



# Ασκήσεις μελέτης B6

## Lab 6

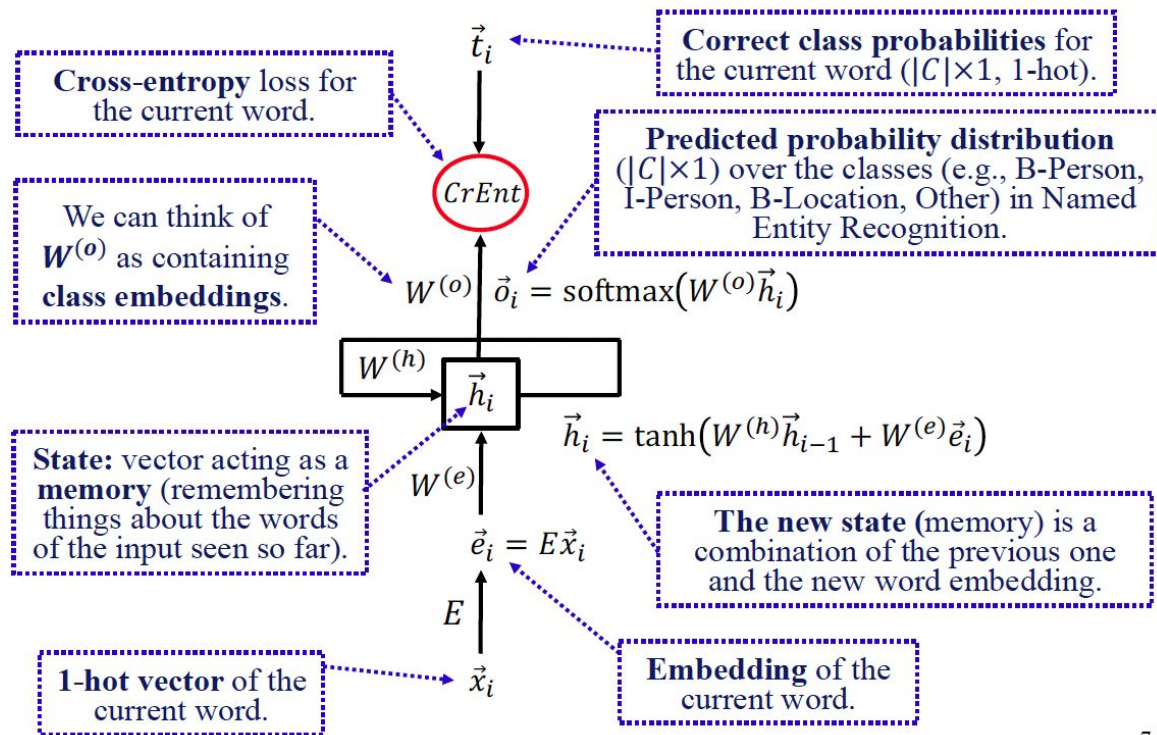
Human-Computer Interaction, AUEB  
Εαρινό εξάμηνο 2022-2023

Lab Assistant: Sofia Eleftheriou



## Άσκηση B6.1.

Θέλουμε να χρησιμοποιήσουμε το ανατροφοδοτούμενο νευρωνικό δίκτυο (RNN) της διαφάνειας 7, για να αναγνωρίζουμε ονόματα προσώπων, οργανισμών και τοποθεσιών. Χρησιμοποιούμε ετικέτες (κατηγορίες) B-I-O, όπως στην άσκηση 4 της προηγούμενης ενότητας, άρα 7 κατηγορίες. Το μέγεθος του λεξιλογίου είναι  $|V| = 100.000$ . Κάθε ενσωμάτωση λέξης (word embedding) είναι ένα διάνυσμα 300 διαστάσεων. Το κρυφό επίπεδο (η κατάσταση του RNN) αποτελείται από 500 νευρώνες, δηλαδή το  $h_i$  είναι διάνυσμα  $500 \times 1$ . 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 ενσωματώσεις των 100.000 λέξεων του λεξιλογίου. Κάθε ενσωμάτωση είναι διάνυσμα (στήλη) 300 διαστάσεων. Άρα ο  $E$  έχει διαστάσεις  $300 \times 100.000$ .

