Οικονομικό Πανεπιστήμιο Αθηνών, Τμήμα Πληροφορικής
Μάθημα: Τεχνητή Νοημοσύνη, 2024–25
Διδάσκων: Ι. Ανδρουτσόπουλος

Ασκήσεις μελέτης της 22ης διάλεξης

22.1. Θέλουμε να χρησιμοποιήσουμε το ανατροφοδοτούμενο νευρωνικό δίκτυο (RNN) των διαφανειών 4 και 5, για να αναγνωρίζουμε ονόματα προσώπων, οργανισμών και τοποθεσιών. Χρησιμοποιούμε ετικέτες (κατηγορίες) B-I-O, όπως στην άσκηση 21.1, άρα 7 κατηγορίες (όχι τρεις). Το μέγεθος του λεξιλογίου είναι $|V| = 100.000$. Κάθε ένθεση λέξης (word embedding) είναι ένα διάνυσμα 300 διαστάσεων. Το κρυφό επίπεδο (η κατάσταση του RNN) αποτελείται από 500 νευρώνες, δηλαδή το $\vec{h}_i$ είναι διάνυσμα 500×1. Ποιες είναι οι διαστάσεις των $E, \vec{e}_i, W^{(h)}, W^{(e)}, W^{(o)}, \vec{o}_i$; Αιτιολογήστε τις απαντήσεις σας.

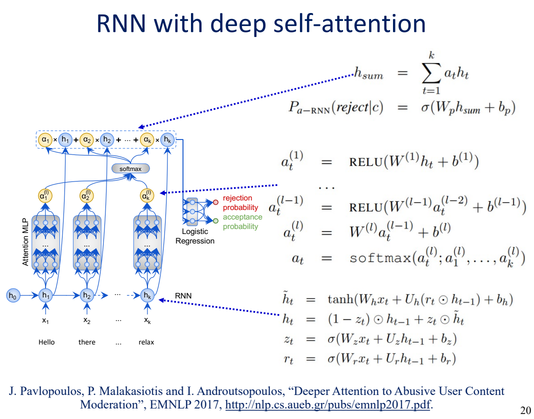
Απάντηση: Ο πίνακας E περιέχει (ως στήλες) τις ενθέσεις των 100.000 λέξεων του λεξιλογίου. Κάθε ένθεση λέξης είναι διάνυσμα (στήλη) 300 διαστάσεων. Άρα ο E έχει διαστάσεις 300×100.000.

Το διάνυσμα $\vec{e}_i$ είναι η ένθεση (embedding) της i-στής λέξης της εισόδου (π.χ. μιας πρότασης), άρα είναι διαστάσεων 300×1. Το ίδιο συμπέρασμα προκύπτει και από την παρατήρηση ότι ο πολλαπλασιασμός $E\vec{x}_i$ επιστρέφει την i-στή στήλη του πίνακα E.

Ο πίνακας $W^{(h)}$ έχει διαστάσεις 500×500, ενώ ο πίνακας $W^{(e)}$ έχει διαστάσεις 500×300, ώστε τα $W^{(h)}\vec{h}_{i-1}$ και $W^{(e)}\vec{e}_i$ να έχουν τις ίδιες διαστάσεις (500×1), να μπορούν να προστεθούν ($W^{(h)}\vec{h}_{i-1} + W^{(e)}\vec{e}_i$) και η νέα κατάσταση $\vec{h}_i = \tanh\left(W^{(h)}\vec{h}_{i-1} + W^{(e)}\vec{e}_i\right)$ να έχει πάλι διαστάσεις 500×1, όπως η προηγούμενη κατάσταση $\vec{h}_{i-1}$. Η tanh εφαρμόζεται σε κάθε στοιχείο του διανύσματος $W^{(h)}\vec{h}_{i-1} + W^{(e)}\vec{e}_i$, χωρίς να αλλάζει τις διαστάσεις του.

