

NER Training does not work when using BILOU tagging #665

New issue

 Closed

brianrusso opened this issue on Dec 1, 2016 · 6 comments



brianrusso commented on Dec 1, 2016 • edited

NER training is not working per document/tutorials.

Specifically, offsets *do* appear to work. entity labels *do not* appear to work. Also the documentation is in conflict with itself which confuses the situation.

Using the entity-label NER [training example](#)

e.g.

```
nlp = spacy.load('en')
doc = Doc(nlp.vocab, [u'rats', u'make', u'good', u'pets'])
gold = GoldParse(doc, [u'U-ANIMAL', u'O', u'O', u'O'])
ner = EntityRecognizer(nlp.vocab, entity_types=['ANIMAL'])
ner.update(doc, gold)
```

As far as I know my syntax is correct. This doesn't work either:

```
doc = Doc(nlp.vocab, words=["The", "Law", "and", "Justice", "party", "is", "growing", "in",
"Poland"])
gold = GoldParse(doc, ['O', 'B-ORG', 'I-ORG', 'I-ORG', 'L-ORG', 'O', 'O', 'O', 'GPE'])
ner.update(doc, gold)
```

I get the following error:

TypeError Traceback (most recent call last)

in ()

----> 1 ner.update(doc, gold)

```
/usr/local/lib/python3.5/dist-packages/spacy/syntax/parser.cpython-35m-x86_64-linux-gnu.so in
spacy.syntax.parser.Parser.update (spacy/syntax/parser.cpp:7788)()
```

```
/usr/local/lib/python3.5/dist-packages/spacy/syntax/ner.cpython-35m-x86_64-linux-gnu.so in
spacy.syntax.ner.BiluoPushDown.preprocess_gold (spacy/syntax/ner.cpp:4782)()
```

```
/usr/local/lib/python3.5/dist-packages/spacy/syntax/ner.cpython-35m-x86_64-linux-gnu.so in
spacy.syntax.ner.BiluoPushDown.lookup_transition (spacy/syntax/ner.cpp:5145)()
```

TypeError: argument of type 'NoneType' is not iterable

I think this is a bug in GoldParse since offsets appear to work.

e.g.

```
ner = EntityRecognizer(nlp.vocab, entity_types=['ORG'])
doc = nlp.make_doc('The Law and Justice party is growing')
gold = GoldParse(doc, entities=[(4,25,'ORG')])
ner.update(doc, gold)
ner(doc)
print(doc.ents[0].text,doc.ents[0].label_)
--> Law and Justice party ORG
```

Assignees

No one assigned

Labels

bug

docs

Projects

None yet

Milestone

Improve training API

Notifications

5 participants



Also, the documentation is very inconsistent/confusing right now.

Conflicting examples:

1. <https://spacy.io/docs/usage/training>
2. <https://spacy.io/docs/usage/entity-recognition>
3. https://github.com/explosion/spaCy/blob/master/examples/training/train_ner.py

example 1 does not work

example 2 works for token offsets, does not work for token-level entity annotation.

example 3 is linked from 1 (as the 'full example'), and they are totally different examples.

Your Environment

Ubuntu

Python 3.5.2

1.2, latest PIP



h1man5hu commented on Dec 1, 2016

Same here. I get this error:

```
File "temp.py", line 18, in <module>
    tagger.update(doc, gold)

File "spacy/tagger.pyx", line 253, in spacy.tagger.Tagger.update (spacy/tagger.cpp:6803)
    self.vocab.morphology.assign_tag(&tokens.c[i], eg.guess)

File "spacy/morphology.pyx", line 41, in spacy.morphology.Morphology.assign_tag
(spacy/morphology.cpp:3943)
    tag_id = self.reverse_index[tag]

KeyError: 0
```


when executing this:


```
pos_tag_map = {
    'N': {"pos": "NOUN"},
    'V': {"pos": "VERB"},
    'J': {"pos": "ADJ"}
}
vocab = Vocab(tag_map=pos_tag_map)
tagger = Tagger(vocab)
train_words = ["I", "like", "green", "eggs"]
doc = Doc(vocab, words=train_words)
train_tags = ["N", "V", "J", "N"]
gold = GoldParse(doc, tags=train_tags)
tagger.update(doc, gold)
```

I've traced the value being passed to `tag_id = self.reverse_index[tag]` and it's coming from `linalg` in `thinc`

```
@staticmethod
cdef inline int arg_max_if_true(
    const weight_t* scores, const int* is_valid, const int n_classes) nogil:
    cdef int i
    cdef int best = 0
    cdef weight_t mode = -900000
    for i in range(n_classes):
        if is_valid[i] and scores[i] > mode:
            mode = scores[i]
            best = i
    return best
```


Now this returns an integer, not a string which `tag_id = self.reverse_index[tag]` is expecting.

 honnibal added `bug` `docs` labels on Dec 4, 2016

 honnibal referenced this issue on Jan 27

SpaCy NER training example from version 1.5.0 doesn't work in 1.6.0 #773

 Closed

 ines added this to the **Improve training API** milestone on Feb 18



honnibal commented on Apr 17


Owner

Fixed in v1.8.0 🐛

```
>>> nlp = spacy.load('en')
>>> from spacy.gold import GoldParse
>>> doc = nlp.make_doc(u'Facebook is a company')
>>> nlp.tagger(doc)
>>> gold = GoldParse(doc, entities=['U-ORG', 'O', 'O', 'O'])
>>> [t for t in gold.ner]
['U-ORG', 'O', 'O', 'O']
>>> nlp.entity.update(doc, gold)
1.0
>>> nlp.entity.update(doc, gold)
1.0
>>> nlp.entity.update(doc, gold)
0.0
>>> nlp.entity(doc)
>>> for ent in doc:
...     print(ent.text, ent.label_)
Facebook ORG
...
```



3

 honnibal closed this on Apr 17

 This was referenced on May 20

French using new Spacy's language model RasaHQ/rasa_nlu#376

 Closed

AttributeError: 'spacy.tokens.token.Token' object has no attribute 'label_' #1131

 Open



yogeshhk commented 14 days ago

I am using spacy 1.9.0 for updating 'en' model with my own tag.