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

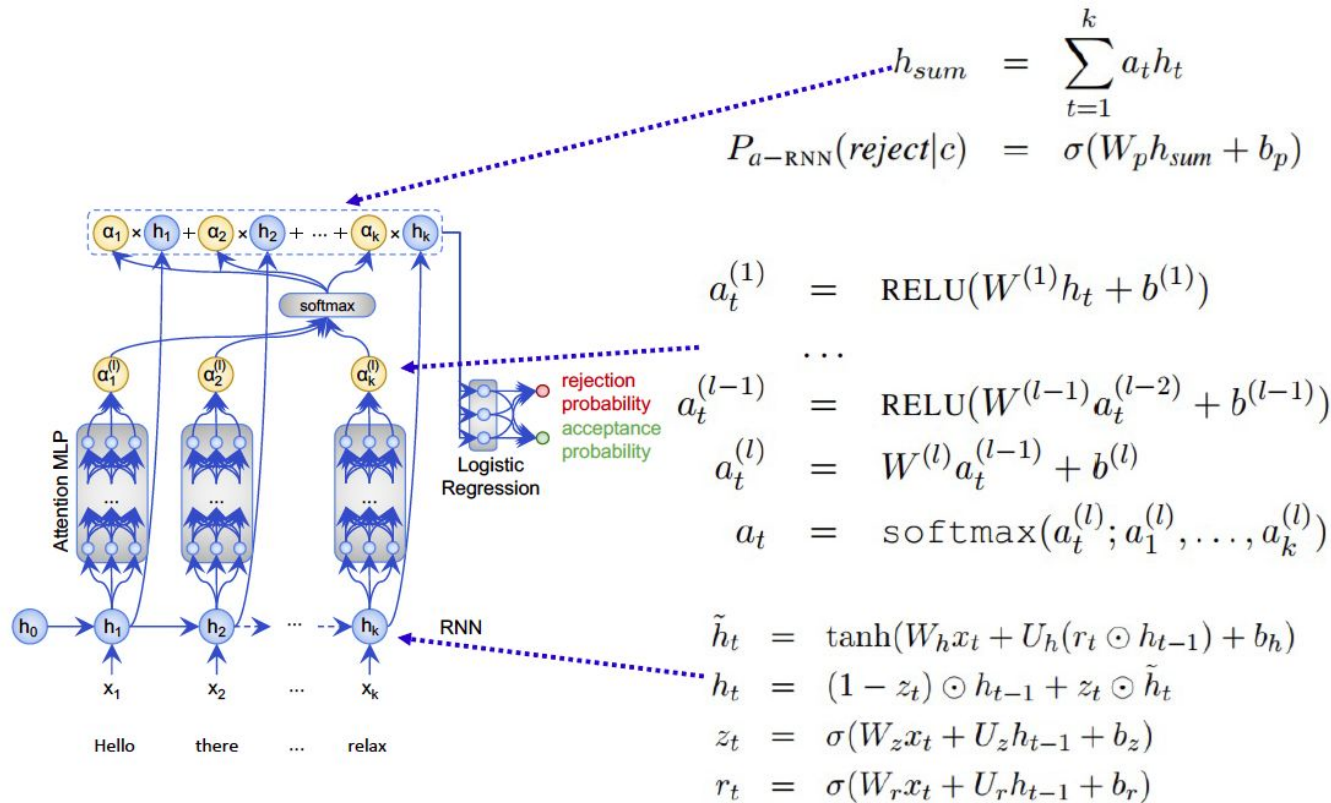
Ο πίνακας  $W^{(h)}$  έχει διαστάσεις  $500 \times 500$ , ενώ ο πίνακας  $W^{(e)}$  έχει διαστάσεις  $500 \times 300$ , ώστε τα  $W^{(h)}\vec{h}_{i-1}$  και  $W^{(e)}\vec{e}_i$  να έχουν τις ίδιες διαστάσεις ( $500 \times 1$ ), να μπορούν να προστεθούν ( $W^{(h)}\vec{h}_{i-1} + W^{(e)}\vec{e}_i$ ) και η νέα κατάσταση  $\vec{h}_i = \tanh(W^{(h)}\vec{h}_{i-1} + W^{(e)}\vec{e}_i)$ : να έχει πάλι διαστάσεις  $500 \times 1$ , 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  $7 \times 500$ , ώστε ο πολλαπλασιασμός  $W^{(o)}\vec{h}_i$  να παράγει διάνυσμα  $7 \times 1$  με έναν πραγματικό αριθμό για κάθε κατηγορία. Η softmax στον υπολογισμό  $\vec{o}_i = \text{softmax}(W^{(o)}\vec{h}_i)$  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  $7 \times 1$ .



