Multiclass Text Classification with

Feed-forward Neural Networks and Word Embeddings

First, we will do some initialization.

In [1]:

```
import random
import torch
import numpy as np
import pandas as pd
from tqdm.notebook import tqdm

# enable tqdm in pandas
tqdm.pandas()

# set to True to use the gpu (if there is one available)
use_gpu = True

# select device
device = torch.device('cuda' if use_gpu and torch.cuda.is_available() else 'cpu
print(f'device: {device.type}')

# random seed
seed = 1234

# set random seed
if seed is not None:
    print(f'random seed: {seed}')
    random.seed(seed)
    np.random.seed(seed)
    torch.manual_seed(seed)
```

```
device: cpu
random seed: 1234
```

We will be using the AG's News Topic Classification Dataset. It is stored in two CSV files: `train.csv` and `test.csv`, as well as a `classes.txt` that stores the labels of the classes to predict.

First, we will load the training dataset using pandas and take a quick look at how the data.

La razon del porque seleccionamos 70% fue debido a que nos ayuda a prevenir problemas debido a los recursos limitados

In [2]:

```
#Obtenemos la informacion de dataset de train, para poder obtener las clases, a
#un 70% de los datos son usados de entrenamiento
train_df = pd.read_csv('/kaggle/input/agnews-pytorch-simple-embed-classif-90/A
train_df.columns = ['class index', 'title', 'description']
train_df = train_df.sample(frac=0.7,random_state=42)
train_df
```

Out[2]:

	class index	title	description
71787	3	BBC set for major shake-up, claims newspaper	London - The British Broadcasting Corporation,...
67218	3	Marsh averts cash crunch	Embattled insurance broker #39;s banks agree t...
54066	2	Jeter, Yankees Look to Take Control (AP)	AP - Derek Jeter turned a season that started ...
7168	4	Flying the Sun to Safety	When the Genesis capsule comes back to Earth w...
29618	3	Stocks Seen Flat as Nortel and Oil Weigh	NEW YORK (Reuters) - U.S. stocks were set to ...
...	...	...	...
53857	1	FDA Accused of Silencing Vioxx Warnings	WASHINGTON - The Food and Drug Administration ...
111476	2	Buckeyes won #39;t play in NCAA or NIT tourneys	COLUMBUS, Ohio Ohio State has sanctioned its m...
6343	3	Rate hikes by Fed work in two ways	If you #39;ve noticed that the price of everyt...
20736	4	NASA Administrator Offers Support for Kennedy ...	The following is a statement from NASA Adminis...
34378	2	Twins make it 3 straight	The Minnesota Twins clinched on a bus in 1991....

84000 rows × 3 columns

The dataset consists of 120,000 examples, each consisting of a class index, a title, and a description. The class labels are distributed in a separated file. We will add the labels to the dataset so that we can interpret the data more easily. Note that the label indexes are one-based, so we need to subtract one to retrieve them from the list.

Se renombran sus columnars para una interpretacion mas facil

In [3]:

```
#Obtiene los titulos de las classes, los cuales se encuentran en el documento
labels = open('/kaggle/input/namesste/classes.txt').read().splitlines()
classes = train_df['class index'].map(lambda i: labels[i-1])
train_df.insert(1, 'class', classes)
train_df
```

Out[3]:

	class index	class	title	description
71787	3	Business	BBC set for major shake-up, claims newspaper	London - The British Broadcasting Corporation,...
67218	3	Business	Marsh averts cash crunch	Embattled insurance broker #39;s banks agree t...
54066	2	Sports	Jeter, Yankees Look to Take Control (AP)	AP - Derek Jeter turned a season that started ...
7168	4	Sci/Tech	Flying the Sun to Safety	When the Genesis capsule comes back to Earth w...
29618	3	Business	Stocks Seen Flat as Nortel and Oil Weigh	NEW YORK (Reuters) - U.S. stocks were set to ...
...	...	...	...	...
53857	1	World	FDA Accused of Silencing Vioxx Warnings	WASHINGTON - The Food and Drug Administration ...
111476	2	Sports	Buckeyes won #39;t play in NCAA or NIT tourneys	COLUMBUS, Ohio Ohio State has sanctioned its m...
6343	3	Business	Rate hikes by Fed work in two ways	If you #39;ve noticed that the price of everyt...
20736	4	Sci/Tech	NASA Administrator Offers Support for Kennedy ...	The following is a statement from NASA Adminis...
34378	2	Sports	Twins make it 3 straight	The Minnesota Twins clinched on a bus in 1991....

84000 rows × 4 columns

Let's inspect how balanced our examples are by using a bar plot.

