were all the texts taken from the same source or are they from multiple sources?
- ❏ Do you see any commonalities or themes across the texts in the corpus?
- ❏ Are there any inconsistencies in formatting and data structure throughout your corpus?
- ❏ Is there any metadata in your corpus and where is it located?

If you are working with textual data, there are a number of ways to collect and compile information for your corpus profile. Some user-friendly (and free) tools are Voyant Tools (https://voyant-tools.org/) and AntConc (https://www.laurenceanthony.net/software/antconc/). Voyant Tools is a web-based tool and does not require any installation. If you are unfamiliar with it, I recommend this tutorial and documentation for understanding what kinds of textual analysis it is capable of: https://voyant-tools.org/docs/#!/guide/tutorial. AntConc, unlike Voyant, needs to be downloaded and installed but also has a lot of documentation: https://www.laurenceanthony.net/software/antconc/releases/AntConc358/help.pdf.

With either of these tools, it is easy to survey a corpus for important structural and thematic elements using different functions: word frequency, concordance, collocations, word clusters, n-grams, sentence length, and vocabulary density. To gain an “aerial” view of your data, I also recommend using topic modeling. One easy-to-use program is DARIAH (https://dariah-de.github.io/TopicsExplorer/). While there are a lot of different perspectives on topic modeling as a form of textual analysis, it can be very useful for reading across a larger corpus for key themes, and thus can help you to better understand what is in your corpus as you plan for data preparation.

On the topic of reading, another useful way to understand a corpus—especially if it was not hand curated or if you are unfamiliar with the file format, structure, and content—is simply to choose random files or sections and begin reading. Reading with data preparation in mind brings up different features than reading strictly for content. Take a sampling of the corpus by choosing random pages or files, and reading those with the above questions in mind. If you are using a textual editor like Oxygen, BBEdit, or Atom, you can additionally read across a corpus by using simple features like “find all” and, for XML documents, XPath. Using tools to aid in exploring a corpus has two positive effects: increased knowledge and improved navigatibility. The more time you spend navigating the corpus without an initial agenda, the easier it is to understand what it contains. Once you have all of this information, next comes analysis and creating an action plan.

Stage 2: Data Analysis and Identifying “Noise”

After your initial data exploration, the next step is to analyze your data to identify what features are “noise” and may impact your word embedding model. Depending on the type of data in your corpus, there are several different features to look for:

- ❏ **Metadata:** after determining if your corpus contains metadata (and if it does, where that metadata appears), you need to determine if that metadata can be of use, either for further exploration or in ordering your texts. However, remember that when the model is trained, any metadata in the corpus will be treated the same as the rest of your textual data and so, if you don't want to study the language of your metadata, you should remove it.
  Examples of metadata include:
  - ❏ publication statements
  - ❏ source descriptions
  - ❏ data licenses or agreement statements
  - ❏ citation information
  - ❏ encoding descriptions
  - ❏ revision statements
  - ❏ copyright information

- ❏ **Transcription Information:** if your data was transcribed, there may be transcription notes or other artifacts of the transcription process. Much as with metadata, it is likely you will want to remove these, as they are not part of the original documents. This information might include:
  - ❏ transcribers notes
  - ❏ figure descriptions
  - ❏ transcribers’ or editors’ annotations
  - ❏ markers for uncertainty, additions, or deletions

- ❏ **Structuring Features:** the textual features that mark document structures can vary widely, but they are generally noisy, with repeated headers, numbers, or other words that will all be treated the same in the word embedding model as the contents of the documents, if they are not removed. Features that we recommend removing include:
  - ❏ page numbers
  - ❏ chapter titles or headers
  - ❏ section, book, act, or volume headers
  - ❏ illustrations
  - ❏ data tables or graphs
  - ❏ speaker labels (in drama; you may also want to consider removing stage directions)
  - ❏ usernames or timestamps

- ❏ URLs
- ❏ running header and footers

- ❏ **Paratexts**: depending on the structure of your textual data, there may be paratexts that do not make sense to keep in your corpus. For example, tables of contents describe the contents of a text, but they are not the text itself. Before removing any paratexts, determine how frequent these features are, identify how you would remove them, and decide if you are interested in them. Most often, it is more effective to remove paratexts, including the following:
  - ❏ tables of contents
  - ❏ pronunciation guides
  - ❏ prefaces
  - ❏ appendices
  - ❏ frontispieces
  - ❏ title pages
  - ❏ acknowledgements
  - ❏ abstracts
  - ❏ letters from the editor (letters from the author may also be worth removing, depending on your project)
  - ❏ advertisements
  - ❏ indexes

