

```
corpus = [
    "The leaves on the tree have fallen.",
    "She is running towards the finish line.",
    "The cars are parked in the parking lot.",
    "A quick brown fox jumps over the lazy dog.",
    "He was walking in the park and saw a beautiful bird."
]
```

```
import nltk
from nltk.stem import PorterStemmer

nltk.download('punkt')

ps = PorterStemmer()

for sentence in corpus:
    words = nltk.word_tokenize(sentence)
    stemmed_words = [ps.stem(word) for word in words]
    print("Original:", sentence)
    print("Stemmed:", " ".join(stemmed_words))
    print()
```

```
Original: The leaves on the tree have fallen.
Stemmed: the leav on the tree have fallen .
```

```
Original: She is running towards the finish line.
Stemmed: she is run toward the finish line .
```

```
Original: The cars are parked in the parking lot.
Stemmed: the car are park in the park lot .
```

```
Original: A quick brown fox jumps over the lazy dog.
Stemmed: a quick brown fox jump over the lazi dog .
```

```
Original: He was walking in the park and saw a beautiful bird.
Stemmed: he wa walk in the park and saw a beauti bird .
```

```
[nltk_data] Downloading package punkt to C:\Users\Deep
[nltk_data] Salunkhe\AppData\Roaming\nltk_data...
[nltk_data] Package punkt is already up-to-date!
```

The PorterStemmer is a common stemming algorithm. It works by chopping off the ends of words in the hope of achieving the correct base form. However, it can sometimes produce non-words and might be too aggressive in some cases. Example: The word running would be stemmed to run, but the word better would be stemmed to bett.

```
from nltk.stem import WordNetLemmatizer
from nltk.corpus import wordnet
```

```

nltk.download('wordnet')
nltk.download('omw-1.4')

lemmatizer = WordNetLemmatizer()

for sentence in corpus:
    words = nltk.word_tokenize(sentence)
    lemmatized_words = [lemmatizer.lemmatize(word, pos=wordnet.VERB)
    for word in words]
    print("Original:", sentence)
    print("Lemmatized:", " ".join(lemmatized_words))
    print()

```

```

[nltk_data] Downloading package wordnet to C:\Users\Deep
[nltk_data] Salunkhe\AppData\Roaming\nltk_data...
[nltk_data] Package wordnet is already up-to-date!
[nltk_data] Downloading package omw-1.4 to C:\Users\Deep
[nltk_data] Salunkhe\AppData\Roaming\nltk_data...

```

Original: The leaves on the tree have fallen.  
Lemmatized: The leave on the tree have fall .

Original: She is running towards the finish line.  
Lemmatized: She be run towards the finish line .

Original: The cars are parked in the parking lot.  
Lemmatized: The cars be park in the park lot .

Original: A quick brown fox jumps over the lazy dog.  
Lemmatized: A quick brown fox jump over the lazy dog .

Original: He was walking in the park and saw a beautiful bird.  
Lemmatized: He be walk in the park and saw a beautiful bird .

The WordNetLemmatizer uses WordNet's built-in lexical database to find the base (lemma) of a word, considering its Part of Speech (POS). This approach is generally more accurate than stemming since it results in actual words, but it requires the correct POS tagging. Example: running is lemmatized to run (verb form), and better would stay better as it's already in its lemma form.

```

import spacy
spacy.cli.download("en_core_web_sm")

# Load English tokenizer, POS tagger, parser, NER, and word vectors
nlp = spacy.load("en_core_web_sm")

for sentence in corpus:

```

```

doc = nlp(sentence)
lemmatized_words = [token.lemma_ for token in doc]
print("Original:", sentence)
print("Lemmatized (spaCy):", " ".join(lemmatized_words))
print()

```

✓ Download and installation successful

You can now load the package via `spacy.load('en_core_web_sm')`

△ Restart to reload dependencies

If you are in a Jupyter or Colab notebook, you may need to restart Python in

order to load all the package's dependencies. You can do this by selecting the

'Restart kernel' or 'Restart runtime' option.

Original: The leaves on the tree have fallen.

Lemmatized (spaCy): the leave on the tree have fall .

Original: She is running towards the finish line.

Lemmatized (spaCy): she be run towards the finish line .

Original: The cars are parked in the parking lot.

