In [1]:
```
import random
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from tqdm import tqdm
import torch
import transformers

sns.set()
plt.rcParams['figure.dpi'] = 100
```

Collection of movie reviews

In [2]:
```
training_reviews_with_labels = [
    ("Watched the whole thing: surprisingly good!", 'positive'),
    ("Beautifully, thoughtfully made", 'positive'),
    ("A good end to a good season", 'positive'),
    ("Better than expected", 'positive'),
    ("Amazing achievement!!", 'positive'),
    ("It's fantastic", 'positive'),
    ("Fully prepared to hate this... completely surprised!", 'positive'),
    ("The right direction", 'positive'),
    ("A visual and storytelling masterpiece.", 'positive'),
    ("Definitely should see if you're a fan of the genre", 'positive'),
    ("Amazing cinematography, most of the storylines are good", 'positive'),
    ("A great series you can actually watch with your family.", 'positive'),
    ("Pretty good start", 'positive'),

    ("Beautiful to watch, a few interesting characters, storyline is good until it isn't",
    ("So much potential, but less realization", 'neutral'),
    ("Missed the mark", 'neutral'),
    ("Has potential, but also flaws", 'neutral'),
    ("Amazing looks but lacking a clear Jacksoneque vision", 'neutral'),
    ("Slow; Slow; Slow", 'neutral'),
    ("It's fine - as in OK - as in mediocre", 'neutral'),
    ("Beautiful but", 'neutral'),
    ("Great CGI but lacks an interesting plot", 'neutral'),

    ("More boring than logic homework", 'negative'),
    ("A major disappointment", 'negative'),
    ("Horrible writing, slow plot, and disrespectful", 'negative'),
    ("An ok fantasy story that has little to do with Tolkien", 'negative'),
    ("Boring, even for generic fantasy", 'negative'),
    ("So many problems", 'negative'),
    ("What is this, an Anti-FanFic or something? Please read the books!", 'negative'),
    ("Just bad all around", 'negative'),
    ("I mourn for what this could have been", 'negative'),
    ("Short version, skip it. You'll know you did right when you don't hear people talkinc
    ("Could have been so much more", 'negative'),
]
```

In [3]:
```
test_reviews_with_labels = [
    ("The best show I've seen so far this year!", 'positive'),
    ("Really enjoyed it", 'positive'),
    ("Amazing.", 'positive'),
    ("Why all the hate? I enjoyed it.", 'positive'),
    ("Beautiful visuals, entertaining, and I believe this show has a lot of potential!",
    ("A beautiful rendering of Middle Earth's history", 'positive'),
```

```
    ("So far, so good... and there's still hope", 'positive'),
    ("I'm a fan", 'positive'),
    ("It works for me", 'positive'),
    ("Not the best, but enjoyed every episode. Can't wait to see much much more.", 'positi
    ("Beautiful, flawed, and a wonderful Fall treat", 'positive'),

    ("Good show with too many subplots", 'neutral'),
    ("Starts badly, gets better", 'neutral'),
    ("Good and bad things", 'neutral'),
    ("Big and beautiful but can use a little help with its energy.", 'neutral'),
    ("Pretty but ultimately hollow and lacking in engagement", 'neutral'),
    ("Beautiful visuals and story overshadowed by unnecessary gore and violence", 'neutral

    ("Not what you're probably expecting", 'negative'),
    ("Poor writing; Uninteresting characters, nonsensical actions.", 'negative'),
    ("Budget was spent on snacks between shots", 'negative'),
    ("If you ignore the source material, it's still boring and weird", 'negative'),
    ("Just a bad show", 'negative'),
    ("It's awful", 'negative'),
    ("I was hopeful...", 'negative'),
    ("Painfully mediocre with a few good spots", 'negative'),
    ("Beautiful to look at... but that’s about it.", 'negative'),
    ("Underwhelming and disappointing", 'negative'),
    ("Tolkien is rolling in his grave. No mystery. No inspiration. Wardrobe & acting is pr
]
```

In [4]:

```
labels = ['positive', 'neutral', 'negative']

training_reviews = [review for review, label in training_reviews_with_labels]
training_labels = np.array([label for review, label in training_reviews_with_labels])

test_reviews = [review for review, label in test_reviews_with_labels]
test_labels = np.array([label for review, label in test_reviews_with_labels])

print('Labels:')
print(labels)
print()

print(f'There are {len(training_reviews)} training reviews')
print('Training labels:')
print(training_labels)
print()

print(f'There are {len(test_reviews)} test reviews')
print('Test labels:')
print(test_labels)
```

```
Labels:
['positive', 'neutral', 'negative']

There are 33 training reviews
Training labels:
['positive' 'positive' 'positive' 'positive' 'positive' 'positive'
 'positive' 'positive' 'positive' 'positive' 'positive' 'positive'
 'positive' 'neutral' 'neutral' 'neutral' 'neutral' 'neutral' 'neutral'
 'neutral' 'neutral' 'neutral' 'negative' 'negative' 'negative' 'negative'
 'negative' 'negative' 'negative' 'negative' 'negative' 'negative'
 'negative']

There are 28 test reviews
Test labels:
['positive' 'positive' 'positive' 'positive' 'positive' 'positive'
 'positive' 'positive' 'positive' 'positive' 'positive' 'neutral'
 'neutral' 'neutral' 'neutral' 'neutral' 'neutral' 'negative' 'negative'
```

```
'negative' 'negative' 'negative' 'negative' 'negative' 'negative'
'negative' 'negative' 'negative']
```