Transformations and Regularizations

For corpora that contain data that is not already in plain text, an important consideration is how you will transform the text. If your data is TEI or any other form of XML, it is fairly straightforward to use XQuery to transform XML to plain text. Additionally, due to how features are tagged in XML documents, you can combine the cleaning and transformation into the same process, removing certain elements or setting the parameters to transform certain portions of the text. The WWP has several XQueries for this purpose on GitHub (https://github.com/NEU-DSG/wwp-public-code-share/tree/master/fulltext). These include an XQuery that can be used to transform non-TEI XML data to plain text (https://github.com/NEU-DSG/wwp-public-code-share/blob/master/fulltext/fulltext2table.non-tei.enmasse.xq).

Another important aspect of data preparation is regularization. Popular forms of regularization in data are modernizing archaic spellings, expanding abbreviations, fixing OCR errors, and correcting misspelled words. If you are using a corpus that was already prepared, it is important to consider what steps have been taken (if any) to regularize the text. See if you can find other versions of your texts that are more or less regularized to use in comparison. Regularization can be quite time-intensive but if you choose not to do it, you may find inconsistencies in your final word embedding model (for example, if the same word might be spelled several different ways).

Below are slides from the Word Vector Institute on data analysis and identifying what features might cause “noise” for your word embedding model:

- Data analysis questions (https://wwp.northeastern.edu/outreach/seminars/wem_2019-07/presentations/word_vectors/word_vectors_process_06.xhtml)
- Data regularization and correction (https://wwp.northeastern.edu/outreach/seminars/wem_2019-07/presentations/word_vectors/word_vectors_process_07.xhtml)

Stage 3: Data Preparation Plan

After completing data analysis for your corpus, the next stage is to create a plan for how you want to modify your original corpus before model training. This plan can take many forms, but the information that it should contain is: a) what textual features you will be removing or changing, b) what documents from within your corpus will be modified for each change, and c) how you will make these changes. This last piece of information, in particular, is important to document and, as you start implementing these changes, is likely to change.

As with the earlier recommendation for taking notes about your corpus, taking notes about the changes you make throughout data preparation is essential, especially if you make a change that you may want to revert. Whether or not you are using a form of version control for your corpus—either with tools like GitHub or by saving different copies on your own computer—documenting changes between corpora is essential. In the event that you want to go back to an earlier form of your corpus, having this documentation (both as a plan and in tracking any changes) is very helpful. For example, here is a brief outline of a generic data preparation plan from the Word Vectors institute this summer:

Corpus Level

- ❏ Data transformation (as needed)
    - ❏ XQuery (date, XQuery file used, parameters set)
- ❏ Restructuring (create a new version with original saved as-is)
    - ❏ Organization statement (by genre, by chronology, by theme, etc.)
    - ❏ New file structure/directory
    - ❏ Folder or file naming conventions

File/Document Level

- ❏ Metadata
    - ❏ Date removed
    - ❏ Files impacted (if not entire corpus)
    - ❏ Content removed (titles, data licenses, copyright information, etc)
    - ❏ Method of removal/cleaning

- ❏ Structural Changes
    - ❏ Headings (chapters, books, sections, etc.)
        - ❏ Date removed, method of removal, content removed, and files impacted
    - ❏ Paratexts (table of contents, frontispieces, etc).
        - ❏ Date removed, method of removal, content removed, and files impacted
- ❏ Regularizations
    - ❏ Date changed
    - ❏ Files impacted
    - ❏ Content changed (words, spellings, organization structure, etc)
    - ❏ Method of regularization (Open Refine, regular expressions, by hand, etc)

Stage 4: Clean and Modify Corpus

After creating your data preparation plan, the next step is to put this plan into action and modify your corpus. This process can take shape in a variety of ways, depending on many factors: your experience with different tools, your research question and issues of interest, and the makeup of your corpus (content, format, and quantity). Indeed, there are many different tools and tutorials for data manipulation that are great for all different skill levels. For the Word Vectors institute, participant and sample corpora were cleaned using simple features of text editors like Oxygen, regular expressions, and XQueries. Here are some useful resources for data cleaning and preparation:

OpenRefine: http://openrefine.org/

A free tool for data manipulation supported by Google that allows for exploration, transformation, data matching, and manipulation. Here are tutorials and documentation on how to use it:

1. "Getting Started with OpenRefine" by Miriam Posner (for an undergraduate class)
2. "Getting Started with OpenRefine" by Thomas Padilla (includes use cases as well)
3. OpenRefine official documentation for users and developers