## Άσκηση B6.2.

Write down the equations for a modified version of the “RNN with deep self-attention” (slides 16–17), where the uni-directional RNN with GRU cells is replaced by a stacked bidirectional RNN with GRU cells. Use the notation  $GRU(h_{t-1}, \tau_t)$  to denote the new state of a GRU cell with previous state  $h_{t-1}$  and input  $\tau_t$ .



$$h_{sum} = \sum_{t=1}^k a_t h_t$$

$$P_{a-RNN}(reject|c) = \sigma(W_p h_{sum} + b_p)$$

$$a_t^{(1)} = \text{RELU}(W^{(1)} h_t + b^{(1)})$$

$$\dots$$

$$a_t^{(l-1)} = \text{RELU}(W^{(l-1)} a_t^{(l-2)} + b^{(l-1)})$$

$$a_t^{(l)} = W^{(l)} a_t^{(l-1)} + b^{(l)}$$

$$a_t = \text{softmax}(a_t^{(l)}; a_1^{(l)}, \dots, a_k^{(l)})$$

$$\tilde{h}_t = \tanh(W_h x_t + U_h (r_t \odot h_{t-1}) + b_h)$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$$

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z)$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r)$$



## Απάντηση:

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

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

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

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

where  $h_0^{(1)}$  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:

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

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

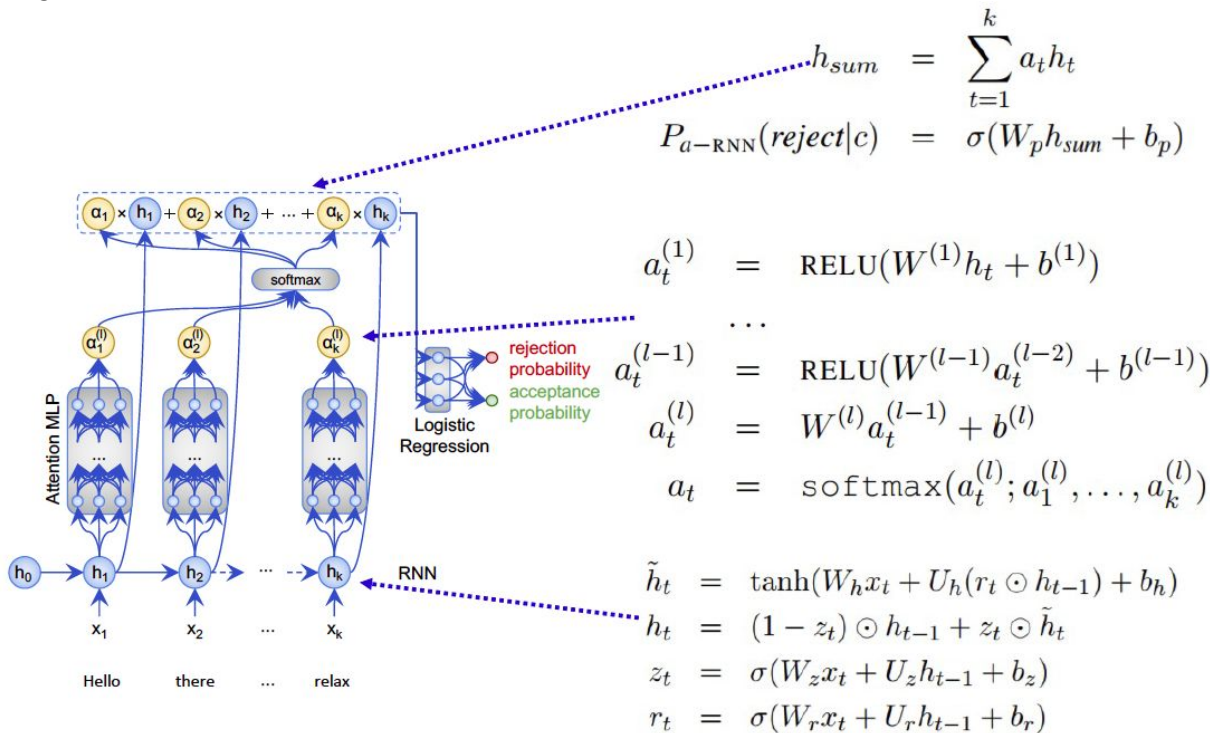
$$h_t^{(m)} = [\vec{h}_t^{(m)}; \overleftarrow{h}_t^{(m)}]$$

The other equations remain as on slide 17.