Zero-shot classification with T0

In [5]:
```
from transformers import AutoModelForSeq2SeqLM, AutoTokenizer

transformers.utils.logging.set_verbosity_error()
transformers.set_seed(42)
```

In [6]:
```
tokenizer = AutoTokenizer.from_pretrained("bigscience/T0_3B")
model = AutoModelForSeq2SeqLM.from_pretrained("bigscience/T0_3B").cuda()
```

In [7]:
```
def generate(prompt, **kwargs):
    x = tokenizer(prompt, return_tensors='pt').input_ids.to(model.device)
    y = model.generate(x, **kwargs).cpu()
    return tokenizer.batch_decode(y, skip_special_tokens=True)[0]
```

In [8]:
```
def construct_prompt(review):
    return f'Review: {review}\nIs this review positive, negative, or neutral?'
```

In [9]:
```
prompt = construct_prompt('Watched the whole thing: surprisingly good!')
print(prompt)

output = generate(prompt, max_new_tokens=3)
print(output)
```

```
Review: Watched the whole thing: surprisingly good!
Is this review positive, negative, or neutral?
positive
```

In [10]:
```
from scipy.special import softmax
from sklearn.metrics import confusion_matrix, accuracy_score, f1_score
from IPython.display import display

predictions = []
table = []

for review, label in zip(test_reviews, test_labels):
    prompt = construct_prompt(review)
    prediction = generate(prompt, max_new_tokens=3).lower()

    row = {
        'Review': review,
        'Label': label,
        'Prediction': prediction,
    }

    predictions.append(prediction)
    table.append(row)

print(f'Accuracy: {accuracy_score(test_labels, predictions) * 100:.1f}%')
print(f'F1 score: {f1_score(test_labels, predictions, labels=labels, average=None)}')
```

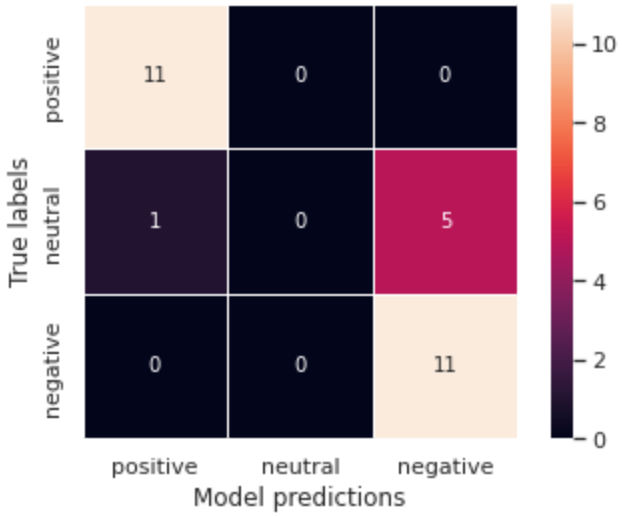
```
Accuracy: 78.6%
F1 score: [0.95652174 0.         0.81481481]
```

In [11]:
```
pd.set_option('display.max_rows', None)
pd.set_option('display.max_colwidth', None)
display(pd.DataFrame(table))
```

	Review	Label	Prediction
0	The best show I've seen so far this year!	positive	positive
1	Really enjoyed it	positive	positive
2	Amazing.	positive	positive
3	Why all the hate? I enjoyed it.	positive	positive
4	Beautiful visuals, entertaining, and I believe this show has a lot of potential!	positive	positive
5	A beautiful rendering of Middle Earth's history	positive	positive
6	So far, so good... and there's still hope	positive	positive
7	I'm a fan	positive	positive
8	It works for me	positive	positive
9	Not the best, but enjoyed every episode. Can't wait to see much much more.	positive	positive
10	Beautiful, flawed, and a wonderful Fall treat	positive	positive
11	Good show with too many subplots	neutral	negative
12	Starts badly, gets better	neutral	negative
13	Good and bad things	neutral	positive
14	Big and beautiful but can use a little help with its energy.	neutral	negative
15	Pretty but ultimately hollow and lacking in engagement	neutral	negative
16	Beautiful visuals and story overshadowed by unnecessary gore and violence	neutral	negative
17	Not what you're probably expecting	negative	negative
18	Poor writing; Uninteresting characters, nonsensical actions.	negative	negative
19	Budget was spent on snacks between shots	negative	negative
20	If you ignore the source material, it's still boring and weird	negative	negative
21	Just a bad show	negative	negative
22	It's awful	negative	negative
23	I was hopeful...	negative	negative
24	Painfully mediocre with a few good spots	negative	negative
25	Beautiful to look at... but that’s about it.	negative	negative
26	Underwhelming and disappointing	negative	negative
27	Tolkien is rolling in his grave. No mystery. No inspiration. Wardrobe & acting is pretty bland.	negative	negative