Se grafican los datos para verificarlos

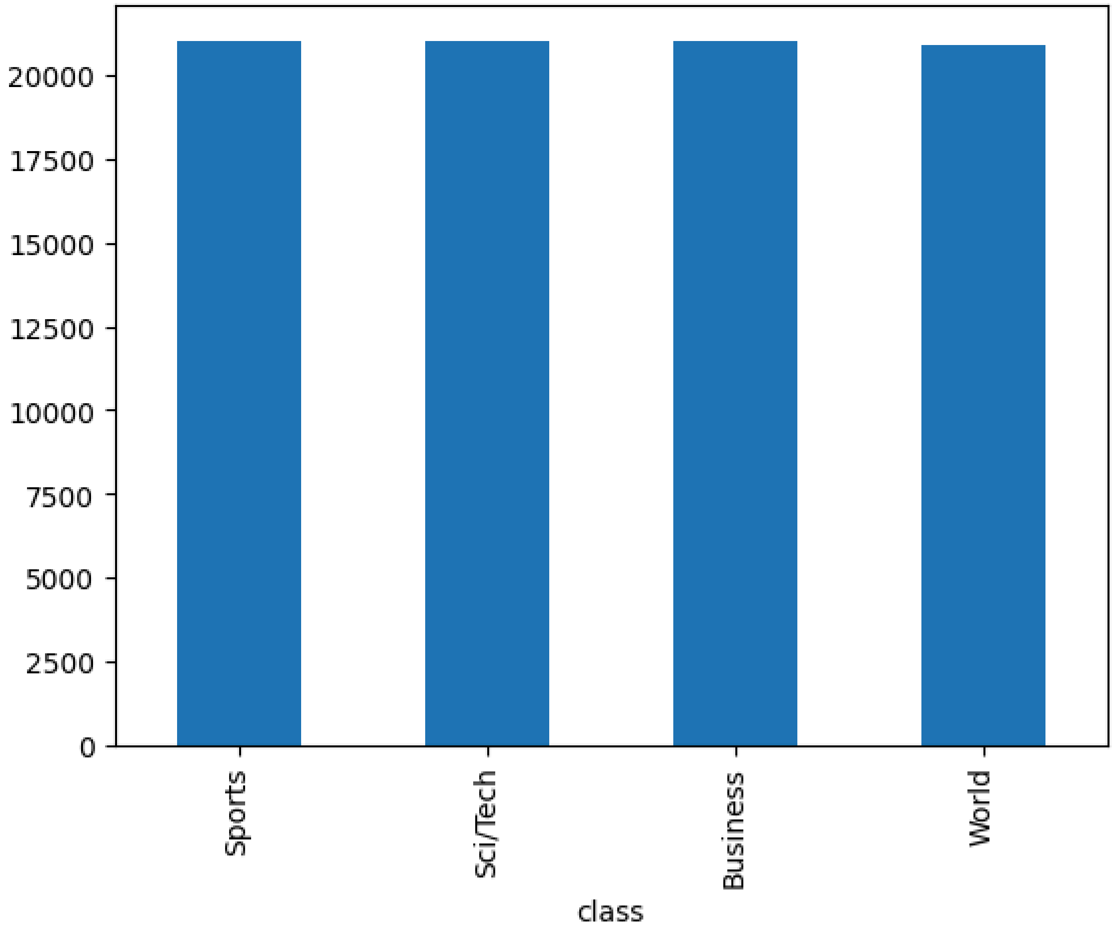
In [4]:

```
#Se grafican para poder observarlos
pd.value_counts(train_df['class']).plot.bar()
```

```
/tmp/ipykernel_30/2157117126.py:2: FutureWarning: pandas.value_counts is depre
cated and will be removed in a future version. Use pd.Series(obj).value_counts
() instead.
  pd.value_counts(train_df['class']).plot.bar()
```

Out[4]:

```
<Axes: xlabel='class'>
```



The classes are evenly distributed. That's great!

However, the text contains some spurious backslashes in some parts of the text. They are meant to represent newlines in the original text. An example can be seen below, between the words "dwindling" and "band".

In [5]:
```
#Nos ayuda a observar que se tienen \ en el texto, lo cual no es bueno
print(train_df.loc[0, 'description'])
```

```
Reuters - Short-sellers, Wall Street's dwindling\band of ultra-cynics, are see
ing green again.
```

We will replace the backslashes with spaces on the whole column using pandas replace method.

Se inicia el procesamiento de texto limpiando los diagonales del texto y convertir las palabras a minusculas

In [6]:
```
# Remplazan los \ por espacios

train_df['text'] = train_df['title'].str.lower() + " " + train_df['description
train_df['text'] = train_df['text'].str.replace('\\', ' ', regex=False)
train_df
```

Out[6]:

	class index	class	title	description	text
71787	3	Business	BBC set for major shake-up, claims newspaper	London - The British Broadcasting Corporation,...	bbc set for major shake-up, claims newspaper l...
67218	3	Business	Marsh averts cash crunch	Embattled insurance broker #39;s banks agree t...	marsh averts cash crunch embattled insurance b...
54066	2	Sports	Jeter, Yankees Look to Take Control (AP)	AP - Derek Jeter turned a season that started ...	jeter, yankees look to take control (ap) ap - ...
7168	4	Sci/Tech	Flying the Sun to Safety	When the Genesis capsule comes back to Earth w...	flying the sun to safety when the genesis caps...
29618	3	Business	Stocks Seen Flat as Nortel and Oil Weigh	NEW YORK (Reuters) - U.S. stocks were set to ...	stocks seen flat as nortel and oil weigh new ...
...	...	...	...	...	...
53857	1	World	FDA Accused of Silencing Vioxx Warnings	WASHINGTON - The Food and Drug Administration ...	fda accused of silencing vioxx warnings washin...
111476	2	Sports	Buckeyes won #39;t play in NCAA or NIT tourneys	COLUMBUS, Ohio Ohio State has sanctioned its m...	buckeyes won #39;t play in ncaa or nit tourney...
6343	3	Business	Rate hikes by Fed work in two ways	If you #39;ve noticed that the price of everyt...	rate hikes by fed work in two ways if you #39;...
20736	4	Sci/Tech	NASA Administrator Offers Support for Kennedy ...	The following is a statement from NASA Adminis...	nasa administrator offers support for kennedy ...
34378	2	Sports	Twins make it 3 straight	The Minnesota Twins clinched on a bus in 1991....	twins make it 3 straight the minnesota twins c...

