ΕΠΕΞΕΡΓΑΣΙΑ ΦΥΣΙΚΗΣ ΓΛΩΣΣΑΣ

Εργασία 2

A. Word embeddings

Χρησιμοποιήστε τα προ-εκπαιδευμένα word embeddings word2vec (word2vec-google-news-300) και GloVe (glove-wiki-gigaword-300) που περιλαμβάνονται στο genism[1].

1. Σύμφωνα με το word2vec, ποιες είναι οι 10 πιο κοντινές λέξεις για την καθεμία από τις λέξεις: 'car', 'jaguar', 'Jaguar', 'facebook'; Ποιες είναι οι αντίστοιχες λέξεις σύμφωνα με το GloVe; Πόσες κοινές λέξεις υπάρχουν στα αποτελέσματα του word2vec και του GloVe;
2. Επαναλάβετε το προηγούμενο ερώτημα για 4 λέξεις τις δικής σας επιλογής.
3. Σύμφωνα με το word2vec/GloVe, ποιες είναι οι 10 πιο κοντινές λέξεις για την λέξη 'student'; Ποια είναι η αντίστοιχη λίστα των 10 πιο κοντινών λέξεων αν δεν επιθυμούμε να συσχετίζονται με φοιτητές πανεπιστημίου; Ποια είναι η αντίστοιχη λίστα αν επιθυμούμε να μην σχετίζονται με μαθητές Δημοτικού-Γυμνασίου-Λυκείου;
4. Δεδομένου ότι το $\vec{x}$ αντιστοιχεί στο διάνυσμα (word embedding) της λέξης *x* βρείτε ποιες είναι οι 2 πιο κοντινές λέξεις για τις ακόλουθες αναλογίες. Για την κάθε αναλογία, να αναφέρετε την διατύπωση του ερωτήματος, τις 2 πιο κοντινές λέξεις που προκύπτουν (εξαιρώντας τις 3 λέξεις που χρησιμοποιούνται στην συγκεκριμένη αναλογία) και την ομοιότητά τους σύμφωνα με το word2vec και τα αντίστοιχα για το GloVe:
   i. $\overrightarrow{king} - \overrightarrow{man} + \overrightarrow{woman}$ = ?
   ii. $\overrightarrow{France} - \overrightarrow{Paris} + \overrightarrow{Tokyo}$ = ?
   iii. $\overrightarrow{trees} - \overrightarrow{apples} + \overrightarrow{grapes}$ = ?
   iv. $\overrightarrow{swimming} - \overrightarrow{walking} + \overrightarrow{walked}$ = ?
   v. $\overrightarrow{doctor} - \overrightarrow{father} + \overrightarrow{mother}$ = ?
5. Επαναλάβετε το προηγούμενο ερώτημα με 3 αναλογίες της δικής σας επιλογής και σχολιάστε τα αποτελέσματα.

Χρήσιμα links:

https://radimrehurek.com/gensim/auto_examples/tutorials/run_word2vec.html

[1] https://radimrehurek.com/gensim/models/word2vec.html#pretrained-models

B. Traditional Text Classification

Θα γίνει χρήση του **AG News Topic Classification dataset**[2]. Τα κείμενα προς ταξινόμηση είναι ειδήσεις (τίτλος και σύντομη περιγραφή είδησης) που ανήκουν σε 4 κατηγορίες: World, Sports, Business, Sci/Tech. Συνολικά, στο dataset υπάρχουν 127.600 ειδήσεις ισοκατανεμημένες στις 4 κατηγορίες που είναι χωρισμένες σε δύο υποσύνολα: training set (120.000 ειδήσεις) και test set (7.600 ειδήσεις). Δείτε την περιγραφή και αποτελέσματα πειραμάτων που αφορούν το AG_NEWS dataset εδώ:

https://papers.nips.cc/paper/2015/file/250cf8b51c773f3f8dc8b4be867a9a02-Paper.pdf

https://paperswithcode.com/sota/text-classification-on-ag-news

1. Βασιζόμενοι στη βιβλιοθήκη συναρτήσεων μηχανικής μάθησης scikit-learn, υλοποιήστε τις ακόλουθες προσεγγίσεις ταξινόμησης κειμένου (σε όλες τις περιπτώσεις, τα κείμενα να μετατραπούν σε lowercase):

   - Multinomial Naïve Bayes[3] με αναπαράσταση των κειμένων σύμφωνα με την προσέγγιση tf-idf word uni-grams[4].
   - Multinomial Naïve Bayes με αναπαράσταση των κειμένων σύμφωνα με την προσέγγιση tf-idf character tri-grams.
   - Support Vector Machines[5] (γραμμικό kernel και C=1) με αναπαράσταση των κειμένων σύμφωνα με την προσέγγιση tf-idf word uni-grams.
   - Support Vector Machines (γραμμικό kernel και C=1) με αναπαράσταση των κειμένων σύμφωνα με την προσέγγιση tf-idf character tri-grams.