Ο πίνακας $W^{(o)}$ έχει διαστάσεις 7×500, ώστε ο πολλαπλασιασμός $W^{(0)}\vec{h}_i$ να παράγει διάνυσμα 7×1 με έναν πραγματικό αριθμό για κάθε κατηγορία. Η softmax στον υπολογισμό $\vec{o}_i = \mathrm{softmax}\left(W^{(o)}\vec{h}_i\right)$ μετατρέπει τους αριθμούς αυτούς σε κατανομή πιθανότητας (μία πιθανότητα για κάθε κατηγορία), χωρίς να αλλάζει τις διαστάσεις του $W^{(o)}\vec{h}_i$. Επομένως το $\vec{o}_i$ έχει και αυτό διαστάσεις 7×1.

22.2. (α) Στο νευρωνικό δίκτυο των διαφανειών 19–21 («RNN with deep self-attention»), οι **καταστάσεις** $h_1, h_2, \dots, h_k$ του RNN είναι διανύσματα **128 διαστάσεων** (η κάθε μία). Τα **κρυφά επίπεδα** (1), …, $(l-1)$ του Attention MLP έχουν **64 νευρώνες** το καθένα και οι έξοδοι των κρυφών επιπέδων είναι $a_t^{(1)}, \dots, a_t^{(l-1)}$. Το **επίπεδο εξόδου** του Attention MLP έχει **έναν μόνο νευρώνα** με έξοδο $a_t^{(l)}$. **Τι διαστάσεις θα έχουν οι πίνακες** $W^{(1)}, W^{(2)}, \dots, W^{(l)}$ **και τα διανύσματα** $b^{(1)}, b^{(2)}, \dots, b^{(l)}$; **Αιτιολογήστε σύντομα τις απαντήσεις σας**.



Διαστάσεις των $W^{(1)}, W^{(2)}, \dots, W^{(l)}$ και **αιτιολόγηση**:

Απάντηση: Ο πίνακας $W^{(1)}$ θα έχει διαστάσεις 64×128, ώστε να μετατρέπει το κάθε διάνυσμα $h_1, h_2, \dots, h_k$ (τις καταστάσεις του RNN), που έχουν διαστάσεις 128×1 το καθένα, σε διανύσματα 64×1, δηλαδή σε διανύσματα με τόσες συνιστώσες όσοι οι νευρώνες του επιπέδου (1) του MLP. Το κάθε παραγόμενο διάνυσμα 64×1 είναι η έξοδος $a_t^{(1)}$ (για $t = 1, \dots, k$) του επιπέδου (1) του MLP, όταν το MLP εφαρμόζεται στην αντίστοιχη κατάσταση h_t του RNN.

Οι πίνακες $W^{(2)}, \dots, W^{(l-1)}$ θα έχουν διαστάσεις 64×64, ώστε να μετατρέπουν τα διανύσματα διαστάσεων 64×1 που παράγει ο $W^{(1)}$ σε διανύσματα πάλι 64×1, δηλαδή σε διανύσματα με τόσες διαστάσεις όσοι οι νευρώνες των επιπέδων (2), …, $(l-1)$ του MLP. Το κάθε παραγόμενο διάνυσμα 64×1 είναι η έξοδος $a_t^{(2)}, \dots, a_t^{(l-1)}$ του επιπέδου (2), …, $(l-1)$, αντίστοιχα, του MLP.

Ο πίνακας $W^{(l)}$ θα έχει διαστάσεις 1×64 (δηλαδή θα είναι ένα διάνυσμα-γραμμή), ώστε να μετατρέπει το κάθε διάνυσμα 64×1 που παράγει ο $W^{(l-1)}$ σε έναν πραγματικό αριθμό (εκφυλισμένο διάνυσμα 1×1), δηλαδή να παράγει τους πραγματικούς αριθμούς $a_1^{(l)}, \dots, a_\kappa^{(l)}$ της διαφάνειας 21.