Lemmatized (spaCy): the car be park in the parking lot .

Original: A quick brown fox jumps over the lazy dog.

Lemmatized (spaCy): a quick brown fox jump over the lazy dog .

Original: He was walking in the park and saw a beautiful bird.

Lemmatized (spaCy): he be walk in the park and see a beautiful bird .

`pip install spacy`

Collecting spacy

Downloading spacy-3.7.6-cp311-cp311-win\_amd64.whl.metadata (27 kB)

Collecting spacy-legacy<3.1.0,>=3.0.11 (from spacy)

Using cached spacy\_legacy-3.0.12-py2.py3-none-any.whl.metadata (2.8 kB)

Collecting spacy-loggers<2.0.0,>=1.0.0 (from spacy)

Using cached spacy\_loggers-1.0.5-py3-none-any.whl.metadata (23 kB)

Collecting murmurhash<1.1.0,>=0.28.0 (from spacy)

Using cached murmurhash-1.0.10-cp311-cp311-win\_amd64.whl.metadata (2.0 kB)

Collecting cymem<2.1.0,>=2.0.2 (from spacy)

Using cached cymem-2.0.8-cp311-cp311-win\_amd64.whl.metadata (8.6 kB)

Collecting preshed<3.1.0,>=3.0.2 (from spacy)

Using cached preshed-3.0.9-cp311-cp311-win\_amd64.whl.metadata (2.2 kB)

Collecting thinc<8.3.0,>=8.2.2 (from spacy)

Downloading thinc-8.2.5-cp311-cp311-win\_amd64.whl.metadata (15 kB)

Collecting wasabi<1.2.0,>=0.9.1 (from spacy)

