In [7]:

```
!wget -O data.tar.gz https://dataverse.harvard.edu/api/access/datafile/:persistentId?pe
rsistentId=doi:10.7910/DVN/6MZN76/CRUNF0
!tar -xf data.tar.gz
```

```
--2022-03-24 17:55:47--  https://dataverse.harvard.edu/api/access/datafil
e/:persistentId?persistentId=doi:10.7910/DVN/6MZN76/CRUNF0
Resolving dataverse.harvard.edu (dataverse.harvard.edu)... 35.170.94.248,
34.237.225.150, 35.168.250.191
Connecting to dataverse.harvard.edu (dataverse.harvard.edu)|35.170.94.248
|:443... connected.
HTTP request sent, awaiting response... 303 See Other
Location: https://dvn-cloud.s3.amazonaws.com/10.7910/DVN/6MZN76/15de5b930d
d-0fada3dbae00?response-content-disposition=attachment%3B%20filename%2A%3D
UTF-8%27%27Dail_debates_1919-2013.tar.gz&response-content-type=applicatio
n%2Fx-gzip&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20220324T175548Z&X-
Amz-SignedHeaders=host&X-Amz-Expires=3600&X-Amz-Credential=AKIAIEJ3NV7UYCS
RJC7A%2F20220324%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Signature=510d95691
76880c05f6541641c70e6f36174e36358919484d1603785039f3df7 [following]
--2022-03-24 17:55:48--  https://dvn-cloud.s3.amazonaws.com/10.7910/DVN/6M
ZN76/15de5b930dd-0fada3dbae00?response-content-disposition=attachment%3B%2
0filename%2A%3DUTF-8%27%27Dail_debates_1919-2013.tar.gz&response-content-t
ype=application%2Fx-gzip&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20220
324T175548Z&X-Amz-SignedHeaders=host&X-Amz-Expires=3600&X-Amz-Credential=A
KIAIEJ3NV7UYCSRJC7A%2F20220324%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Signa
ture=510d9569176880c05f6541641c70e6f36174e36358919484d1603785039f3df7
Resolving dvn-cloud.s3.amazonaws.com (dvn-cloud.s3.amazonaws.com)... 52.21
7.169.17
Connecting to dvn-cloud.s3.amazonaws.com (dvn-cloud.s3.amazonaws.com)|52.2
17.169.17|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 959382206 (915M) [application/x-gzip]
Saving to: ‘data.tar.gz’

data.tar.gz         100%[===================>] 914.94M  41.7MB/s    in 21s

2022-03-24 17:56:09 (43.8 MB/s) - ‘data.tar.gz’ saved [959382206/95938220
6]
```

In [3]:

```
import spacy
import pandas as pd
from tqdm.auto import tqdm

pd.options.plotting.backend = "plotly"
```

In [4]:

```
en = spacy.load("en_core_web_sm") # loading spacy model
```

In [5]:

```
import neuralcoref
from spacy.tokens import Token

Token.set_extension("corefs", default=[])
coref = neuralcoref.NeuralCoref(en.vocab)
en.add_pipe(coref, name="neuralcoref")
```

```
100%|██████████| 40155833/40155833 [00:00<00:00, 47927419.56B/s]
```

In [9]:

```
df = pd.read_table('Dail_debates_1919-2013.tab')
```

In [12]:

```
df.date = pd.to_datetime(df.date)
df2 = df[df.date.dt.year>1999]
df3 = df2.sample(frac=.002, random_state=123)
docs = df3['speech'].apply(en)
```

In [14]:

```
DEONTIC_VERBS = ["can", "could", "may", "might", "must", "shall", "should"]
```

In [55]:

```
def get_deontic(sent):
  for token in sent:
      if token.lemma_ in DEONTIC_VERBS:
          return token
  return None


def get_verb(deontic):
  return deontic.head if deontic is not None else ""


def get_child_dep(verb, dep):
  return [child for child in verb.children if child.dep_ == dep]


def get_subjects_clausual(verb):
  # [y for x in [a, b] for y in x]
  return [ch for child in get_child_dep(verb, "csubj") for ch in get_child_dep(child,
"nsubj")]

def get_coref(token):
  corefs = token._.corefs
  if len(corefs) == 0 or token.pos_ != "PRON":
      return ""
  return corefs[0]
```