84000 rows × 5 columns

Now we will proceed to tokenize the title and description columns using NLTK's word_tokenize(). We will add a new column to our dataframe with the list of tokens.

Se tokenizan las frases separando cada oracion para facilitar el proceso al modelo

In [7]:

```
#Tokeniza nuestras oraciones para su posterior analisis, creando una nueva colu

from nltk.tokenize import word_tokenize

train_df['tokens'] = train_df['text'].progress_map(word_tokenize)
train_df
```

```
  0%|          | 0/84000 [00:00<?, ?it/s]
```

Out[7]:

	class index	class	title	description	text	tokens
71787	3	Business	BBC set for major shake-up, claims newspaper	London - The British Broadcasting Corporation,...	bbc set for major shake-up, claims newspaper l...	[bbc, set, for, major, shake-up, ,, claims, ne...
67218	3	Business	Marsh averts cash crunch	Embattled insurance broker #39;s banks agree t...	marsh averts cash crunch embattled insurance b...	[marsh, averts, cash, crunch, embattled, insur...
54066	2	Sports	Jeter, Yankees Look to Take Control (AP)	AP - Derek Jeter turned a season that started ...	jeter, yankees look to take control (ap) ap - ...	[jeter, ,, yankees, look, to, take, control, (...
7168	4	Sci/Tech	Flying the Sun to Safety	When the Genesis capsule comes back to Earth w...	flying the sun to safety when the genesis caps...	[flying, the, sun, to, safety, when, the, gene...
29618	3	Business	Stocks Seen Flat as Nortel and Oil Weigh	NEW YORK (Reuters) - U.S. stocks were set to ...	stocks seen flat as nortel and oil weigh new ...	[stocks, seen, flat, as, nortel, and, oil, wei...
...	...	...	...	...	...	...
53857	1	World	FDA Accused of Silencing Vioxx Warnings	WASHINGTON - The Food and Drug Administration ...	fda accused of silencing vioxx warnings washin...	[fda, accused, of, silencing, vioxx, warnings,...
111476	2	Sports	Buckeyes won #39;t play in NCAA or NIT tourneys	COLUMBUS, Ohio Ohio State has sanctioned its m...	buckeyes won #39;t play in ncaa or nit tourney...	[buckeyes, won, #, 39, ;, t, play, in, ncaa, o...
6343	3	Business	Rate hikes by Fed work in two ways	If you #39;ve noticed that the price of everyt...	rate hikes by fed work in two ways if you #39;...	[rate, hikes, by, fed, work, in, two, ways, if...
20736	4	Sci/Tech	NASA Administrator Offers Support for Kennedy ...	The following is a statement from NASA Adminis...	nasa administrator offers support for kennedy ...	[nasa, administrator, offers, support, for, ke...
34378	2	Sports	Twins make it 3 straight	The Minnesota Twins clinched on a bus in 1991....	twins make it 3 straight the minnesota twins c...	[twins, make, it, 3, straight, the, minnesota,...

84000 rows × 6 columns

Now we will load the GloVe word embeddings.

contiene vectores de palabras preentrenados de GloVe con 300 dimensiones por palabras

In [8]:

```
#contiene vectores de palabras preentrenados de GloVe con 300 dimensiones por
from gensim.models import KeyedVectors
```

```
glove = KeyedVectors.load_word2vec_format("/kaggle/input/glove-fasttext-embedd
glove.vectors.shape
```

Out[8]:

```
(400000, 300)
```

The word embeddings have been pretrained in a different corpus, so it would be a good idea to estimate how good our tokenization matches the GloVe vocabulary.

Se defiine un filtro para asegurarse que no se sobre eentrene el modelo, asegurandoce que solo se tomen en cuenta las palabras que se repitan mas de diez veces