### Άσκηση B6.3

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). 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  $c$ .



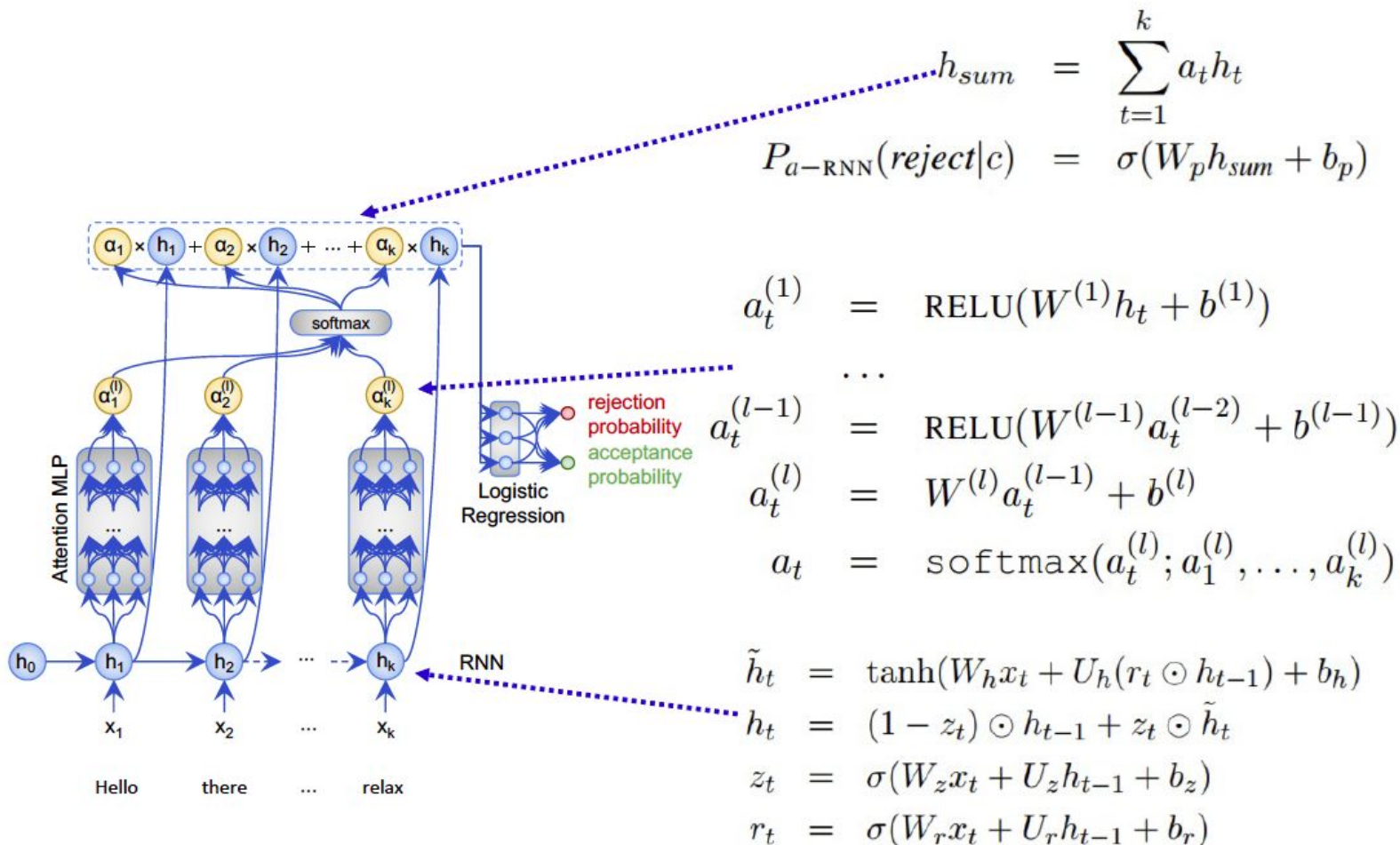


## Απάντηση:

Let  $C$  be the set of possible classes (labels). We modify the self-attention MLP of slide 17, 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 slide 17 is now applied label-wise, on the attention scores of a particular class, i.e., for each possible class  $c$ :