2. Συμπληρώστε τον ακόλουθο πίνακα για τα παραπάνω μοντέλα χρησιμοποιώντας την επίδοσή τους (accuracy on test set), το μέγεθος της διάστασης της αναπαράστασης (πλήθος διαφορετικών n-grams) που χρησιμοποιήθηκε, και το χρονικό κόστος σε δευτερόλεπτα.

	NB (word 1-grams)	NB (char 3-grams)	SVM (word 1-grams)	SVM (char 3-grams)
Accuracy				
Dimensionality				
Time cost				

3. Υπάρχουν συγκεκριμένα κείμενα στο test set που ΟΛΑ τα παραπάνω μοντέλα κάνουν λάθος στην ταξινόμησή τους; Δείξτε τα περιεχόμενα ενός τέτοιου κειμένου. Πόσα είναι αυτά τα κείμενα ανά κατηγορία (World, Sports, Business, Sci/Tech); Για αυτά τα κείμενα, ποιο είναι το πιο συχνό ζευγάρι σωστής κατηγορίας – λάθος πρόβλεψης;

Χρήσιμα Links:

https://scikit-learn.org/stable/modules/svm.html

[2] https://www.kaggle.com/datasets/amananandrai/ag-news-classification-dataset
[3] https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html
[4] https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html
[5] https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html

Γ. Text Classification with RNNs

Δίνεται κώδικας (rnn.py) που υλοποιεί ένα νευρωνικό δίκτυο με ανατροφοδότηση (Recurrent Neural Network) με βάση το **PyTorch**[6] και το εφαρμόζει στο **AG News Topic Classification dataset**[7]. Γίνεται επίσης χρήση συναρτήσεων από τις βιβλιοθήκες της Python: **torchtext**, **sklearn**, **pandas, tqdm**. Συνιστάται η χρήση του Anaconda[8] για την εύκολη εγκατάσταση όλων των παραπάνω βιβλιοθηκών στο μηχάνημά σας. Εναλλακτικά, μπορείτε να εκτελέσετε τον κώδικα online δημιουργώντας κατάλληλο Kaggle Notebook[9] ή Colab notebook[10].

Αρχικά, όλα τα κείμενα χωρίζονται σε tokens και διατηρούνται το πολύ 25 tokens από κάθε κείμενο. Σε περίπτωση που κάποια κείμενα έχουν λιγότερα από 25 tokens, συμπληρώνονται τα κενά με ένα ειδικό σύμβολο (<PAD>), μία διαδικασία που καλείται padding. Για παράδειγμα, η είδηση:

> E-mail scam targets police chief Wiltshire Police warns about "phishing" after its fraud squad chief was targeted.

Αφού προ-επεξεργαστεί και χωριστεί σε tokens γίνεται:

> ['e-mail', 'scam', 'targets', 'police', 'chief', 'wiltshire', 'police', 'warns', 'about', 'phishing', 'after', 'its', 'fraud', 'squad', 'chief', 'was', 'targeted', '.']

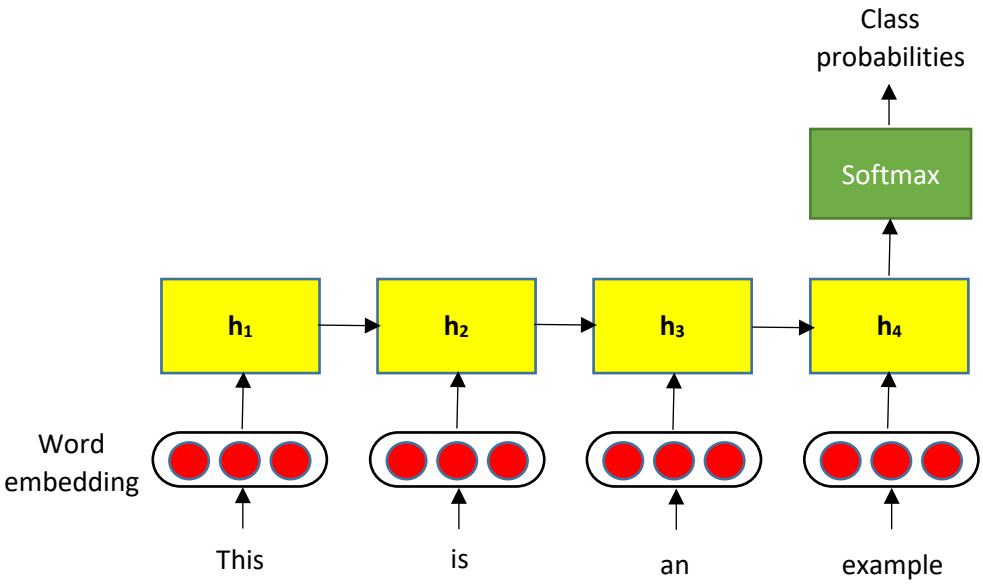
Επειδή έχει 18 tokens συμπληρώνονται οι υπόλοιπες θέσεις μέχρι το 25 με το ειδικό σύμβολο (padding):