In [9]:

```
from collections import Counter

def count_unknown_words(data, vocabulary):
    counter = Counter()
    for row in tqdm(data):
        counter.update(tok for tok in row if tok not in vocabulary)
    return counter

#Descubrimos cuantas veces cada palabra desconocida ocurre en el corpus
c = count_unknown_words(train_df['tokens'], glove.key_to_index)

#Encontramos el numero total de tokens en el corpus
total_tokens = train_df['tokens'].map(len).sum()

#Nos muestra estadisticas de los datos desconocidos
unk_tokens = sum(c.values())
percent_unk = unk_tokens / total_tokens
distinct_tokens = len(list(c))

print(f'total number of tokens: {total_tokens:,}')
print(f'number of unknown tokens: {unk_tokens:,}')
print(f'number of distinct unknown tokens: {distinct_tokens:,}')
print(f'percentage of unkown tokens: {percent_unk:.2%}')
print('top 50 unknown words:')
for token, n in c.most_common(10):
    print(f'\t{n}\t{token}')
```

```
  0%|          | 0/84000 [00:00<?, ?it/s]
total number of tokens: 3,691,911
number of unknown tokens: 46,427
number of distinct unknown tokens: 18,956
percentage of unkown tokens: 1.26%
top 50 unknown words:
        2055    /b
        1502    href=
        1501    /a
        1280    //www.investor.reuters.com/fullquote.aspx
        1280    target=/stocks/quickinfo/fullquote
        417     /p
        356     newsfactor
        340     cbs.mw
        300     color=
        291     face=
```

Glove embeddings seem to have a good coverage on this dataset -- only 1.25% of the tokens in the dataset are unknown, i.e., don't appear in the GloVe vocabulary.

Still, we will need a way to handle these unknown tokens. Our approach will be to add a new embedding to GloVe that will be used to represent them. This new embedding will be initialized as the average of all the GloVe embeddings.

We will also add another embedding, this one initialized to zeros, that will be used to pad the sequences of tokens so that they all have the same length. This will be useful when we train with mini-batches.

Se usan los nuevos embeddings con un tokken "Unk" desconocido y padding "pad" para saber como tratar con palabras desconocidas y de relleno, despues se generan IDs de cada uno de los tokens para que el modelo pueda procesarlos por embeddings

In [10]:

```python
# string values Correspondientes a los nuevos enbeddings
unk_tok = '[UNK]'
pad_tok = '[PAD]'

# Los embeddings empiezan con un valor
unk_emb = glove.vectors.mean(axis=0)
pad_emb = np.zeros(300)

# Los añade al Glove
glove.add_vectors([unk_tok, pad_tok], [unk_emb, pad_emb])

# Obtiene los ID de los tokens de los nuevos embeddings
unk_id = glove.key_to_index[unk_tok]
pad_id = glove.key_to_index[pad_tok]

unk_id, pad_id
```

Out[10]:

```
(400000, 400001)
```

In [11]:

```python
from sklearn.model_selection import train_test_split

train_df, dev_df = train_test_split(train_df, train_size=0.8)
train_df.reset_index(inplace=True)
dev_df.reset_index(inplace=True)
```

We will now add a new column to our dataframe that will contain the padded sequences of token ids.

Se genera una nueva columna con los tokens IDs de las palabras de relleno

In [12]:

```python
#Seleccionamos a las palabras que se repiitan mas de 10 veces a lo largo de los
threshold = 10
tokens = train_df['tokens'].explode().value_counts()
vocabulary = set(tokens[tokens > threshold].index.tolist())
print(f'vocabulary size: {len(vocabulary):,}')
```

```
vocabulary size: 14,309
```

In [13]:

```python
# Encontramos el largo de la palabra mas grande
max_tokens = train_df['tokens'].map(len).max()

# return unk_id for infrequent tokens too
```

```python
def get_id(tok):
    if tok in vocabulary:
        return glove.key_to_index.get(tok, unk_id)
    else:
        return unk_id

# function that gets a list of tokens and returns a list of token ids,
# with padding added accordingly
def token_ids(tokens):
    tok_ids = [get_id(tok) for tok in tokens]
    pad_len = max_tokens - len(tok_ids)
    return tok_ids + [pad_id] * pad_len

# add new column to the dataframe
train_df['token ids'] = train_df['tokens'].progress_map(token_ids)
train_df
```

```
  0%|          | 0/67200 [00:00<?, ?it/s]
```

Out[13]:

	index	class index	class	title	description	text	tokens	token ids
0	109275	4	Sci/Tech	Mmo2, Lucent to deploy converged fixed-mobile ...	UK mobile operator Mmo2 and US telecoms equipm...	mmo2, lucent to deploy converged fixed-mobile ...	[mmo2, ,, lucent, to, deploy, converged, fixed...	[122597, 1, 15725, 4, 8169, 21252, 400000, 849...
1	89047	3	Business	Spitzer Plans to Sue Insurer	New York Attorney General Eliot Spitzer will f...	spitzer plans to sue insurer new york attorney...	[spitzer, plans, to, sue, insurer, new, york, ...	[12185, 559, 4, 6415, 10646, 50, 196, 1223, 21...
2	118050	1	World	Britain Cannot Detain Terror Suspects Indefini...	Nine Law Lords ruled in favour of a group of m...	britain cannot detain terror suspects indefini...	[britain, can, not, detain, terror, suspects, ...	[695, 86, 36, 14097, 1974, 2330, 9595, 45, 202...
3	106813	1	World	Belgrade attack #39;was road rage #39;	A feared assassination attempt on Serbia #39;s...	belgrade attack #39;was road rage #39; a fear...	[belgrade, attack, #, 39, ;, was, road, rage, ...	[4038, 436, 2749, 3403, 89, 15, 586, 9012, 274...
4	84844	3	Business	Arctic Thaw Threatens People, Polar Bears	OSLO (Reuters) - Global warming is heating th...	arctic thaw threatens people, polar bears osl...	[arctic, thaw, threatens, people, ,, polar, be...	[7574, 20189, 6805, 69, 1, 10158, 4509, 6737, ...
...	...	...	...	...	...	...	...	...
67195	67493	3	Business	Jeans Maker VF Sees Earns Up 24 Percent (Reuters)	Reuters - VF Corp , the world's largest\jeans ...	jeans maker vf sees earns up 24 percent (reute...	[jeans, maker, vf, sees, earns, up, 24, percen...	[400000, 2737, 400000, 3109, 12803, 60, 795, 7...
67196	58333	3	Business	Temasek Makes S$7.4 Bln Profit, Gets Top AAA ...	Temasek Holdings Pte earned S$7.4 billion ($...	temasek makes s $7.4 bln profit, gets top aaa ...	[temasek, makes, s, $, 7.4, bln, profit, ,, ge...	[400000, 907, 1534, 80, 14321, 17494, 1269, 1,...
67197	112554	3	Business	Local gamer: Grand Theft Auto #39; steals the ...	Just how excited is Justin Field about the new...	local gamer: grand theft auto #39; steals the ...	[local, gamer, :, grand, theft, auto, #, 39, ;...	[250, 400000, 45, 1063, 6539, 2612, 2749, 3403...