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







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  $W_{p,c}$  (with bias term  $b_{p,c}$ ) per class  $c$ , to compute the probability of the corresponding class:

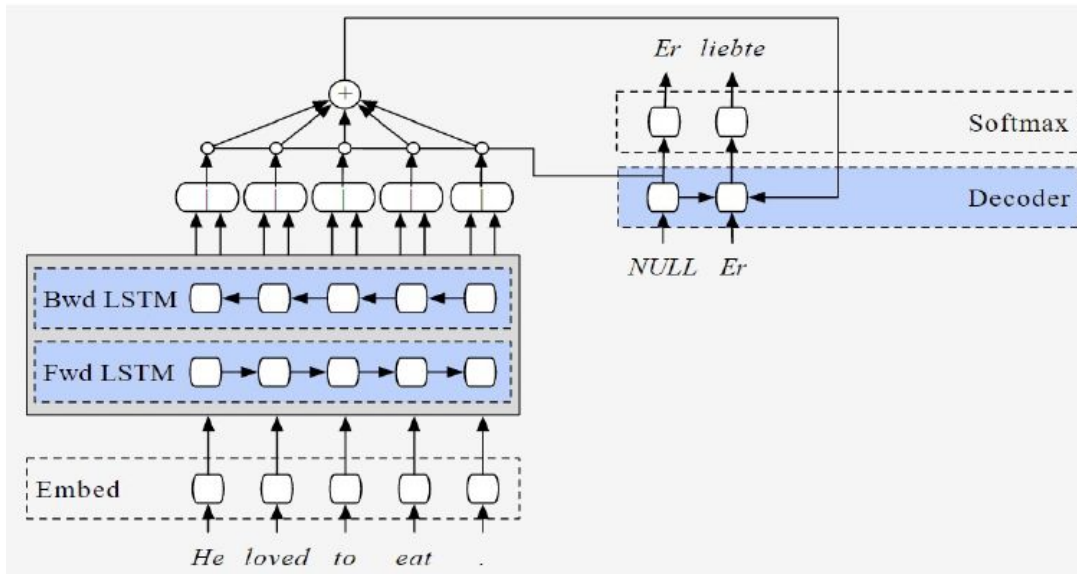
$$P(c|x_1, \dots, x_k) = \sigma(W_{p,c} h_{sum,c} + b_{p,c})$$

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



## Άσκηση B6.4

Εκπαιδεύουμε το παρακάτω νευρωνικό μοντέλο μηχανικής μετάφρασης.



Έστω  $V$  και  $V'$  τα λεξιλόγια της γλώσσας-πηγής (Αγγλικά) και της γλώσσας-στόχου (Γερμανικά) αντίστοιχα. Κάθε παράδειγμα εκπαίδευσης είναι ένα ζεύγος αποτελούμενο από μια ακολουθία one-hot διανυσμάτων:  $x_1, x_2, x_3, \dots, x_n \in \{0, 1\}^{|V|}$  που αντιστοιχούν σε μια αγγλική πρόταση (κάθε διάνυσμα δείχνει σε ποια θέση του αγγλικού λεξικού  $V$  βρίσκεται η αντίστοιχη λέξη) και μια ακολουθία one-hot διανυσμάτων:  $y_1, y_2, y_3, \dots, y_n \in \{0, 1\}^{|V'|}$  που αντιστοιχούν σε μια γερμανική πρόταση που είναι η σωστή (gold) μετάφραση της αγγλικής (κάθε διάνυσμα δείχνει σε ποια θέση του γερμανικού λεξικού  $V'$  βρίσκεται η αντίστοιχη λέξη). Έστω  $E \in \mathbb{R}^{d^{(e)} \times |V|}$  και  $E' \in \mathbb{R}^{d^{(e)} \times |V'|}$  οι πίνακες με τα word embeddings (το καθένα  $d^{(e)}$  διαστάσεων) των δύο γλωσσών αντίστοιχα. Οι παρακάτω τύποι περιγράφουν αναλυτικά τη λειτουργία του μοντέλου και τον υπολογισμό του σφάλματος ( $L$ ) για ένα παράδειγμα εκπαίδευσης. Συμπληρώστε τα κενά (στη λύση έχουν συμπληρωθεί με κόκκινο). Ο συμβολισμός  $[\dots; \dots]$  παριστάνει συνένωση (concatenation). Τα  $f$  και  $g$  παριστάνουν συναρτήσεις ενεργοποίησης.