In [209]:

```
def ig_sent_tag(sent):
  deontic = get_deontic(sent)
  attributes, objects, verbs = [], [], []
  verb = get_verb(deontic)

  while verb:
    attr = verb
    verbs.append(verb)
    verb = None

    subject = get_child_dep(attr, "nsubj")
    passive_subject = get_child_dep(attr, "nsubjpass")

    if len(subject) == len(passive_subject) == 0:
        attributes = get_subjects_clausual(attr)
    attributes.extend(subject)
    objects.extend(passive_subject)
    if attr.dep_ == "conf" and attr.pos_ == "VERB":
      verb = attr.head
    last_subj = None
    for subj in attributes:
      if last_subj == subj:
        continue
      if subj.dep_ == "conj":
        attributes.append(subj)
      attributes.extend(get_child_dep(subj, "conj"))
      if subj.pos_ == "PRON":
        subj.pos_ = get_coref(subj)
      last_subj = subj

    for obj in objects:
      objects.extend(get_child_dep(obj, "conj"))
  return {
    "deontic": deontic.lemma_ if deontic else "",
    "attributes": attributes,
    "objects": objects,
    "verbs": verbs
}
```

In [236]:

```
def ig_tagging(docs):
  return pd.DataFrame([{**ig_sent_tag(sent), **{"doc_id": i, "party_name": df3.loc[i].p
arty_name}} for i, doc in docs.iteritems() for sent in list(doc.sents) if get_deontic(s
ent)])
```

In [237]:

```
igt = ig_tagging(docs.iloc[:500])
```

In [238]:

```
igt
```

Out[238]:

	deontic	attributes	objects	verbs	doc_id	party_name
0	can	[person]	[]	[obtain]	3473408	Progressive Democrats
1	may	[married]	[]	[accept]	3473408	Progressive Democrats
2	must	[marriage]	[]	[subsisting]	3473408	Progressive Democrats
3	must	[Persons]	[]	[apply]	3473408	Progressive Democrats
4	may	[Minister]	[]	[grant]	3473408	Progressive Democrats
...	...	...	...	...	...	...
350	may	[]	[shellfish]	[marketed]	3880130	Fianna Fáil
351	must	[school]	[]	[ensure]	3653165	Fianna Fáil
352	must	[it]	[]	[act]	3653165	Fianna Fáil
353	can	[]	[identifies]	[used]	3653165	Fianna Fáil
354	may	[parents]	[]	[appeal]	3653165	Fianna Fáil

355 rows × 6 columns

In [300]:

```
pivot = igt.pivot_table(index=['party_name'], columns='deontic', aggfunc='size', fill_v
alue=0)
```

In [301]:

```
col_names = list(pivot.columns)
# sum deonties by parties
pivot['sum'] = pivot[list(pivot.columns)].sum(axis=1)
# sum deonties by type
rows_sum = pivot.sum(axis=0)
rows_sum.name = "all parties"
pivot = pivot.append(rows_sum)
```

In [302]:

```
pivot
```

Out[302]:

deontic party_name	can	could	may	must	shall	should	sum
Democratic Left	0	0	0	1	0	0	1
Fianna Fáil	35	13	26	27	2	33	136
Fine Gael	31	9	25	7	0	19	91
Green Party	6	0	1	3	0	5	15
Independent	3	1	0	5	0	2	11
People Before Profit Alliance	0	0	0	0	0	1	1
Progressive Democrats	1	0	5	5	0	2	13
Sinn Féin	7	5	7	17	0	22	58
Socialist Party	0	1	0	0	0	2	3
The Labour Party	4	3	5	7	0	4	23
The Workers' Party	0	1	2	0	0	0	3
all parties	87	33	71	72	2	90	355

In [303]:

```
# convert numbers to percentage
for col_name in col_names:
  pivot[col_name] = pivot[col_name]/pivot['sum']
pivot = pivot.drop("sum", axis=1)
```

In [304]:

```
pivot.plot(kind='bar')
```