	index	class index	class	title	description	text	tokens	token ids
67198	116840	3	Business	Sprint, Nextel Agree To Merge	The deal, valued at $35 billion, will create ...	sprint, nextel agree to merge the deal, valued...	[sprint, ,, nextel, agree, to, merge, the, dea...	[5514, 1, 17774, 2137, 4, 9194, 0, 435, 1, 595...
67199	34067	3	Business	Export Cut to China Seen as Clever Strategy on...	Yukos, the Russian oil giant, is playing a wea...	export cut to china seen as clever strategy on...	[export, cut, to, china, seen, as, clever, str...	[2467, 611, 4, 132, 541, 19, 11114, 1747, 13, ...

67200 rows × 8 columns

```
In [14]: max_tokens = dev_df['tokens'].map(len).max()
         dev_df['token ids'] = dev_df['tokens'].progress_map(token_ids)
         dev_df
```

```
  0%|          | 0/16800 [00:00<?, ?it/s]
```

Out[14]:

	index	class index	class	title	description	text	tokens	token ids
0	111352	4	Sci/Tech	Canon loses printer recycling case	Refilling, reselling cartridges doesn't violat...	canon loses printer recycling case refilling, ...	[canon, loses, printer, recycling, case, refil...	[9579, 7233, 13568, 12520, 305, 400000, 1, 400...
1	102053	4	Sci/Tech	'EICU' Lets Doctors Monitor Many Patients (AP)	AP - Your next doctor could be keeping an eye ...	'eicu' lets doctors monitor many patients (ap)...	['eicu, ', lets, doctors, monitor, many, patie...	[400000, 57, 8235, 1768, 3933, 109, 1615, 23, ...
2	50868	4	Sci/Tech	Yahoo CEO Sees No Need to Join Media Merger Fr...	Reuters - In an era of widespread media\consol...	yahoo ceo sees no need to join media merger fr...	[yahoo, ceo, sees, no, need, to, join, media, ...	[6600, 3695, 3109, 84, 408, 4, 1429, 493, 3176...
3	27469	2	Sports	Sportsview: Chargers Are Surprise Winners (AP)	AP - So the San Diego Chargers shocked the NFL...	sportsview: chargers are surprise winners (ap)...	[sportsview, :, chargers, are, surprise, winne...	[400000, 45, 12104, 32, 2661, 2945, 23, 1582, ...
4	66091	3	Business	Stocks Fall on J.P. Morgan Chase and Oil	NEW YORK (Reuters) - U.S. stocks fell on Wedn...	stocks fall on j.p. morgan chase and oil new ...	[stocks, fall, on, j.p., morgan, chase, and, o...	[895, 807, 13, 12227, 3123, 4212, 5, 316, 50, ...
...	...	...	...	...	...	...	...	...
16795	109691	1	World	Former Marine Testifies to Atrocities in Iraq	A former U.S. Marine staff sergeant testified ...	former marine testifies to atrocities in iraq ...	[former, marine, testifies, to, atrocities, in...	[157, 2266, 27149, 4, 8088, 6, 233, 7, 157, 99...
16796	35541	4	Sci/Tech	Blogging the Story Alive	Bloggers force CBS News to admit to a serious ...	blogging the story alive bloggers force cbs ne...	[blogging, the, story, alive, bloggers, force,...	[30031, 0, 523, 2977, 19305, 352, 3286, 172, 4...
16797	106135	3	Business	Getting your report	Consumers in Arizona and 12 other Western stat...	getting your report consumers in arizona and 1...	[getting, your, report, consumers, in, arizona...	[881, 392, 255, 2034, 6, 2203, 5,

	index	class index	class	title	description	text	tokens	token ids
								421, 68, 556...
16798	61875	3	Business	GM reports poor quarterly profits	DETROIT: General Motors Corp posted on Thursda...	gm reports poor quarterly profits detroit: gen...	[gm, reports, poor, quarterly, profits, detroi...	[2907, 687, 992, 6206, 2243, 2369, 45, 216, 46...
16799	40321	3	Business	For Cingular, Becoming No. 1 Also Poses Risks	The union of Cingular and AT T Wireless would ...	for cingular, becoming no. 1 also poses risks ...	[for, cingular, ,, becoming, no, ., 1, also, p...	[10, 31779, 1, 1663, 84, 2, 176, 52, 9734, 334...

