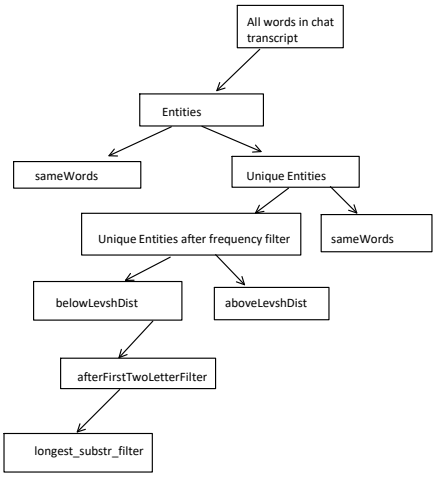
SpellCheck Output Writeup

Sunday, August 4, 2019 2:10 PM

Structure of data from Levana's dataset (testSpellError.csv)

Hierarchy of Information:



	Definition
Entities	Can be one of the following - Action: ○ Handled by testSpellingErrorAct.py file ○ Fourth column from the dataset Levana sent ○ Compared against the action entity library - General Entity ○ Handled by testSpellingErrorProd.py file ○ Fifth column from the dataset Levana sent ○ All entities in fifth column with confidence "middle" ○ Compared against the general entity library - Product Entity ○ Handled by testSpellingErrorProd.py file ○ Fifth column from the dataset Levana sent ○ All entities in fifth column with confidence "high" ○ Compared against the product entity library; compares against both entities and synonyms to see if there is a match
aboveLevshDist	All **Entities** with a Levenshtein distance higher than the Levenshtein threshold compared to ANY of the entity library words - Informative to look through to see which ones to add to the library because these are the entities that were different from any of the current library words that they could be completely different terms
Unique Entities	Subset of **belowLevshDist** that all have Levenshtein distance > 0 compared to library word (i.e, filters out all identified library words) - These are instances, so can be repetitions of the same unique entity word/phrase - Metric for how representative the library words are; if number unique entities is significantly less than total entities, that means library has strong coverage of the identified entities
Unique entities after frequency filter	All unique entities after filtering out the most frequent occurrences of the same entity words - Frequency analysis handled in FreqDist.py file; removes words that occur greater than or equal to 2 (or 3 depending on entity) times in the entire dataset. Rationale: The more frequent that the same word (same ordering of same set of letters) occurs, the more unlikely it is to be a spelling error because multiple users have applied that same spelling
belowLevshDist	All **unique entities after frequency filter** with a Levenshtein distance <= the Levenshtein threshold compared to AT LEAST ONE entity library word - List that matches each instance of an entity with an entity library word that meets the above requirement
sameWords	Subset of **Unique Entities** that have Levenshtein distance of 0 with an action entity library word (in other words, an action entity library word has been identified in the chat transcript)
afterFirstTwoLetterFilter	Subset of **belowLevshDist**; must contain the same first letter as the action entity library word matched to
longest_substr_filter	Subset of **afterFirstTwoLetterFilter**. Final filter in the process that requires matches to have substring of at least *x* characters in common, where x depends on the number of letters in the string of the identified entity - Output: *[entity with potential spelling, a potential match from the entity library based on similarity requirements above]*

SpellCheck Output Key Highlights:

	Number of identified entities in dataset	Number of Unique entities (instances)	Number of unique entities after frequency filter	After first two letter filters	After final filter (longest substring filter)	Entities above Levenshtein distance
Product Entities	30,402	1,134	7	[['window hello', 'windows hello'], ['window hello', 'windows hello'], ['Windows IoT Core', 'windows 10 iot core'], ['Windows IoT Core', 'windows 10 iot core'], ['Windows IoT Core', 'windows 10 iot core'], ['Windows IoT Core', 'windows 10 iot core'], ['window hello', 'windows hello'], ['window hello', 'windows hello'], ['window PC', 'windows pc'], ['window PC', 'windows pc']]	[['window hello', 'windows hello'], ['Windows IoT Core', 'windows 10 iot core'], ['Windows IoT Core', 'windows 10 iot core'], ['window hello', 'windows hello'], ['window PC', 'windows pc']]	[['window hello', 'windows hello'], ['Windows IoT Core', 'windows 10 iot core'], ['Windows IoT Core', 'windows 10 iot core'], ['window hello', 'windows hello'], ['window PC', 'windows pc']]
General Entities	29,120	10,079	3,781	[['account id', 'accounting'], ['account id', 'accounting'], ['subscription id', 'subscription'], ['subscription id', 'subscriptions'], ['transcript id', 'transcripts']]	[['account id', 'accounting'], ['account id', 'accounting'], ['subscription id', 'subscription'], ['subscription id', 'subscriptions'], ['transcript id', 'transcripts']]	See separate document
****Actions**	59,523	*(after frequency elimination)* 63	*(after same words filter)* 23	[['eliminate', 'eliminated'], ['actiave', 'achieve'], ['actiave', 'activate'], ['actiave', 'activate'], ['assiociate', 'associate'], ['assiociate', 'associate'], ['assiociate', 'associated'], ['assiociate', 'associate'], ['assiociate', 'associate'], ['assiociate', 'associated'], ['genrate', 'generate'], ['seperate', 'separate'], ['seperate', 'separate'], ['suppot', 'support'], ['acced', 'accept'], ['acced', 'access'], ['craeate', 'create'], ['craeate', 'created'], ['suppot', 'support'], ['clash', 'clashes'], ['lapse', 'lapses']]	[['eliminate', 'eliminated'], ['actiave', 'activate'], ['actiave', 'activate'], ['assiociate', 'associate'], ['assiociate', 'associate'], ['assiociate', 'associated'], ['assiociate', 'associate'], ['assiociate', 'associate'], ['assiociate', 'associated'], ['genrate', 'generate'], ['seperate', 'separate'], ['seperate', 'separate'], ['suppot', 'support'], ['acced', 'accept'], ['acced', 'access'], ['craeate', 'create'], ['craeate', 'created'], ['suppot', 'support'], ['clash', 'clashes'], ['lapse', 'lapses'], ['earnrenew', 'earn'], ['earnrenew', 'renew']]	['action', 'dissasscoiation', 'tear'] ***These are spelling errors that are not being caught by the algorithm, perhaps because it still differs from the vocab list*

***filtered out most common actions first, then library word occurrences mainly for performance and efficiency reasons*

- *Given proportion of action entity instances and actual unique action entities,* would have been inefficient to compare every instance against library word first
- *Also want to compare the lemmatized version of the action to the library words, would take way too long to lemmatize 59,523 words instead of 63*

Frequency Distribution of different entities:

General Entity Frequency Analysis:
- FreqDist with 3754 samples and 29120 outcomes (so 3654 unique general entities out of 29120 total instances of general entities identified)

Product Entity Frequency Anlaysis:
- FreqDist with 271 samples and 30402 outcomes (so 271 unique product entities out of 30402 total instances of product entities identified)

Action Entity Frequency Analysis:
- <FreqDist with 319 samples and 59523 outcomes> (so 391 unique action entities out of 59523 total instances of action entities identified)

However, don't know the frequency distribution

Analysis of above results:

Product Entities:
- The fact that the majority of the identified entities in the dataset were eliminated after searching for identical library words indicates the current state of the product entity library is highly representative of the product entities identified from a given user chat transcript.
- Only 7 unique entities after eliminating the product entities with the most frequent occurrences, showing there is less likely to be spelling errors among the product entities. 203 out of the 271 samples (unique product entities) occurred more than once (i.e, multiple occurrences of the same ordering of the same set of letters); only 68 samples had a singular frequency in the dataset
- Final filter after the longest substring evaluation shows that filtering process is effective in eliminating noisy data and outputting results that would actually be potential spell errors

General Entities:

- 34% of the total entities identified in dataset are instances of unique entities (not in the library). Around 37% of the unique entities remain after eliminating the most frequent ones. Shows decent coverage of the general entity library in its current state but could be improved with more manual review
- Most are eliminated after the spell check filtering process as there are only 2 out of 3,781 total entities being evaluated for spell check that were identified to be similar enough to any of the library words. Indicates either that library is not representative enough of actual general entities, or that there are rarely ever spelling errors

Action Entities:

- Likely to hold highest accuracy rate in detecting actual spelling errors, given that majority of final output are actual misspelled words (whereas the output in product or general entities were correctly spelled words that were just similar *enough* in distance to some entity library word to be flagged).
- Failed to catch the spelling errors in aboveLevshDist however