The following are updates made from *7v* to 8v

1. Rewrite the emoji module – now it uses counter and regex to process the emoji to avoid loop.
2. Add multi-thread using concurrent.feature this package to do input parallelization;
3. Turn TPL\_Int\_Score.py into two classes, with the first called Prep involving all the file imports and only run once to save computational cost;

The following are updates made from *6v* to 7v

1. Turn intst into raw count, that includes - vq\_intst\_tempo, vq\_intst\_silence, tk\_intst\_tactileemojis, tk\_intst\_bodilyemoticons, vk\_intst\_bodilyemojis, vk\_intst\_bodilyemoticons, a\_intst\_nonbodilyemojis, a\_intst\_nonbodilyemoticons all of them used to count the mere repetition but then divided by 3 (in 6v), now we let them keep the raw count of repetition and hence remove dividing them by 3 (e.g., 👍👍👍👍👍, was counted as 5/3 = 1.6666, now 5, also notice that the basic unit in emphasis is three repetition (i.e., wowowow has to appear three times), hence we did not change vq\_intst\_emphasis)
2. We modified check\_emphasis function: 1. For the third case, where one has to add the first letter to form the basic unit of repetition (e.g., wowowow – detect repetition of ‘wo’) to be consistent with the previous cases
3. Update the dictionary, in 6v all the dictionaries are updated by Jan 16, 2020. We update all the dictionaries to Jan 5, 2021, which include excluding some keywords, e.g., Alternant – lol, lmao, including new emojis
4. Excluded Acronyms.xlsx contains the same content as Doppleganger Dictionary, except that I did not rename it as Doppleganger Dictionary
5. ~~We updated all the dictionaries except Emoji Dictionary.xlsx, the updated Emoji Dictionary in is in picture format, that is not the format we are able to edit and process in the classifier. We need the same format that is being used by the column Browser in the table of this~~[~~webpage~~](https://unicode.org/emoji/charts/full-emoji-list.html)~~, which is in the format of Unicode (notice that although conceptually, the column Browser contains the same Unicode as the column code in that table, the column 'code' is not directly readable by python because there are some small tweaks.)~~
6. In the function Detect\_Asterisk\_TPL, sub function check\_asterisk, we allow VK asterisk symbols to count anything surrounded by \* \* as well as :: :: as VK except keywords, however, the codes in 6v were wrong:

* Wrong code: nvk\_count = nvk\_count + len([" ".join(word) in non\_key\_word\_dic]),

This line of code would always return 1 because len([" ".join(word) in non\_key\_word\_dic]) is always 1 (len[False])

* Right code that we changed into:

join\_word = [" ".join(word)]

nvk\_count = nvk\_count + len([w for w in join\_word if w in non\_key\_word\_dic])

1. We include 🎶 in detecting VQ volume for Detect\_Asterisk\_TPL function, on top of symbol ♫
2. Since x and xx has moved from TK emoticon to TK alpha-haptic, we simplify the coding of the emoticon module by removing the lines of codes in check\_emoticon.py that count x and xx
3. The folder TPL\_support\_file is now in sync on google drive Berkeley account, and shared with Andrea.
4. One can find the 7th version at /Users/yixiang-xu/Research/TPL/code as well as the new computer cluster - /home/yixiang\_xu\_berkeley\_edu/research/TPL/
5. Detect\_Asterisk\_TPL remove ' ', and " "
6. VQ Silence module (which counts element of VQ tempo, e.g., …, \_\_\_) has been updated: a. to avoid counting those silence note “…” before http:, https:, more for, because they seem more an indicator of incomplete sentence then silence, we added the following: re.sub(r"\.\.\.(\s){0,}http:", r" http:", textline), re.sub(r"\.\.\.(\s){0,}https:", r" https:", textline), re.sub(r"\.\.\.(\s){0,}more for", r" more for", textline)

Secondly, the module is updated such that elements containing symbols #, @, RT, http, https: would not be counted