16800 rows × 8 columns

Now we will get a numpy 2-dimensional array corresponding to the token ids, and a 1-dimensional array with the gold classes. Note that the classes are one-based (i.e., they start at one), but we need them to be zero-based, so we need to subtract one from this array.

Creamos una clase especial para Pytorch, con la que pueda entrar por medio de indice a loos pares de datos

In [15]:
```
from torch.utils.data import Dataset
# Clase personalizada de Dataset para PyTorch que permite manejar pares de dat
class MyDataset(Dataset):
    def __init__(self, x, y):
        self.x = x
        self.y = y

    def __len__(self):
        return len(self.y)

    def __getitem__(self, index):
        x = torch.tensor(self.x[index])
        y = torch.tensor(self.y[index])
        return x, y
```

Next, we construct our PyTorch model, which is a feed-forward neural network with two layers:

Generamos el modelo de PyThorch con dos capas neuronales, con un feed foward, el cual toma en cuenta el padding para poder mejorar la presicion del embedding

In [16]:
```
from torch import nn
import torch.nn.functional as F

class Model(nn.Module):
```

```
# Constructor de la clase Model. Define las capas del modelo, incluyendo la cap

    def __init__(self, vectors, pad_id, hidden_dim, output_dim, dropout):
        super().__init__()
        # embeddings must be a tensor
        if not torch.is_tensor(vectors):
            vectors = torch.tensor(vectors)
        # keep padding id
        self.padding_idx = pad_id
        # embedding layer
        self.embs = nn.Embedding.from_pretrained(vectors, padding_idx=pad_id)
        # feedforward layers
        self.layers = nn.Sequential(
            nn.Dropout(dropout),
            nn.Linear(vectors.shape[1], hidden_dim),
            nn.ReLU(),
            nn.Dropout(dropout),
            nn.Linear(hidden_dim, output_dim),
        )

# Método forward que define cómo se procesa el input a través del modelo.
    def forward(self, x):
        # get boolean array with padding elements set to false
        not_padding = torch.isin(x, self.padding_idx, invert=True)
        # get lengths of examples (excluding padding)
        lengths = torch.count_nonzero(not_padding, axis=1)
        # get embeddings
        x = self.embs(x)
        # calculate means
        x = x.sum(dim=1) / lengths.unsqueeze(dim=1)
        # pass to rest of the model
        output = self.layers(x)
        # calculate softmax if we're not in training mode
        #if not self.training:
        #    output = F.softmax(output, dim=1)
        return output
```

Next, we implement the training procedure. We compute the loss and accuracy on the development partition after each epoch.

Se cargan los datos de entrenamiento para su analisis, se establece el tamaño, el numero de lootes, la tasa de aprendizahe y se usa el iniciadr ADAM, por cada lote se obtiene la perdida y la precision, lo que facilita su analisis posterior

In [17]:

```
from torch import optim
from torch.utils.data import DataLoader
from sklearn.metrics import accuracy_score

# hyperparameters
lr = 1e-3
weight_decay = 0
batch_size = 500
shuffle = True
n_epochs = 5
hidden_dim = 50
output_dim = len(labels)
dropout = 0.1
vectors = glove.vectors
```

```
# Inicia el modelo, la funcion de perdida, el optimiizador y el cargador de da
model = Model(vectors, pad_id, hidden_dim, output_dim, dropout).to(device)
loss_func = nn.CrossEntropyLoss()
optimizer = optim.Adam(model.parameters(), lr=lr, weight_decay=weight_decay)
train_ds = MyDataset(train_df['token ids'], train_df['class index'] - 1)
train_dl = DataLoader(train_ds, batch_size=batch_size, shuffle=shuffle)
dev_ds = MyDataset(dev_df['token ids'], dev_df['class index'] - 1)
dev_dl = DataLoader(dev_ds, batch_size=batch_size, shuffle=shuffle)

train_loss = []
train_acc = []

dev_loss = []
dev_acc = []

# Entrena el modelo
for epoch in range(n_epochs):
    losses = []
    gold = []
    pred = []
    model.train()
    for X, y_true in tqdm(train_dl, desc=f'epoch {epoch+1} (train)'):
        # clear gradients
        model.zero_grad()
        # send batch to right device
        X = X.to(device)
        y_true = y_true.to(device)
        # predict label scores
        y_pred = model(X)
        # compute loss
        loss = loss_func(y_pred, y_true)
        # accumulate for plotting
        losses.append(loss.detach().cpu().item())
        gold.append(y_true.detach().cpu().numpy())
        pred.append(np.argmax(y_pred.detach().cpu().numpy(), axis=1))
        # backpropagate
        loss.backward()
        # optimize model parameters
        optimizer.step()
    train_loss.append(np.mean(losses))
    train_acc.append(accuracy_score(np.concatenate(gold), np.concatenate(pred)

    model.eval()
    with torch.no_grad():
        losses = []
        gold = []
        pred = []
        for X, y_true in tqdm(dev_dl, desc=f'epoch {epoch+1} (dev)'):
            X = X.to(device)
            y_true = y_true.to(device)
            y_pred = model(X)
            loss = loss_func(y_pred, y_true)
            losses.append(loss.cpu().item())
            gold.append(y_true.cpu().numpy())
            pred.append(np.argmax(y_pred.cpu().numpy(), axis=1))
        dev_loss.append(np.mean(losses))
        dev_acc.append(accuracy_score(np.concatenate(gold), np.concatenate(pre
```