**Κωδικοποιητής:** ( $i \in \{1, 2, 3, \dots, n\}$ )

$$e_i = E x_i \in \mathbb{R}^{d^{(e)}}$$

$$\vec{h}_i = \text{LSTM}(\vec{h}_{i-1}, e_i) \in \mathbb{R}^{d^{(h)}}$$

$$\vec{h}_0 \in \mathbb{R}^{d^{(h)}}$$

$$\tilde{h}_i = \text{LSTM}(\tilde{h}_{i+1}, e_i) \in \mathbb{R}^{d^{(h)}}$$

$$\tilde{h}_{n+1} \in \mathbb{R}^{d^{(h)}}$$

$$h_i = [\vec{h}_i; \tilde{h}_i] \in \mathbb{R}^{2 \cdot d^{(h)}}$$

**Αποκωδικοποιητής:** ( $i \in \{1, 2, 3, \dots, n\}, j \in \{1, 2, 3, \dots, m\}$ )

$$t_j = E' y_j \in \mathbb{R}^{d^{(e)}} \quad (\text{To embedding της σωστής γερμανικής λέξης στη θέση } j.)$$

$$z_j = \text{LSTM}(z_{j-1}, [t_{j-1}; c_j]) \in \mathbb{R}^{d^{(z)}}$$

$$z_0 \in \mathbb{R}^{d^{(z)}}, t_0 \in \mathbb{R}^{d^{(e)}}$$

$$\tilde{a}_{i,j} = v^T \cdot f(W^{(a)} h_i + U^{(a)} z_{j-1} + b^{(a)}) \in \mathbb{R}$$

$$W^{(a)} \in \mathbb{R}^{d^{(z)} \times 2 \cdot d^{(h)}}$$

$$U^{(a)} \in \mathbb{R}^{d^{(z)} \times d^{(z)}}$$

$$b^{(a)} \in \mathbb{R}^{d^{(z)}}, v \in \mathbb{R}^{d^{(z)}}$$

$$a_{i,j} = \frac{\exp(\tilde{a}_{i,j})}{\sum_{i'} \exp(\tilde{a}_{i',j})}$$



**Αποκωδικοποιητής:** ( $i \in \{1, 2, 3, \dots, n\}, j \in \{1, 2, 3, \dots, m\}$ )

$$c_j = W^{(c)} \cdot g(\sum_i a_{i,j} h_i + b^{(c)}) \in \mathbb{R}^{d^{(e)}}$$

$$W^{(c)} \in \mathbb{R}^{d^{(e)} \times 2 \cdot d^{(h)}}$$

$$b^{(c)} \in \mathbb{R}^{2 \cdot d^{(h)}}$$

$$\tilde{o}_j = W^{(o)} z_j + b^{(o)} \in \mathbb{R}^{|V'|}$$

$$W^{(o)} \in \mathbb{R}^{|V'| \times d^{(z)}}$$

$$b^{(o)} \in \mathbb{R}^{|V'|}$$

$$o_{j,k} = \frac{\exp(\tilde{o}_{j,k})}{\sum_{k=1}^{|V'|} \exp(\tilde{o}_{j,k})}$$

(Πόσο πιθανό θεωρεί το μοντέλο η  $k$ -στή λέξη του γερμανικού λεξιλογίου να είναι η σωστή για την  $j$ -στή θέση της μετάφρασης.)

$$r_j = \operatorname{argmax}_l y_{j,l}$$

(Σύμφωνα με το 1-hot  $y_j$ , η σωστή λέξη στην  $j$ -στή θέση της μετάφρασης βρίσκεται στη θέση  $r_j$  του γερμανικού λεξιλογίου.)

$$L = -\sum_j \log o_{j,r_j}$$

(Ελαχιστοποιώντας το  $L$ , μεγιστοποιούμε την πιθανότητα που δίνει το μοντέλο στις σωστές λέξεις, σε όλες τις θέσεις της μετάφρασης.)