1. VQ rhythm versus spelling – the codes are perfect, however, we made update on comments to both modules this time, for rhythm, it only captures {word}{symbol}, the unit is not letter but word (checked against brown dictionary), whereas spelling only captures {letter}{symbol}, the unit is letter
2. for VQ spelling, check against the doppelganger dictionary, i.e., Detect\_Spelling(textline,word\_list,brown\_dic,acron\_list), when it used to be

Detect\_Spelling(textline,word\_list,brown\_dic), to avoid cases like R.I.P. be detected

1. for VQ censorship each of the case pattern, we added " " in front of the regular expression, to ensure that the detected item is not part of some word, e.g., we want a's -- ass, but not Tiana's. We also add " " in front of the textline, i.e., textline\_kw = " " + textline + " ", so that when the curse word is the first word, hence no " " would in front of it, it will still be able to be detected
2. for Detect\_Emojicon\_TPL, we edited the function block\_emoticon, where we added two restrictions: detected emoticon if starting with digit (i.e., 0-9) or symbol “:” or symbol “=”, or ending with digit (i.e., 0-9) or symbol “:” or symbol “=”, the item before or after could not be digit or letter, this help rule out fake emoticon cases like -- 1:30, 20:02, 3=3, D=3

we also detected some errors in the codes of block\_emoticon, in particular, in order to detect whether the beginning or ending letter obey the restriction rule:

add line 89: k\_hat = re.sub(r'\\','',k\_hat)

line 68, 84, 114: now - if mt\_span[i][0]-1>=0: before - if mt\_span[i][0]-1>0:

1. remove x from the dictionary
2. (Not directly related to change in the classifier)

to run task in cluster, the following updates were made to the executation python scripts:

Update all codes for spelling by adding acron\_list to function VQ spelling

Cluster update from yogi or research.haas to the new cluster

6v to 7v

The following are updates made from 5v to 6v

1. There is a mistake in the codes of 5v. In the TPL\_Int\_Score.py code, the following line is incorrect -

a\_nonbodilyemoji = a\_nonbodilyemoticons + a\_emoji\_symbol

Though this has been fixed in 5v already, it is after our submission to JMR Dec 2019.

1. Add the following into the emoticon lists -- ;P : )
2. Update the dictionary to include new keywords without symbols such as “ay” (in total 6 new keywords without symbols, ding, ay, yoink, yup, ole, nom)
3. Update the dictionary to include new Dopplegangers including in total 30 new (ME, A&E, BBC, E!, FX, CNN, FCUK, CVS, AT&T, 3M, H&R, TCBY, DSW, AFLAC, GEICO, LEGO, KFC, FIAT, SAP, GE, AMC, AOL, ASICS, BIC, BP, IGN, ITC, RCA, UPS, YKK)
4. Update the VK asterisk symbols to count anything surrounded by :: :: as VK except keywords
5. Update the VQ silence to allow for two periods in a row “..” which used to be three periods. But still if the two periods “..” happens at the end of a sentence, it would not be counted.
6. Change the count of tempo, now for the pure counts of tempo, no repeated count within a word for two different letter elongations, e.g. Greaaatttt will only be counted once, no change in how to calculate its intensity score though.
7. Count {word}{period}{word}{period}{word}{exclamation mark} in VQ rhythm
8. Add a TPL raw counts
9. VQ emphasis, repetition two-unit for keywords and three-unit for other English words. It used to be repetition two-unit for all words. (still working on)
10. \*high five back\* used to counted twice, once by TK, once by VK, because VK module is not good at detecting phrase. Now this one would only be counted as TK.
11. Include “\_\_” in VQ silence as well.
12. Remove “~~” from the emoticon list so it would only be count by VQ silence and avoid repeated counting.
13. Fix a code mistake in VQ silence to give rise to accurate measure of intensity score: [max(1,len(w[0])/3) for w in vq\_matched] should be [max(1,len(w)/3) for w in vq\_matched]
14. Update the dictionary of keywords with symbols by eliminating the following ding, speak, talk, voice, chat, say, tell accent, rant, face, eyebrow, nose, eye, lip, mouth, blush, cheek; by the way, blink is maintained though Andrea suggested removing as well.