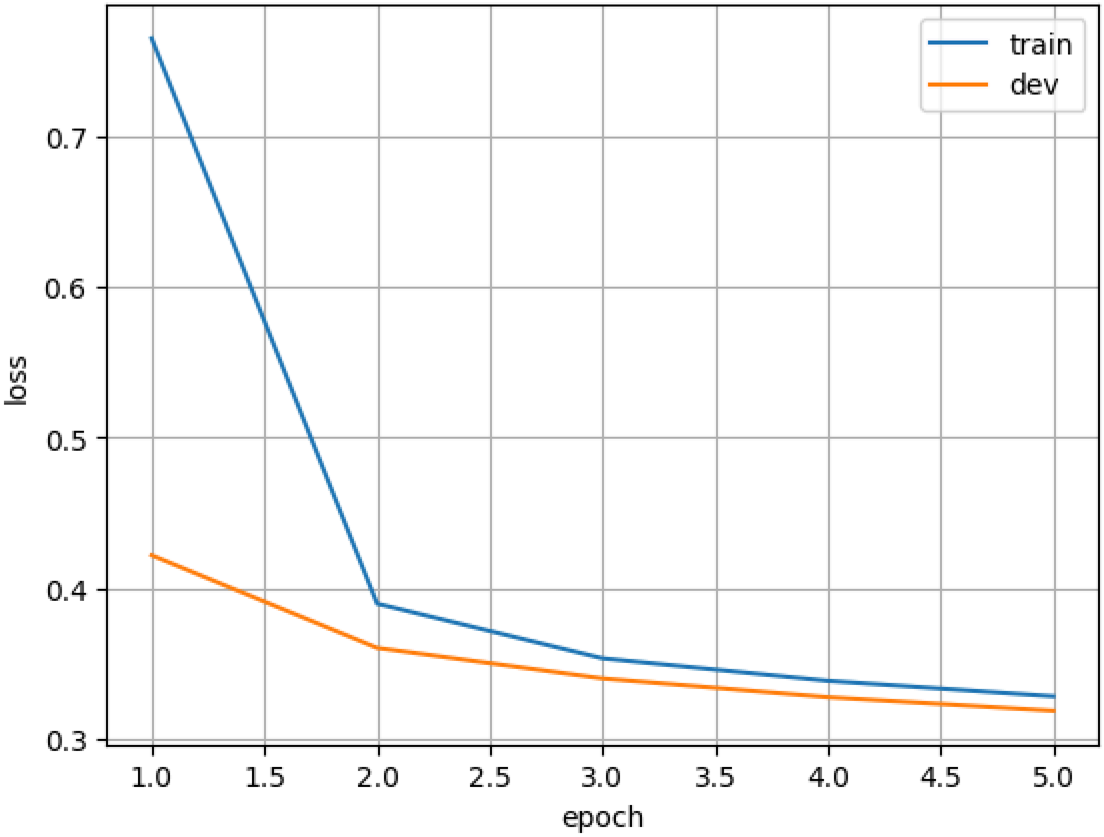
```
epoch 1 (train):   0%|          | 0/135 [00:00<?, ?it/s]
epoch 1 (dev):   0%|          | 0/34 [00:00<?, ?it/s]
epoch 2 (train):   0%|          | 0/135 [00:00<?, ?it/s]
epoch 2 (dev):   0%|          | 0/34 [00:00<?, ?it/s]
epoch 3 (train):   0%|          | 0/135 [00:00<?, ?it/s]
epoch 3 (dev):   0%|          | 0/34 [00:00<?, ?it/s]
epoch 4 (train):   0%|          | 0/135 [00:00<?, ?it/s]
epoch 4 (dev):   0%|          | 0/34 [00:00<?, ?it/s]
epoch 5 (train):   0%|          | 0/135 [00:00<?, ?it/s]
epoch 5 (dev):   0%|          | 0/34 [00:00<?, ?it/s]
```

Let's plot the loss and accuracy on dev:

