```python
# -*- coding:utf-8 -*-
import re, datetime, csv , jaconv, spacy
class Ext:
    def __init__(self,rawText):
        self.rawText = rawText
        self.datas = {"dates": "", "times": "", "places": ""}
        self.nlp = spacy.load("ja_ginza")

    def NotationVariability(self,rawText):
        dt_now = datetime.datetime.now()
        text = jaconv.z2h(rawText, kana=False, digit=True, ascii=True)
        with open("dict/NotationVariability.csv",encoding="UTF-8") as f:
            for row in csv.reader(f):
                for i in range(1, len(row)):
                    text = text.replace(row[i], row[0])
        text = re.sub("明日|あした|あす",f"{dt_now.month}/{dt_now.day+1}",text)
        text = re.sub("明後日|あさって|みょうごにち", f"{dt_now.month}/{dt_now.day+2}",
text)
        return text

    def DateCheck(self, text):
        pattern1 = r"([0-9]{1,2}月[0-9]{1,2}日)|([0-9]{1,2}/[0-9]{1,2})|([0-9]{1,2}
日)"
        pattern2 = r"[0-9]{1,2}時[0-9]{0,2}分?"
        dates = re.findall(pattern1, text)
        times = re.findall(pattern2, text)
        return dates,times

    def PlaceCheck(self, text):
        places = []
        doc = self.nlp(text)
        for ent in doc.ents:
            if ent.label_ == "City" or ent.label_ == "Province" or ent.label_ ==
"Road":
                places.append(ent.text)
            #print(ent.text,ent.label_)
        return places

    def Extract(self):
        textNotate = self.NotationVariability(self.rawText)
        dates, times = self.DateCheck(textNotate)
        places = self.PlaceCheck(textNotate)
        return dates,times,places
```