> ['e-mail', 'scam', 'targets', 'police', 'chief', 'wiltshire', 'police', 'warns', 'about', 'phishing', 'after', 'its', 'fraud', 'squad', 'chief', 'was', 'targeted', '.', <PAD>, <PAD>, <PAD>, <PAD>, <PAD>, <PAD>, <PAD>]

Δημιουργείται ένα λεξιλόγιο (vocabulary) στο οποίο συμμετέχουν όλα τα tokens που εμφανίζονται τουλάχιστον 10 φορές. Όλες οι εμφανίσεις λέξεων τόσο στο training set όσο και στο test set που δεν ανήκουν στο λεξιλόγιο αντικαθίστανται από ένα άλλο ειδικό σύμβολο (<UNK>). Για παράδειγμα, αν θεωρήσουμε ότι στην παραπάνω είδηση όλα τα tokens εκτός από το 'wiltshire' περιλαμβάνονται στο λεξιλόγιο, η λίστα των tokens θα μετατραπεί ως εξής:

['e-mail', 'scam', 'targets', 'police', 'chief', <UNK>, 'police', 'warns', 'about', 'phishing', 'after', 'its', 'fraud', 'squad', 'chief', 'was', 'targeted', '.', <PAD>, <PAD>, <PAD>, <PAD>, <PAD>, <PAD>, <PAD>]

[6] https://pytorch.org/tutorials/beginner/basics/intro.html
[7] https://www.kaggle.com/datasets/amananandrai/ag-news-classification-dataset
[8] https://www.anaconda.com/
[9] https://www.kaggle.com/code
[10] https://research.google.com/colaboratory/faq.html



Τα κείμενα που βρίσκονται στην παραπάνω μορφή χρησιμοποιούνται για να εκπαιδεύσουν ένα RNN μονής κατεύθυνσης (1RNN) όπως φαίνεται στο παραπάνω σχήμα. Αρχικά, κάθε λέξη αντιστοιχίζεται σε ένα embedding και στη συνέχεια το RNN δέχεται ως είσοδο την ακολουθία από τα embeddings και παράγει στο τέλος της ακολουθίας την έξοδο η οποία μέσω της softmax μετατρέπεται σε πιθανότητες που επιτρέπουν την ταξινόμηση του κειμένου (στην κατηγορία με την μέγιστη τιμή πιθανότητας). Τα embeddings μαθαίνονται στη διάρκεια της εκπαίδευσης μαζί με τις υπόλοιπες παραμέτρους του δικτύου.

Χρησιμοποιείται η μέθοδος Adam για την προσαρμογή του ρυθμού εκπαίδευσης και το cross-entropy loss ως κριτήριο μάθησης. Ακόμη, έχουν οριστεί οι ακόλουθες τιμές υπερ-παραμέτρων:

EPOCHS = 15 (διάρκεια εκπαίδευσης του δικτύου)
LEARNING_RATE = 1e-3 (ρυθμός μάθησης)
BATCH_SIZE = 1024 (μέγεθος μίνι-συστάδων, mini-batches, για την εκπαίδευση του δικτύου)
EMBEDDING_DIM = 100 (διάσταση των embeddings)
HIDDEN_DIM = 64 (διάσταση του κρυμμένου επιπέδου)

Στη διάρκεια της εκπαίδευσης του δικτύου, στο τέλος κάθε εποχής, εμφανίζονται πληροφορίες που αφορούν την επίδοση του μοντέλου στο training set (loss) ενώ μετά την ολοκλήρωση της εκπαίδευσης το τελικό μοντέλο αξιολογείται στο test set και εμφανίζονται αναλυτικά αποτελέσματα.