Regular Expressions

Regular expressions are helpful for finding, modifying, or removing repetition in data by describing a sequence of characters in a text or dataset. Many text editors will have an option to use regular expressions along with the "find all" or "find and replace" features. While there are a few different notations for regular expressions (it is helpful to check which is used in your preferred text editor), here are some generally useful introductory and intermediate sources:

1. "Understanding Regular Expressions" from *The Programming Historian* by Doug Knox
2. Regexone has many different interactive tutorials
3. *Regular Expressions Cookbook* by Jan Goyvaerts and Steven Levithan

In many of the corpora that were cleaned and processed for the Word Vectors institute, there were similar features that were removed or modified, including chapter headings, illustrations, and many more. When we used regular expressions to identify and remove these features, here is the process we used:

1. Use “Find All” in a text editor to note the different instances of the feature in question (i.e. “Chapter” could look like “CHAPTER 21”, “chapter 1,” “Chapter the Twenty-first,” “Chapter: 21”).
2. Construct a regular expression to include different variations of the feature in question so they can be removed all at once.
3. Without replacing anything, test the regular expression using “Find All”.
    a. Make sure that *all* results can be safely deleted.
    b. Check that instances found manually in #1 are also found by the regular expression.
    c. If needed, revise the regular expression and test again. It is more important to ensure that every match can be deleted than it is to match every possible variation in #1.
4. Ideally, make a copy of the corpus as a backup. Label it clearly.
5. Use the “Replace” feature to replace all matches with either nothing (an empty field) or a single space.
6. If the regular expression could not capture all variations found in #1, return to #2.

The following are some of the regular expressions that were frequently used in data preparation for the institute (pay attention to case sensitivity and spacing):

- **Chapters**: for different cases of capital letters or headings spelled out, see the following iterations of the simple regex → **Chapter \w+**
    - Different titles → **(chapter|book) \w+** (separate all choices with a |)
    - Numbers spelled out → **Chapter (the)?\w+(\.\w+|-\w+)?((ONE|TWO|THREE|FOUR|FIVE|SIX|SEVEN|EIGHT|NINE)\w*)?**
    - Headings on newline → **\nChapter \w+** or **\n(chapter|book) \w+**
- **Illustrations**: for illustration captions using square brackets → **\[Illustration([^\]]+)?\]**
- **Timestamps**: following the pattern 00:00 am or pm→ **\d\d:\d\d (\d\d)?**
- **Dates:** for month followed by year → **(January|February|March|April|May|June|July|August|September|October|November|December) \d{4}**
- **Speaker labels:** for speaker names which appear alone on a single line → **\n[A-Z\s-]+\n**
- **Roman numerals:** in chapter titles → (**[MDCLXVI]+$\W)**

A Note Regarding Underscores

As discussed during the institute, the word2Vec R package treats a corpus as a “bag of words.” During the model training process, punctuation is largely removed, but _underscores_ are an

exception. For any corpus with texts from Project Gutenberg or several other text transcription projects, underscores are frequently used to mark italics. We advise people to remove underscores because the model treats these words as distinct from the non-underscored version of the word, even if they are the same from a human reading perspective. However, when used deliberately, this behavior can also be quite useful. If there are words or phrases that you would like to be treated as a single token in the model, using an underscore to differentiate them (i.e. free_trade or queer_liberation) will let you explore phrases in a trained model.

Using this feature in your data manipulation stage is fairly easy. Using a “find and replace” feature, search for the words or phrases of interest and replace with the desired phrase with underscores in place of spaces. For more information, here is a helpful exploration by Kavita Ganesan: “How to incorporate phrases into Word2Vec-a text mining approach.”

Stage 5: Reflect and Repeat

The final stage of data preparation before training a word embedding model is simply to reflect on the process so far. Throughout our discussions at the Word Vector institute we noticed that, regardless of how much you document and plan, new issues with data preparation will arise as you are doing the work. Before you train your first (or twentieth) model, it is useful to think about how you have prepared the data for this step. Are there issues that you could not or did not address in this round of preparation? If so, what are these and why did you choose to leave them as is? Might you need to change them in the future?

Data preparation is iterative; it is tempting to try and make all the changes as once, but slowing the process down to observe, reflect, and explore the data and resulting word embedding model is an important step, especially at the beginning of a project. After training your first model, there will likely be new changes or ways to organize your data that you will be interested in exploring. Documenting your preparation process and reflecting on it will help as you move forward to testing your model and exploring the effect of training parameters.