Διαστάσεις των $b^{(1)}, b^{(2)}, \dots, b^{(l)}$ και **αιτιολόγηση**:

Απάντηση: Το $b^{(1)}$ θα είναι ένα διάνυσμα 64×1, ώστε να προστίθεται στο διάνυσμα 64×1 που παράγει ο πολλαπλασιασμός $W^{(1)} h_t$ (για $t = 1, \dots, k$) και να παράγει το διάνυσμα $a_t^{(1)}$, που είναι επίσης διαστάσεων 64×1. (Κάθε συνάρτηση ενεργοποίησης, εδώ η ReLU, δεν αλλάζει τις διαστάσεις του διανύσματος στο οποίο εφαρμόζεται, απλά εφαρμόζεται σε κάθε στοιχείο του διανύσματος.)

Ομοίως τα $b^{(2)}, \dots, b^{(l-1)}$ θα είναι το καθένα ένα διάνυσμα 64×1, ώστε να προστίθεται στο διάνυσμα 64×1 που παράγει ο πολλαπλασιασμός $W^{(2)} a_t^{(1)}, \dots, W^{(l-1)} a_t^{(l-2)}$ αντίστοιχα (για $t = 1, \dots, k$) και να παράγονται τα διανύσματα $a_t^{(2)}, \dots, a_t^{(l-1)}$, αντίστοιχα, που είναι επίσης διαστάσεων 64×1 το καθένα.

Το $b^{(l)}$ θα είναι ένας πραγματικός αριθμός (εκφυλισμένο διάνυσμα 1×1), ώστε να προστίθεται στον πραγματικό αριθμό που παράγει ο πολλαπλασιασμός $W^{(l)} a_t^{(l-1)}$ (για $t = 1, \dots, k$) και να παράγονται οι πραγματικοί αριθμοί $a_1^{(l)}, \dots, a_\kappa^{(l)}$ της διαφάνειας 21.

(β) Θέλουμε να χρησιμοποιήσουμε το νευρωνικό δίκτυο των διαφανειών 19–21, τώρα για να κατατάξουμε tweets (που αναφέρονται σε ένα προϊόν) στις **κατηγορίες c_1 (θετική γνώμη), c_2 (αρνητική γνώμη), c_3 (ουδέτερη γνώμη), c_4 (θετική και αρνητική γνώμη μαζί)**. Κάθε tweet θα κατατάσσεται σε ακριβώς μία κατηγορία. **Αντικαθιστούμε τον τύπο** $P_{a-RNN}(\text{reject}|c) = \sigma(W_p h_{sum} + b_p)$ με τον **παρακάτω τύπο** που θα πρέπει να **παράγει** (στο αριστερό του μέρος) ένα **διάνυσμα** $p \in \mathbb{R}^4$, το οποίο θα περιέχει τις **πιθανότητες** (κατά το νευρωνικό δίκτυο) **το εισερχόμενο tweet να ανήκει σε κάθε μία από τις τέσσερις κατηγορίες**. **Συμπληρώστε το δεξί μέρος του τύπου, αναφέροντας τις διαστάσεις κάθε πίνακα και διανύσματος** που θα εμφανίζεται στο δεξί μέρος του τύπου. **Αιτιολογήστε σύντομα την απάντησή σας**.

Νέος τύπος:

$$p = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \in \mathbb{R}^4$$

Απάντηση:

Ο νέος τύπος θα είναι ο ακόλουθος.

$$p = \text{softmax}\left(W_p h_{sum} + b_p\right)$$

Ο πίνακας W_p θα έχει διαστάσεις 4×128, ώστε να μετατρέπει το διάνυσμα h_{sum} (το οποίο έχει διαστάσεις 128×1, αφού είναι άθροισμα των καταστάσεων του RNN) σε διάνυσμα τεσσάρων πραγματικών αριθμών, τους οποίους κατόπιν η softmax μετατρέπει σε κατανομή πιθανότητας (τέσσερις αριθμούς, τον καθένα μεταξύ 0 και 1, με άθροισμα 1).

Το b_p θα είναι διάνυσμα 4×1, ώστε να προστίθεται στο διάνυσμα 4×1 που παράγει ο πολλαπλασιασμός $W_p h_{sum}$.