1. Χρησιμοποιώντας κάθε φορά κατάλληλες παραλλαγές του κώδικα, εξετάστε την επίδοση των παρακάτω μοντέλων:
   - RNN μονής κατεύθυνσης και ενός στρώματος (1RNN).
   - RNN διπλής κατεύθυνσης και ενός στρώματος (1Bi-RNN)
   - RNN διπλής κατεύθυνσης και δύο στρωμάτων (2Bi-RNN)
   - LSTM μονής κατεύθυνσης και ενός στρώματος (1LSTM).
   - LSTM διπλής κατεύθυνσης και ενός στρώματος (1Bi-LSTM)
   - LSTM διπλής κατεύθυνσης και δύο στρωμάτων (2Bi-LSTM)

Συμπληρώστε τον παρακάτω πίνακα με βάση την επίδοση (accuracy on test set), την πολυπλοκότητα των μοντέλων (αριθμός παραμέτρων) και το χρονικό κόστος τους (μέσος όρος δευτερολέπτων ανά εποχή):

	1RNN	1Bi-RNN	2Bi-RNN	1LSTM	1Bi-LSTM	2Bi-LSTM
Accuracy						
Parameters						
Time cost						

Σχολιάστε τα αποτελέσματα που προέκυψαν. Πώς επηρρεάζεται η επίδοση του μοντέλου από την πολυπλοκότητά του; Πώς επηρρεάζεται το χρονικό κόστος από την πολυπλοκότητα του μοντέλου. Κατά πόσο βοηθάει η χρήση 2 στρωμάτων στην βελτίωση της επίδοσης των μοντέλων;

2. Υπάρχουν συγκεκριμένα κείμενα στο test set που ΟΛΑ τα παραπάνω μοντέλα κάνουν λάθος στην ταξινόμησή τους; Δείξτε τα περιεχόμενα ενός τέτοιου κειμένου. Πόσα είναι αυτά τα κείμενα ανά κατηγορία (World, Sports, Business, Sci/Tech); Για αυτά τα κείμενα, ποιο είναι το πιο συχνό ζευγάρι σωστής κατηγορίας – λάθος πρόβλεψης;

3. Αλλάξτε την τιμή της παραμέτρου MAX_WORDS από 25 σε 50 και επαναλάβετε το ερώτημα 1. Δείξτε τον νέο πίνακα με τα αποτελέσματα και σχολιάστε τα σε σύγκριση με αυτά του ερωτήματος 1. Πώς επηρρεάζει αυτή η αλλαγή τα μοντέλα RNN και πώς τα μοντέλα LSTM τόσο ως προς την επίδοσή τους όσο και ως προς την πολυπλοκότητά τους;

4. Τροποποιήστε τον αρχικό κώδικα ώστε τα embeddings του δικτύου να αρχικοποιούνται με προ-εκπαιδευμένα embeddings. Χρησιμοποιήστε τα pre-trained word embeddings glove-6B-100d[11]. Αναφέρετε ποιες αλλαγές έγιναν στον κώδικα και επαναλάβετε το ερώτημα 1 (με τιμή MAX_WORDS=25). Σχολιάστε τον τρόπο με τον οποίο επηρρεάζεται η επίδοση και η πολυπλοκότητα των μοντέλων με τη χρήση προ-εκπαιδευμένων embeddings.

5. Επαναλάβετε το προηγούμενο ερώτημα, αυτή τη φορά όμως τα embeddings αφού αρχικοποιηθούν με τα προ-εκπαιδευμένα GloVe embeddings δεν θα πρέπει να τροποποιούνται στη διάρκεια εκπαίδευσης του δικτύου (freeze embeddings). Αναφέρετε τις αλλαγές που έγιναν στον κώδικα και σχολιάστε τα αποτελέσματα που προέκυψαν σε σύγκριση με αυτά του προηγούμενου ερωτήματος.

6. Τροποποιήστε τον αρχικό κώδικα ώστε να μπορεί να εφαρμοστεί σε ένα άλλο dataset, το **IMDB movie review**[12] όπου 50k κριτικές ταινιών ταξινομούνται σε 2 κατηγορίες (θετικές αρνητικές). Χρησιμοποιήστε το 80% του dataset ως training set και το υπόλοιπο 20% ως test set. Επαναλάβετε το ερώτημα 1 για το IMDB dataset.

[11] https://nlp.stanford.edu/projects/glove/
είναι διαθέσιμα και στο **torchtext**: https://pytorch.org/text/stable/vocab.html#vectors
[12] https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews

Οδηγίες παράδοσης:

Η εργασία είναι **ατομική**. Πρέπει να υποβάλλετε μία αναφορά με τις απαντήσεις στα ερωτήματα της εργασίας καθώς και τον κώδικα που χρησιμοποιήσατε (τμήμα Α και Β) και μόνο τις αλλαγές που κάνατε στον κώδικα (για κάθε ερώτημα στο τμήμα Γ). Συστήνεται η χρήση **Jupyter Notebook**[13].

[13] https://www.dataquest.io/blog/jupyter-notebook-tutorial/