

Имеется датасет **ner\_datasetreference.csv**. В нем предложения уже разбиты на слова:

	Sentence #	Word	POS	Tag
0	Sentence: 1	Thousands	NNS	O
1	NaN	of	IN	O
2	NaN	demonstrators	NNS	O
3	NaN	have	VBP	O
4	NaN	marched	VBN	O
5	NaN	through	IN	O
6	NaN	London	NNP	B-geo
7	NaN	to	TO	O
8	NaN	protest	VB	O
9	NaN	the	DT	O
10	NaN	war	NN	O
11	NaN	in	IN	O
12	NaN	Iraq	NNP	B-geo
13	NaN	and	CC	O
14	NaN	demand	VB	O
15	NaN	the	DT	O
16	NaN	withdrawal	NN	O
17	NaN	of	IN	O
18	NaN	British	JJ	B-gpe
19	NaN	troops	NNS	O

**Sentence #** - индикатор, который означает, что началось следующее предложение.

**Word** - колонка со словами предложения.

**POS** - колонка с лексемами, игнорируйте эту колонку для данного задания (необязательно, можете использовать если знаете как).

**Tag** - колонка с тэгами слов.

**Задание:** решить задачу **NER**. Сущности, которые нас интересуют находятся в колонке **Tag**. Сущности с тэгами «**O**» нас не интересуют, нас интересуют все остальные сущности. Решите задачу **NER** с использованием **keras** (tensorflow.keras).

**Важно:** решите задачу построив **LSTM** модель.

Как метрику для оценки тренировки нейросети используйте **accuracy**.

**Особенно важно:** при тренировке нейросети на самых первых эпохах вы получите высокую ассигасу. Это связано с тем, что тэгов «**O**» большинство (а они нам не интересны), т.е. если вы с вероятностью 95% (например) будете рандомно назначать словам тэг «**O**», то будете с высокой вероятностью

угадывать тэги «**O**» и ваша **accuracy** сразу получится достаточно высокой. Поэтому, исключите тэги «**O**» из подсчета accuracy при тренировке нейросети.