22.3. Write down the equations for a modified version of the "RNN with deep self-attention" (slides 19–21), where the uni-directional RNN with GRU cells is replaced by a stacked bi-directional RNN with GRU cells. Use the notation $\text{GRU}(h_{t-1}, \tau_t)$ to denote the new state of a GRU cell with previous state h_{t-1} and input τ_t.

Answer: At the first layer of the GRU RNN, we have (for $t = 1, \dots, k$):

$$\vec{h}_t^{(1)} = \text{GRU}\left(\vec{h}_{t-1}^{(1)}, x_t\right)$$

$$\overleftarrow{h}_t^{(1)} = \text{GRU}\left(\overleftarrow{h}_{t+1}^{(1)}, x_t\right)$$

$$h_t^{(1)} = [\vec{h}_t^{(1)}; \overleftarrow{h}_t^{(1)}]$$

where $\vec{h}_0^{(1)}$ is the initial state of the left-to-right GRU RNN of the first layer, $\overleftarrow{h}_{k+1}^{(1)}$ is the initial state of the right-to-left GRU RNN of the first layer, `;' denotes concatenation, and $x_1, \dots, x_k$ are the word embeddings of the input word sequence.

Similarly, at the m-th layer of the GRU RNN:

$$\vec{h}_t^{(m)} = \text{GRU}\left(\vec{h}_{t-1}^{(m)}, h_t^{(m-1)}\right)$$

$$\overleftarrow{h}_t^{(m)} = \text{GRU}\left(\overleftarrow{h}_{t+1}^{(m)}, h_t^{(m-1)}\right)$$

$$h_t^{(m)} = [\vec{h}_t^{(m)}; \overleftarrow{h}_t^{(m)}]$$

The other equations remain as on slide 21.

22.4. Modify the equations of the neural network of the previous exercise to support *multi-label classification*, i.e., cases where the same text (e.g., tweet) may belong in multiple classes (labels). As a twist, use a separate *label-specific self-attention-head* for each class, which will produce a different distribution of attention scores $a_{c,1}, \dots, a_{c,k}$ (where k is again the length of the input text, counted in words) and a different $h_{sum,c}$ for each class c. Feed the $h_{sum,c}$ of each class c to a separate (different per class) dense layer with a sigmoid to produce the probability that the input text should be assigned class c.

Answer: Let C be the set of possible classes (labels). We modify the self-attention MLP of slides 19–21, so that $a_t^{(l)} \in \mathbb{R}^{|C|}$, i.e., $a_t^{(l)}$ is now a vector (not a scalar) containing $|C|$ attention scores $a_{1,t}, \dots, a_{|C|,t}$ for word position t, one for each possible class. To achieve this, we modify the dimensions of $W^{(l)}$ and $b^{(l)}$ of layer l of the self-attention MLP, to be $|C| \times d$ and $|C|$, respectively, where d is the dimensionality of the previous layer $a_t^{(l-1)}$.

The softmax of slides 19–21 is now applied *label-wise*, on the attention scores of a particular class, i.e., for each possible class c:

$$a_{c,t} = \text{softmax}\left(a_{c,t}^{(l)}; a_{c,1}^{(l)}, \dots, a_{c,k}^{(l)}\right) = \frac{\exp(a_{c,t}^{(l)})}{\sum_{t'=1}^{k} \exp(a_{c,t'}^{(l)})}$$

We form a separate weighted sum $h_{sum,c}$ for each possible class c:

$$h_{sum,c} = \sum_{t=1}^{k} a_{c,t} h_t^{(M)}$$

where M is the number of stacked GRU RNNs of the previous exercise, and we feed each $h_{sum,c}$ to a separate dense layer (a transposed vector really, why?) $W_{p,c}$ with bias term $b_{p,c}$ per class c, to compute the probability of the corresponding class:

$$P(c|x_1, \dots, x_k) = \sigma(W_{p,c} h_{sum,c} + b_{p,c})$$

The other equations of the neural network remain as in the previous exercise.

(b) Couldn't we use a single (shared) self-attention-head (and a single h_{sum}) for all the classes? What would change in that case in the equations above? What is the advantage of using a separate *label-specific* self-attention-head for each class?