Downloading wasabi-1.1.3-py3-none-any.whl.metadata (28 kB)  
Collecting srsly<3.0.0,>=2.4.3 (from spacy)  
Downloading srsly-2.4.8-cp311-cp311-win\_amd64.whl.metadata (20 kB)  
Collecting catalogue<2.1.0,>=2.0.6 (from spacy)  
Using cached catalogue-2.0.10-py3-none-any.whl.metadata (14 kB)  
Collecting weasel<0.5.0,>=0.1.0 (from spacy)  
Downloading weasel-0.4.1-py3-none-any.whl.metadata (4.6 kB)  
Collecting typer<1.0.0,>=0.3.0 (from spacy)  
Downloading typer-0.12.5-py3-none-any.whl.metadata (15 kB)  
Requirement already satisfied: tqdm<5.0.0,>=4.38.0 in d:\anaconda\lib\site-packages (from spacy) (4.65.0)  
Requirement already satisfied: requests<3.0.0,>=2.13.0 in d:\anaconda\lib\site-packages (from spacy) (2.31.0)  
Requirement already satisfied: pydantic!=1.8,!1.8.1,<3.0.0,>=1.7.4 in d:\anaconda\lib\site-packages (from spacy) (1.10.12)  
Requirement already satisfied: jinja2 in d:\anaconda\lib\site-packages (from spacy) (3.1.3)  
Requirement already satisfied: setuptools in d:\anaconda\lib\site-packages (from spacy) (68.2.2)  
Requirement already satisfied: packaging>=20.0 in d:\anaconda\lib\site-packages (from spacy) (23.1)  
Collecting langcodes<4.0.0,>=3.2.0 (from spacy)  
Downloading langcodes-3.4.0-py3-none-any.whl.metadata (29 kB)  
Requirement already satisfied: numpy>=1.19.0 in d:\anaconda\lib\site-packages (from spacy) (1.26.4)  
Collecting language-data>=1.2 (from langcodes<4.0.0,>=3.2.0->spacy)  
Downloading language\_data-1.2.0-py3-none-any.whl.metadata (4.3 kB)  
Requirement already satisfied: typing-extensions>=4.2.0 in d:\anaconda\lib\site-packages (from pydantic!=1.8,!1.8.1,<3.0.0,>=1.7.4->spacy) (4.9.0)  
Requirement already satisfied: charset-normalizer<4,>=2 in d:\anaconda\lib\site-packages (from requests<3.0.0,>=2.13.0->spacy) (2.0.4)  
Requirement already satisfied: idna<4,>=2.5 in d:\anaconda\lib\site-packages (from requests<3.0.0,>=2.13.0->spacy) (3.4)  
Requirement already satisfied: urllib3<3,>=1.21.1 in d:\anaconda\lib\site-packages (from requests<3.0.0,>=2.13.0->spacy) (2.0.7)  
Requirement already satisfied: certifi>=2017.4.17 in d:\anaconda\lib\site-packages (from requests<3.0.0,>=2.13.0->spacy) (2024.2.2)  
Collecting blis<0.8.0,>=0.7.8 (from thinc<8.3.0,>=8.2.2->spacy)  
Using cached blis-0.7.11-cp311-cp311-win\_amd64.whl.metadata (7.6 kB)  
Collecting confection<1.0.0,>=0.0.1 (from thinc<8.3.0,>=8.2.2->spacy)  
Downloading confection-0.1.5-py3-none-any.whl.metadata (19 kB)  
Requirement already satisfied: colorama in d:\anaconda\lib\site-packages (from tqdm<5.0.0,>=4.38.0->spacy) (0.4.6)  
Requirement already satisfied: click>=8.0.0 in d:\anaconda\lib\site-packages (from typer<1.0.0,>=0.3.0->spacy) (8.1.7)  
Collecting shellingham>=1.3.0 (from typer<1.0.0,>=0.3.0->spacy)  
Downloading shellingham-1.5.4-py2.py3-none-any.whl.metadata (3.5 kB)

```

Requirement already satisfied: rich>=10.11.0 in d:\anaconda\lib\site-
packages (from typer<1.0.0,>=0.3.0->spacy) (13.3.5)
Collecting cloudpathlib<1.0.0,>=0.7.0 (from weasel<0.5.0,>=0.1.0-
>spacy)
  Downloading cloudpathlib-0.19.0-py3-none-any.whl.metadata (14 kB)
Requirement already satisfied: smart-open<8.0.0,>=5.2.1 in d:\
anaconda\lib\site-packages (from weasel<0.5.0,>=0.1.0->spacy) (5.2.1)
Requirement already satisfied: MarkupSafe>=2.0 in d:\anaconda\lib\
site-packages (from jinja2->spacy) (2.1.3)
Collecting marisa-trie>=0.7.7 (from language-data>=1.2-
>langcodes<4.0.0,>=3.2.0->spacy)
  Downloading marisa_trie-1.2.0-cp311-cp311-win_amd64.whl.metadata
(9.0 kB)
Requirement already satisfied: markdown-it-py<3.0.0,>=2.2.0 in d:\
anaconda\lib\site-packages (from rich>=10.11.0->typer<1.0.0,>=0.3.0-
>spacy) (2.2.0)
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in d:\anaconda\
lib\site-packages (from rich>=10.11.0->typer<1.0.0,>=0.3.0->spacy)
(2.15.1)
Requirement already satisfied: mdurl~=0.1 in d:\anaconda\lib\site-
packages (from markdown-it-py<3.0.0,>=2.2.0->rich>=10.11.0-
>typer<1.0.0,>=0.3.0->spacy) (0.1.0)
Downloading spacy-3.7.6-cp311-cp311-win_amd64.whl (12.1 MB)
----- 0.0/12.1 MB ? eta -:--:--
- 0.6/12.1 MB 17.8 MB/s eta
0:00:01
-- 0.7/12.1 MB 9.5 MB/s eta
0:00:02
-- 0.9/12.1 MB 7.0 MB/s eta
0:00:02
--- 0.9/12.1 MB 7.5 MB/s eta
0:00:02
--- 0.9/12.1 MB 7.5 MB/s eta
0:00:02
--- 1.0/12.1 MB 3.9 MB/s eta
0:00:03
----- 1.6/12.1 MB 5.3 MB/s eta
0:00:02
----- 1.7/12.1 MB 5.0 MB/s eta
0:00:03
----- 1.9/12.1 MB 4.9 MB/s eta
0:00:03
----- 2.1/12.1 MB 4.9 MB/s eta
0:00:03
----- 2.1/12.1 MB 4.9 MB/s eta
0:00:03
----- 2.1/12.1 MB 4.9 MB/s eta
0:00:03
----- 2.7/12.1 MB 4.6 MB/s eta