Here is a code snippet:

```
model_name = 'en'
entity_label = 'U-CATCHPHRASE'
output_directory = './data/spacy_ner.model'

def train_ner(nlp, train_data, output_dir):
    # Add new words to vocab
    for raw_text, _ in train_data:
        doc = nlp.make_doc(raw_text)
        for word in doc:
            _ = nlp.vocab[word.orth]

    for itn in range(5):
        random.shuffle(train_data)
        print("NER Training iteration {}".format(itn))
        for raw_text, entity_tags in train_data:
            print("NER Training raw text {}".format(raw_text))
            print("NER Training tags {}".format(entity_tags))

            doc = nlp.make_doc(raw_text)
```

```

gold = GoldParse(doc, entities=entity_tags)
nlp.tagger(doc)
loss = nlp.entity.update(doc, gold)
nlp.end_training()
nlp.save_to_directory(output_dir)

```

Raw text is like : "view of the fact that the suit "

Entity tags are: ['O', 'O', 'O', 'U-CATCHPHRASE', 'O', 'U-CATCHPHRASE', 'U-CATCHPHRASE']

Getting error:

Traceback (most recent call last):

```

File "spacy/syntax/parser.pyx", line 320, in spacy.syntax.parser.Parser.update (spacy/syntax
/parser.cpp:10286)
  self.moves.preprocess_gold(gold)
File "spacy/syntax/ner.pyx", line 100, in spacy.syntax.ner.BiluoPushDown.preprocess_gold
(spacy/syntax/ner.cpp:4960)
  gold.c.ner[i] = self.lookup_transition(gold.ner[i])
File "spacy/syntax/ner.pyx", line 136, in spacy.syntax.ner.BiluoPushDown.lookup_transition
(spacy/syntax/ner.cpp:5653)
  raise KeyError(name)
KeyError: 'U-CATCHPHRASE'

```

What's wrong here? Any suggestions?



honnibal commented 14 days ago

Owner

Try `nlp.entity.add_label('CATCHPHRASE')` before training?



yogeshhk commented 14 days ago

Yes that helped. Thanks!!

So `add_label` takes the new NER tag word, but the training data needs to be provided with "U-" at the start, else there is error of not finding U- or B- at the start of the tags.

I ran with a few sample sentences and IOB tags, converted them to iob_to_biluo and that seems to work ok.

But when I ran with another training set, I ran into another problem (sorry to keep bugging you).

Now the new label is 'LEGAL' and here is the code:

```

model_name = 'en'
entity_label = 'LEGAL'
output_directory = './data/spacy_ner.model'

nlp = spacy.load(model_name)
nlp.entity.add_label(entity_label)
train_data = [(" ".join(text), tags) for text, tags in zip(X,y)]
ner = train_ner(nlp, train_data, output_directory)

def train_ner(nlp, train_data, output_dir):
    for raw_text, _ in train_data:
        doc = nlp.make_doc(raw_text)
        for word in doc:
            _ = nlp.vocab[word.orth]

    for itn in range(10):
        random.shuffle(train_data)
        for raw_text, entity_tags in train_data:
            doc = nlp.make_doc(raw_text)
            gold = GoldParse(doc, entities=entity_tags)
            nlp.tagger(doc)
            loss = nlp.entity.update(doc, gold)
nlp.end_training()
nlp.save_to_directory(output_dir)

```

and the error is:

Traceback (most recent call last):

```
gold = GoldParse(doc, entities=entity_tags)
File "spacy/gold.pyx", line 294, in spacy.gold.GoldParse.__init__ (spacy/gold.cpp:10834)
self.ner[i] = entities[gold_i]
IndexError: list index out of range
```

I checked that number of words in doc and tags in entity_tags are of same number. So there is one-to-one correspondence.

Then I saw that some of the tags were not converted by `job_to_biluoto` BILUO format. That could be a bug for further investigation. As a workaround, when I converted them manually like:

```
tags = [w.replace('B-LEGAL', 'U-' + entity_label) for w in tags]
tags = [w.replace('I-LEGAL', 'U-' + entity_label) for w in tags]
```

still the error of "index out of range" persisted. Can you guess?



yogeshhk commented 9 days ago

For experiment, I skipped the Index Exception, using Try block, and allowed non-error sentences to be passed for training.

The output was as below:

```
LEGAL consideration of other aspects
LEGAL consideration of other aspects
LEGAL qualification awarded by
LEGAL qualification is
LEGAL circumstantial evidence see
LEGAL circumstantial evidence its not
LEGAL circumstantial evidence the
LEGAL circumstantial evidence is
```

This obviously does not look very appropriate. Some extra tokens are appearing towards end. Am I missing something, say hyperparameters? Is it unidirectional or bidirectional RNN/LSTM that is being used? Or just that low volume of training data is causing it?