Baseline definitions for text-analytics guides

In accompaniment of ***Towards a "shut-up-and-take-my-money!" Magic Wand for DIY Text Miners by Hab Elassaad Bader on Medium . For more background context, please read the companion article. Licensing: Released freely into the Public Domain in the name of knowledge. Do with it what thou wilt.***

A suitable companion document for exercises and workshops too!

Outcomes.

This is what I deem to be the "basic usual set of text-analytics taskflows" that are a catch all for typical average use-cases:

a. **pre-processing**	Getting text ready for analysis; explain to people they should get everything into one place if their ideas are all over the place, and file format changes to fit into the needed parameters. Cleaning it, spellchecks. Most people will be getting their archive off from something like *OneNote/Apple Notes* or *Evernote*, which may have hidden code that is interpreted as extra carriage returns if brought into a richtext word processor. And people may have used different delimiters for separating their ideas (e.g. blank lines, hyphened lists, asterisk bullets, etc.) for which the most common few should be guided how to homogenize. Extreme of this is OCRing handwritten cocktail-napkin doodles.
b. **overall topics**	identifying overall machine-guessed subject matters, ranked, from the whole dataset/corpus irrespective of concept-sections
c. **auto-tagging**	multiple tags assigned to each distinct idea-section (from the words in its paragraph, matched with similar tags from other idea-sections in the dataset). Idea/concept/sections could remain as groups with sub-tokens under each, or by now machine-split out into multiple documents, one for each
d. **entity-extraction**	of named generally known concepts. could be flagging locations, economic industries, popular people, technologies, listed companies/ brands from frequently appearing nouns, built-in lists pulled from Internet-based public databases like DBPedia/ Freebase/IAB/IPTC
e. **auto + manual classification**	A mix of machine-automated and human interactivity is key to have the ability for the beholder to fine-tune the accuracy and make new similar connections by re-examining the machine-clustered n-grams from a different textual dimension… i.e. through drag-and-drop re-grouping/re-sorting of ideas together by a shared tag or entity. graphically retrain the system without code
f. **visualization**	via 1 simple view or report (e.g. 2D/3D mindmap or tagcloud). ability to display this to illustrate their machine-interpreted ideas when shared with others, e.g. in a presentation, pitch, lecture, or brainstorm
g. **exporting and conversion**	of formats to not lose the richness of work down when re-opening and parsing in different end-user tools; springboard to more advanced specialized analysis later with different programs, migration, portability, and future-proofing

Inclusion of the above -nothing more, nothing less- constitutes what I believe should become chapters in a guide for new generalist computational-linguistics practitioners, or modules in the UML diagram of a complete text-mining application system for end users, given your average Tom-Dick-and-Harry's scenario.

Inputs.

It's pretty granted that for any instructional-steps in our field to be reliable, the original content has to follow a consistent repeated pattern that sticks to rules… in this case, always a parseable UTF-8 file (e.g. a .Txt in TextEdit for Mac or Notepad for Windows with no Markdown/HTML code… YET).

For sake of example—regardless if it has 10 or 1000s of lines, let's assume that the original source contains always follow these strict delineation conventions:

```
(Line 1) Concept 1 name (carriage-return/enter)
(Next line or lines) Description of the above concept 1
(TWO or MORE carriage-returns/enters in a row = at least 1
   blank line/empty space between the next. it's a separator)
(Next line or lines) Concept 2 name (carriage-return/enter)
(Next line or lines) Description of the above concept 2
   etc… continues to repeat…
```

When still in 1 master PKB repository file, the names (titles) of concepts — distinct groups of ideas — before splitting each into its own document in the overall corpus, could be said to be the NLP equivalent of a 'primary key' of a record in an RDBMS. Much later, at visualization stages, the original text of descriptors of ideas should move along with their concept name whichever way they are rearranged/regrouped, using their text as attributes. This is how spatial note applications work — the most likely final resting place of post-analysis ideas for non-data-scientists.

Of course, descriptions may go into several lines with auto-wrapping, or forced into several lines with carriage-returns/enter. And for a typical person, concepts often will be imperfectly markdown-ish, laying out subconcepts that are ordered (numbered) or unordered lists with hyphens… which may vary per concept as not all ideas require being expounded on as much as others. For the individual doing a rudimentary analysis of their jumbled thoughts or ones made by others they've compiled/collated, subconcepts don't really matter as far as NLP for idea-mapping is concerned… just the overall topics and the rearrangement of higher-level distinct concepts (subconcepts will be towed along with their respective main concept at the clustering/visualization stages).

Conditions.

- *All 3rd party runtime command-line tools, GUI apps with plugins, or code-compilers to complete the taskflow must be either open-source, or proprietary but free or affordable (e.g. available to a single-user licensed at <$100). Any white-collar who can afford a computer in an impoverished nation should be able to muster this much up.*

- *It's OK if logic-calls are sent to a web service or inputs pulled from public databases (e.g. taxonomies for common entity-extraction or classification), but no raw or n-grammed portions of the actual end-user's text undergoing analysis can go out onto the Internet, where it can open up a can of worms on copyright-safeguards, even if encrypted or private. I.e. no SAAS text analysis web-services nor the try-me demos on their websites. We're talking wizard-installs on a MacOS or Windows desktop. Many commercial solutions today do offer this, but most do so only for large organizations.*

In this ubiquitously-digitally-converged world full of everyday-researchers who all need basic data-science literacy— in a decade that rests on actual data scientists helping pseudo-meta-self-scientists (not only brands doing social-listening/sentiment analysis of feedback and reviews or big pharmas scanning white-paper literature reviews to surface offlabel drug uses), you can't factor in expensive, prebuilt, 360' commercial solutions that miraculously do it for you.

And aspiring researchers (who are underfunded, often working on confidentially-sensitive ground-breaking works where it's career-changing to get in the journal first), as well as for individuals at home wanting to process their life diaries, they have information-privacy hesitations to hosted tools