```

0:00:03	-----	2.8/12.1 MB 4.5 MB/s eta
0:00:03	-----	3.0/12.1 MB 4.5 MB/s eta
0:00:03	-----	3.2/12.1 MB 4.5 MB/s eta
0:00:03	-----	3.3/12.1 MB 4.4 MB/s eta
0:00:02	-----	3.5/12.1 MB 4.4 MB/s eta
0:00:02	-----	3.7/12.1 MB 4.3 MB/s eta
0:00:02	-----	3.8/12.1 MB 4.4 MB/s eta
0:00:02	-----	4.0/12.1 MB 4.4 MB/s eta
0:00:02	-----	4.0/12.1 MB 4.2 MB/s eta
0:00:02	-----	4.4/12.1 MB 4.3 MB/s eta
0:00:02	-----	4.5/12.1 MB 4.3 MB/s eta
0:00:02	-----	4.7/12.1 MB 4.3 MB/s eta
0:00:02	-----	4.8/12.1 MB 4.3 MB/s eta
0:00:02	-----	5.0/12.1 MB 4.2 MB/s eta
0:00:02	-----	5.2/12.1 MB 4.2 MB/s eta
0:00:02	-----	5.3/12.1 MB 4.2 MB/s eta
0:00:02	-----	5.5/12.1 MB 4.2 MB/s eta
0:00:02	-----	5.6/12.1 MB 4.2 MB/s eta
0:00:02	-----	5.8/12.1 MB 4.2 MB/s eta
0:00:02	-----	6.0/12.1 MB 4.2 MB/s eta
0:00:02	-----	6.2/12.1 MB 4.2 MB/s eta
0:00:02	-----	6.4/12.1 MB 4.1 MB/s eta
0:00:02	-----	6.5/12.1 MB 4.2 MB/s eta
0:00:02	-----	6.7/12.1 MB 4.1 MB/s eta
0:00:02		

```
----- 6.9/12.1 MB 4.1 MB/s eta
0:00:02
----- 7.0/12.1 MB 4.1 MB/s eta
0:00:02
----- 7.2/12.1 MB 4.1 MB/s eta
0:00:02
----- 7.4/12.1 MB 4.1 MB/s eta
0:00:02
----- 7.6/12.1 MB 4.1 MB/s eta
0:00:02
----- 7.8/12.1 MB 4.1 MB/s eta
0:00:02
----- 8.0/12.1 MB 4.1 MB/s eta
0:00:02
----- 8.2/12.1 MB 4.1 MB/s eta
0:00:01
----- 8.3/12.1 MB 4.1 MB/s eta
0:00:01
----- 8.5/12.1 MB 4.1 MB/s eta
0:00:01
----- 8.7/12.1 MB 4.1 MB/s eta
0:00:01
----- 8.9/12.1 MB 4.1 MB/s eta
0:00:01
----- 9.0/12.1 MB 4.1 MB/s eta
0:00:01
----- 9.2/12.1 MB 4.1 MB/s eta
0:00:01
----- 9.3/12.1 MB 4.1 MB/s eta
0:00:01
----- 9.6/12.1 MB 4.0 MB/s eta
0:00:01
----- 9.7/12.1 MB 4.1 MB/s eta
0:00:01
----- 10.0/12.1 MB 4.1 MB/s eta
0:00:01
----- 10.1/12.1 MB 4.1 MB/s eta
0:00:01
----- 10.3/12.1 MB 4.0 MB/s eta
0:00:01
----- 10.5/12.1 MB 4.0 MB/s eta
0:00:01
----- 10.6/12.1 MB 3.9 MB/s eta
0:00:01
----- 10.8/12.1 MB 3.9 MB/s eta
0:00:01
----- 11.0/12.1 MB 3.9 MB/s eta
0:00:01
----- 11.2/12.1 MB 3.9 MB/s eta
```

```
0:00:01
----- -- 11.4/12.1 MB 4.0 MB/s eta
0:00:01
----- - 11.6/12.1 MB 4.0 MB/s eta
0:00:01
----- - 11.7/12.1 MB 3.9 MB/s eta
0:00:01
----- 11.9/12.1 MB 3.9 MB/s eta
0:00:01
----- 12.1/12.1 MB 3.9 MB/s eta
0:00:01
----- 12.1/12.1 MB 3.9 MB/s eta
0:00:01
----- 12.1/12.1 MB 3.8 MB/s eta
0:00:00
Using cached catalogue-2.0.10-py3-none-any.whl (17 kB)
Using cached cymem-2.0.8-cp311-cp311-win_amd64.whl (39 kB)
Downloading langcodes-3.4.0-py3-none-any.whl (182 kB)
----- 0.0/182.0 kB ? eta -:--:--
----- 182.0/182.0 kB 5.5 MB/s eta
0:00:00
Using cached murmurhash-1.0.10-cp311-cp311-win_amd64.whl (25 kB)
Using cached preshed-3.0.9-cp311-cp311-win_amd64.whl (122 kB)
Using cached spacy_legacy-3.0.12-py2.py3-none-any.whl (29 kB)
Using cached spacy_loggers-1.0.5-py3-none-any.whl (22 kB)
Downloading srsly-2.4.8-cp311-cp311-win_amd64.whl (479 kB)
----- 0.0/479.7 kB ? eta -:--:--
----- 479.7/479.7 kB 10.0 MB/s eta
0:00:00
Downloading thinc-8.2.5-cp311-cp311-win_amd64.whl (1.5 MB)
----- 0.0/1.5 MB ? eta -:--:--
----- 0.4/1.5 MB 11.8 MB/s eta