Se graffica la perdida y la exactitud

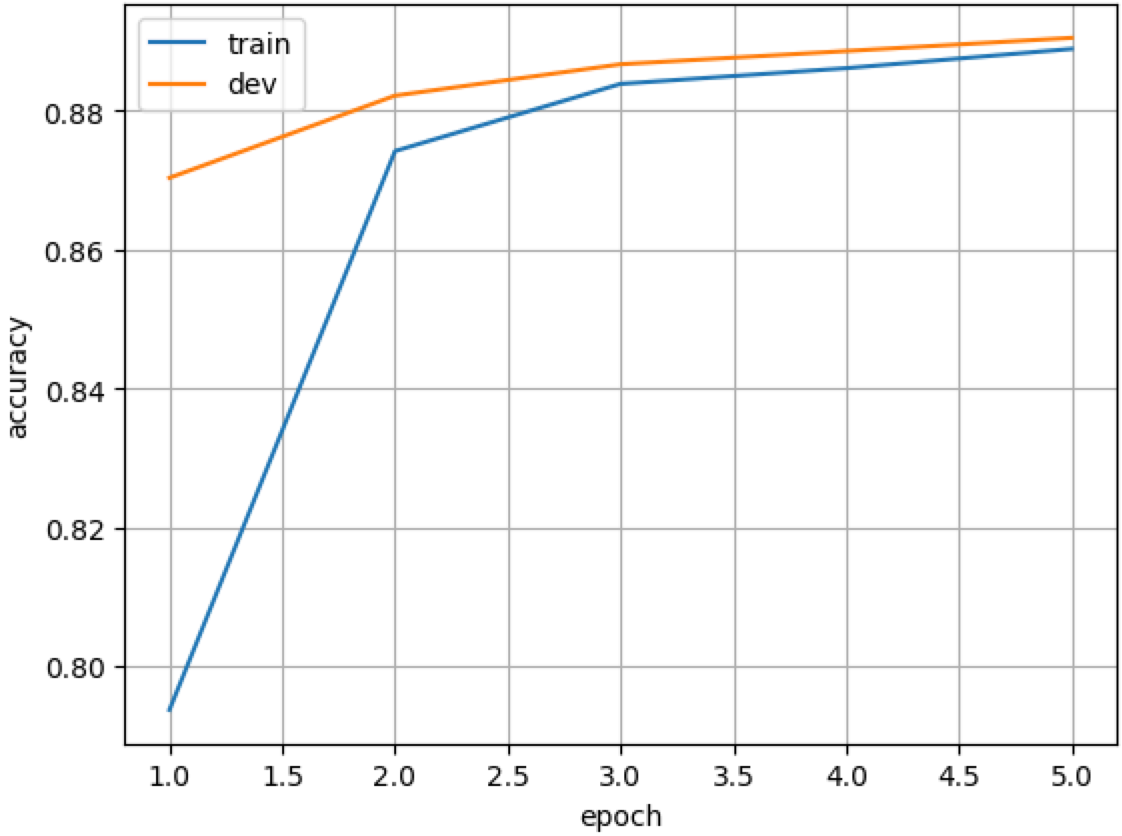
In [18]:

```
#Grafica la perdida en dev y la exactitud on dev
import matplotlib.pyplot as plt
%matplotlib inline

x = np.arange(n_epochs) + 1

plt.plot(x, train_loss)
plt.plot(x, dev_loss)
plt.legend(['train', 'dev'])
plt.xlabel('epoch')
plt.ylabel('loss')
plt.grid(True)
```



In [19]:

```
plt.plot(x, train_acc)
plt.plot(x, dev_acc)
plt.legend(['train', 'dev'])
plt.xlabel('epoch')
plt.ylabel('accuracy')
plt.grid(True)
```



Next, we evaluate on the testing partition:

Se hace con los datos de test para asegurarnos que sea funcional

In [20]:
```
# Repite todo lo anterior, pero con los datos de test
test_df = pd.read_csv('/kaggle/input/agnews-pytorch-simple-embed-classif-90/AG_
test_df.columns = ['class index', 'title', 'description']
test_df['text'] = test_df['title'].str.lower() + " " + test_df['description'].s
test_df['text'] = test_df['text'].str.replace('\\', ' ', regex=False)
test_df['tokens'] = test_df['text'].progress_map(word_tokenize)
max_tokens = dev_df['tokens'].map(len).max()
test_df['token ids'] = test_df['tokens'].progress_map(token_ids)
```

```
  0%|          | 0/7600 [00:00<?, ?it/s]
  0%|          | 0/7600 [00:00<?, ?it/s]
```

In [21]:
```
from sklearn.metrics import classification_report

# Evalua el modelo
model.eval()

dataset = MyDataset(test_df['token ids'], test_df['class index'] - 1)
data_loader = DataLoader(dataset, batch_size=batch_size)
y_pred = []

# No guarda las gradientes
with torch.no_grad():
    for X, _ in tqdm(data_loader):
        X = X.to(device)
        # predict one class per example
        y = torch.argmax(model(X), dim=1)
        # convert tensor to numpy array (sending it back to the cpu if needed)
        y_pred.append(y.cpu().numpy())
        # print results
    print(classification_report(dataset.y, np.concatenate(y_pred), target_names
```

```
  0%|          | 0/16 [00:00<?, ?it/s]
              precision    recall  f1-score   support

       World       0.92      0.87      0.90      1900
      Sports       0.95      0.97      0.96      1900
    Business       0.83      0.86      0.85      1900
    Sci/Tech       0.87      0.86      0.86      1900

    accuracy                           0.89      7600
   macro avg       0.89      0.89      0.89      7600
weighted avg       0.89      0.89      0.89      7600
```