In [12]:
```
mat = confusion_matrix(test_labels, predictions, labels=labels)

sns.heatmap(mat, square=True, annot=True, xticklabels=labels, yticklabels=labels, linewidt
plt.xlabel("Model predictions")
plt.ylabel("True labels")
plt.show()
```



Zero-shot classification with OpenAI models

In [13]:
```
import openai

with open('OPENAI_API_KEY', 'r') as f:
    openai.api_key = f.readline().strip()
```

In [14]:
```
from sklearn.metrics import confusion_matrix, accuracy_score, f1_score
from IPython.display import display

predictions = []
table = []

for review, label in zip(test_reviews, test_labels):
    prompt = construct_prompt(review)

    response = openai.Completion.create(
        model="text-davinci-003",
        prompt=prompt,
        temperature=0,
        max_tokens=5,
        top_p=1,
        frequency_penalty=0,
        presence_penalty=0
    )

    prediction = response['choices'][0]['text'].strip().lower()

    row = {
        'Review': review,
        'Label': label,
        'Prediction': prediction,
    }

    predictions.append(prediction)
    table.append(row)

print(f'Accuracy: {accuracy_score(test_labels, predictions) * 100:.1f}%')
print(f'F1 score: {f1_score(test_labels, predictions, labels=labels, average=None)}')
```

```
Accuracy: 82.1%
F1 score: [1.         0.61538462 0.76190476]
```

In [15]:

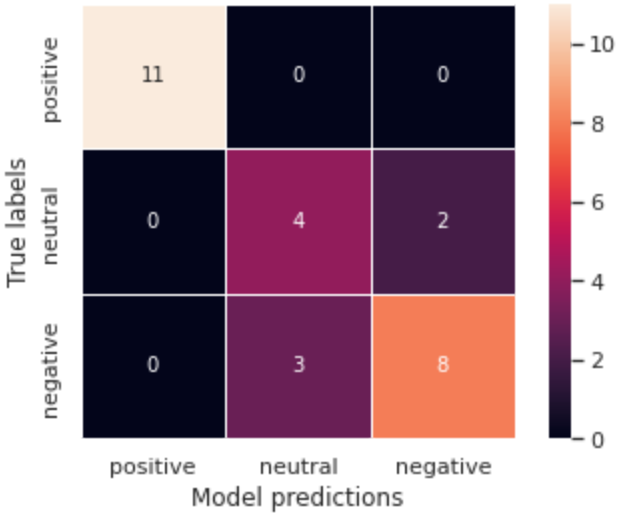
```
pd.set_option('display.max_rows', None)
pd.set_option('display.max_colwidth', None)
display(pd.DataFrame(table))
```

	Review	Label	Prediction
0	The best show I've seen so far this year!	positive	positive
1	Really enjoyed it	positive	positive
2	Amazing.	positive	positive
3	Why all the hate? I enjoyed it.	positive	positive
4	Beautiful visuals, entertaining, and I believe this show has a lot of potential!	positive	positive
5	A beautiful rendering of Middle Earth's history	positive	positive
6	So far, so good... and there's still hope	positive	positive
7	I'm a fan	positive	positive
8	It works for me	positive	positive
9	Not the best, but enjoyed every episode. Can't wait to see much much more.	positive	positive
10	Beautiful, flawed, and a wonderful Fall treat	positive	positive
11	Good show with too many subplots	neutral	neutral
12	Starts badly, gets better	neutral	neutral
13	Good and bad things	neutral	neutral
14	Big and beautiful but can use a little help with its energy.	neutral	neutral
15	Pretty but ultimately hollow and lacking in engagement	neutral	negative
16	Beautiful visuals and story overshadowed by unnecessary gore and violence	neutral	negative
17	Not what you're probably expecting	negative	neutral
18	Poor writing; Uninteresting characters, nonsensical actions.	negative	negative
19	Budget was spent on snacks between shots	negative	neutral
20	If you ignore the source material, it's still boring and weird	negative	negative
21	Just a bad show	negative	negative
22	It's awful	negative	negative
23	I was hopeful...	negative	neutral
24	Painfully mediocre with a few good spots	negative	negative
25	Beautiful to look at... but that’s about it.	negative	negative
26	Underwhelming and disappointing	negative	negative
27	Tolkien is rolling in his grave. No mystery. No inspiration. Wardrobe & acting is pretty bland.	negative	negative

In [16]:

```
mat = confusion_matrix(test_labels, predictions, labels=labels)

sns.heatmap(mat, square=True, annot=True, xticklabels=labels, yticklabels=labels, linewidt
plt.xlabel("Model predictions")
plt.ylabel("True labels")
plt.show()
```


positive
neutral
negative
True labels
11
0
0
0
4
2
0
3
8
positive
neutral
negative
Model predictions
10
8
6
4
2
0