0:00:01
----- 0.6/1.5 MB 5.8 MB/s eta
0:00:01
----- 0.7/1.5 MB 5.7 MB/s eta
0:00:01
----- 0.9/1.5 MB 5.1 MB/s eta
0:00:01
----- 1.0/1.5 MB 5.0 MB/s eta
0:00:01
----- 1.2/1.5 MB 4.8 MB/s eta
0:00:01
----- 1.4/1.5 MB 4.7 MB/s eta
0:00:01
----- 1.5/1.5 MB 4.7 MB/s eta
0:00:01
----- 1.5/1.5 MB 4.3 MB/s eta
0:00:00
```



```
Downloading typer-0.12.5-py3-none-any.whl (47 kB)
----- 0.0/47.3 kB ? eta -:--:--
----- 47.3/47.3 kB 1.2 MB/s eta
0:00:00
Downloading wasabi-1.1.3-py3-none-any.whl (27 kB)
Downloading weasel-0.4.1-py3-none-any.whl (50 kB)
----- 0.0/50.3 kB ? eta -:--:--
----- 50.3/50.3 kB 2.5 MB/s eta
0:00:00
Using cached blis-0.7.11-cp311-cp311-win_amd64.whl (6.6 MB)
Downloading cloudpathlib-0.19.0-py3-none-any.whl (49 kB)
----- 0.0/49.4 kB ? eta -:--:--
----- 49.4/49.4 kB 1.2 MB/s eta
0:00:00
Downloading confection-0.1.5-py3-none-any.whl (35 kB)
Downloading language_data-1.2.0-py3-none-any.whl (5.4 MB)
----- 0.0/5.4 MB ? eta -:--:--
----- 0.6/5.4 MB 17.5 MB/s eta
0:00:01
----- 0.7/5.4 MB 7.8 MB/s eta
0:00:01
----- 0.9/5.4 MB 7.2 MB/s eta
0:00:01
----- 1.1/5.4 MB 6.3 MB/s eta
0:00:01
----- 1.3/5.4 MB 5.8 MB/s eta
0:00:01
----- 1.5/5.4 MB 5.4 MB/s eta
0:00:01
----- 1.6/5.4 MB 5.2 MB/s eta
0:00:01
----- 1.8/5.4 MB 4.9 MB/s eta
0:00:01
----- 1.9/5.4 MB 4.9 MB/s eta
0:00:01
----- 2.2/5.4 MB 4.7 MB/s eta
0:00:01
----- 2.3/5.4 MB 4.8 MB/s eta
0:00:01
----- 2.5/5.4 MB 4.6 MB/s eta
0:00:01
----- 2.6/5.4 MB 4.6 MB/s eta
0:00:01
----- 2.8/5.4 MB 4.6 MB/s eta
0:00:01
----- 3.0/5.4 MB 4.5 MB/s eta
0:00:01
----- 3.1/5.4 MB 4.5 MB/s eta
0:00:01
```

```

----- 3.3/5.4 MB 4.5 MB/s eta
0:00:01
----- 3.5/5.4 MB 4.4 MB/s eta
0:00:01
----- 3.7/5.4 MB 4.4 MB/s eta
0:00:01
----- 3.8/5.4 MB 4.3 MB/s eta
0:00:01
----- 4.0/5.4 MB 4.3 MB/s eta
0:00:01
----- 4.2/5.4 MB 4.4 MB/s eta
0:00:01
----- 4.4/5.4 MB 4.3 MB/s eta
0:00:01
----- 4.5/5.4 MB 4.3 MB/s eta
0:00:01
----- 4.7/5.4 MB 4.3 MB/s eta
0:00:01
----- 4.9/5.4 MB 4.2 MB/s eta
0:00:01
----- 5.1/5.4 MB 4.2 MB/s eta
0:00:01
----- 5.2/5.4 MB 4.2 MB/s eta
0:00:01
----- 5.4/5.4 MB 4.2 MB/s eta
0:00:01
----- 5.4/5.4 MB 4.2 MB/s eta
0:00:01
----- 5.4/5.4 MB 4.0 MB/s eta
0:00:00
Downloading shellingham-1.5.4-py2.py3-none-any.whl (9.8 kB)
Downloading marisa_trie-1.2.0-cp311-cp311-win_amd64.whl (152 kB)
----- 0.0/152.6 kB ? eta -:--:--
----- 152.6/152.6 kB 4.6 MB/s eta
0:00:00
Installing collected packages: cymem, wasabi, spacy-loggers, spacy-
legacy, shellingham, murmurhash, marisa-trie, cloudpathlib, catalogue,
blis, srsly, preshed, language-data, typer, langcodes, confection,
weasel, thinc, spacy
Successfully installed blis-0.7.11 catalogue-2.0.10 cloudpathlib-
0.19.0 confection-0.1.5 cymem-2.0.8 langcodes-3.4.0 language-data-
1.2.0 marisa-trie-1.2.0 murmurhash-1.0.10 preshed-3.0.9 shellingham-
1.5.4 spacy-3.7.6 spacy-legacy-3.0.12 spacy-loggers-1.0.5 srsly-2.4.8
thinc-8.2.5 typer-0.12.5 wasabi-1.1.3 weasel-0.4.1
Note: you may need to restart the kernel to use updated packages.

```

spaCy provides lemmatization out of the box using its pre-trained models. It's more sophisticated than NLTK's WordNetLemmatizer, as it considers the context of the word within

the sentence. Example: The phrase was walking would correctly be lemmatized to be walk (where was becomes be and walking becomes walk).

#### *# Combining spaCy with NLTK for stemming*

```
for sentence in corpus:
    doc = nlp(sentence)
    stemmed_words = [ps.stem(token.text) for token in doc]
    print("Original:", sentence)
    print("Stemmed (spaCy+Porter):", " ".join(stemmed_words))
    print()
```

Original: The leaves on the tree have fallen.

Stemmed (spaCy+Porter): the leav on the tree have fallen .

Original: She is running towards the finish line.

Stemmed (spaCy+Porter): she is run toward the finish line .

Original: The cars are parked in the parking lot.

Stemmed (spaCy+Porter): the car are park in the park lot .

Original: A quick brown fox jumps over the lazy dog.

Stemmed (spaCy+Porter): a quick brown fox jump over the lazi dog .

Original: He was walking in the park and saw a beautiful bird.

Stemmed (spaCy+Porter): he wa walk in the park and saw a beauti bird .

NLTK's PorterStemmer is a rule-based stemmer that sometimes produces results that are not real words (e.g., running -> run, better -> bett). It's faster but less accurate than lemmatization.

NLTK's WordNetLemmatizer uses WordNet's lexical database for more accurate lemmatization. However, it requires correct POS tagging to work well, which can be a limitation if you're processing large amounts of text without POS tagging.

spaCy's Lemmatizer is more advanced and context-aware, making it generally more accurate than NLTK's WordNetLemmatizer. It's better suited for tasks requiring higher accuracy in natural language processing.

spaCy + NLTK's PorterStemmer allows for stemming within the spaCy framework, though it's not as common as using lemmatization in spaCy.

For most modern NLP applications, lemmatization (especially using spaCy) is preferred over stemming due to its accuracy and context-awareness. However, stemming might still be useful in specific scenarios where speed is crucial and slight inaccuracies can be tolerated.