



Εισαγωγή στην Αναγνώριση Ομιλίας με Βαθιά Μάθηση

2025–26

Ίων Ανδρουτσόπουλος

<http://www.aueb.gr/users/ion/>

Οι διαφάνειες αυτές βασίζονται κυρίως στην ύλη του βιβλίου *Speech and Language Processing* των D. Jurafsky και J.H. Martin, 2^η έκδοση, Pearson Education, 2009 και 3^η έκδοση (υπό προετοιμασία).

Τι θα ακούσετε

- **Εισαγωγή** στην αναγνώριση ομιλίας.
- **Δημιουργία παραστάσεων τμημάτων** ομιλίας με **προ-εκπαιδευμένους Transformers**.
 - wav2vec, HuBERT.
- **Μοντέλα αναγνώρισης ομιλίας**.
 - Μοντέλα κωδικοποιητή/αποκωδικοποιητή.
 - Μοντέλα κωδικοποιητή μόνο.
 - Χρήση γλωσσικών μοντέλων.
- **Μέτρα αξιολόγησης** αναγνώρισης ομιλίας.
- **Παλαιότερη τεχνολογία** (προαιρετική ύλη).
 - Διανύσματα **MFCC**.
 - Hidden Markov Models (**HMMs**).

Γιατί αναγνώριση ομιλίας;

- Χρήστες με **δυσκολίες ακοής, κίνησης, όρασης**.
 - Π.χ. αυτόματη **παραγωγή υποτίτλων**.
 - **Χειρισμός συσκευών μέσω προφορικών εντολών**.
- Όταν τα **χέρια ή τα μάτια** είναι **απασχολημένα**.
 - Π.χ. **περπάτημα, οδήγηση**.
- Ιδιαίτερα σε συστήματα **προφορικών διαλόγων**.
 - Π.χ. **κλείσιμο εισιτηρίων μέσω τηλεφώνου**.
- **Εξαγωγή πληροφοριών ή γνώμης**.
 - Π.χ. από τηλεφωνικές **συνδιαλέξεις ή εκπομπές**.
- **Φυσικότερη ή εντυπωσιακότερη επικοινωνία**.
 - Π.χ. **υπαγόρευση μηνυμάτων ή κειμένου**.
 - Π.χ. αλληλεπίδραση με **ρομπότ ή παιχνίδια**.
 - Προσοχή στις **λανθασμένες προσδοκίες!**



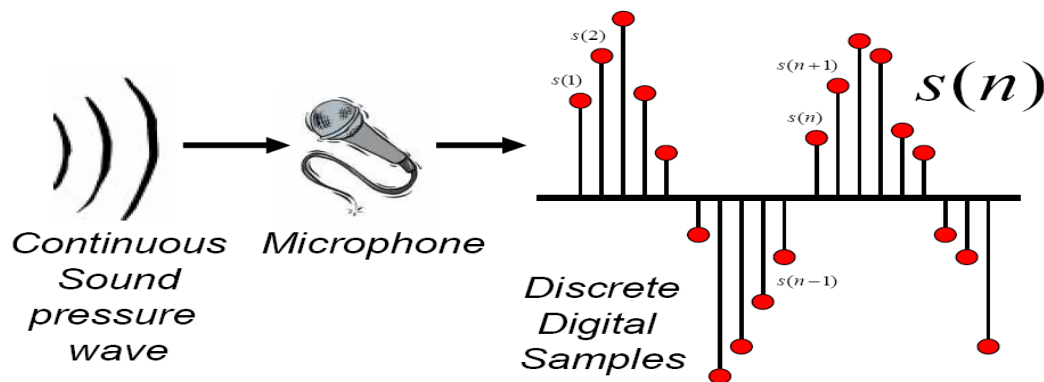
Τι επηρεάζει την αναγνώριση;

- **Μέγεθος λεξιλογίου.**
 - **Εύκολο:** αναγνώριση αριθμών ή δεκάδων λέξεων.
 - **Πιο δύσκολο:** αναγνώριση δεκάδων χιλιάδων λέξεων (π.χ. στην υπαγόρευση κειμένου).
- **Μεμονωμένες λέξεις ή συνεχής ομιλία.**
 - Σε συνομιλίες μεταξύ ανθρώπων συνήθως δεν υπάρχουν κενά μεταξύ των λέξεων.
 - Η αναγνώριση μεμονωμένων λέξεων είναι πιο εύκολη.
- **Για συγκεκριμένο χρήστη ή όχι;**
 - Π.χ. τα συστήματα υπαγόρευσης συχνά βελτιώνονται με δείγματα ομιλίας του συγκεκριμένου χρήστη.
 - Τα περισσότερα συστήματα πλέον δεν απαιτούν ειδική εκπαίδευση ανά χρήστη.

Τι επηρεάζει την αναγνώριση;

- **Μητρική γλώσσα ή όχι; Διάλεκτοι.... Ηλικία...**
 - Συνήθως υποστηρίζονται καλύτερα **συγκεκριμένες γλώσσες και διάλεκτοι**, κυρίως για **ενήλικες**.
- **Μικρόφωνα, πλήθος χρηστών, θόρυβος.**
 - **Ευκολότερο: ένας χρήστης με ακουστικό κεφαλής σε ήσυχο γραφείο.**
 - **Πολύ δυσκολότερο: πολλοί χρήστες σε θορυβώδες περιβάλλον (π.χ. συνεδρίαση) με μακρινά μικρόφωνα.**
- **Είδος συνομιλίας.**
 - Η αυτόματη αναγνώριση ομιλίας **μεταξύ ανθρώπων (π.χ. πρακτικά συνεδριάσεων)** είναι πολύ πιο δύσκολη.
 - Οι άνθρωποι **απλοποιούν** την ομιλία τους όταν μιλούν σε **μηχανές (ή σε παιδιά ή σε μαθητές ξένων γλωσσών)**.

Ψηφιακή παράσταση σήματος

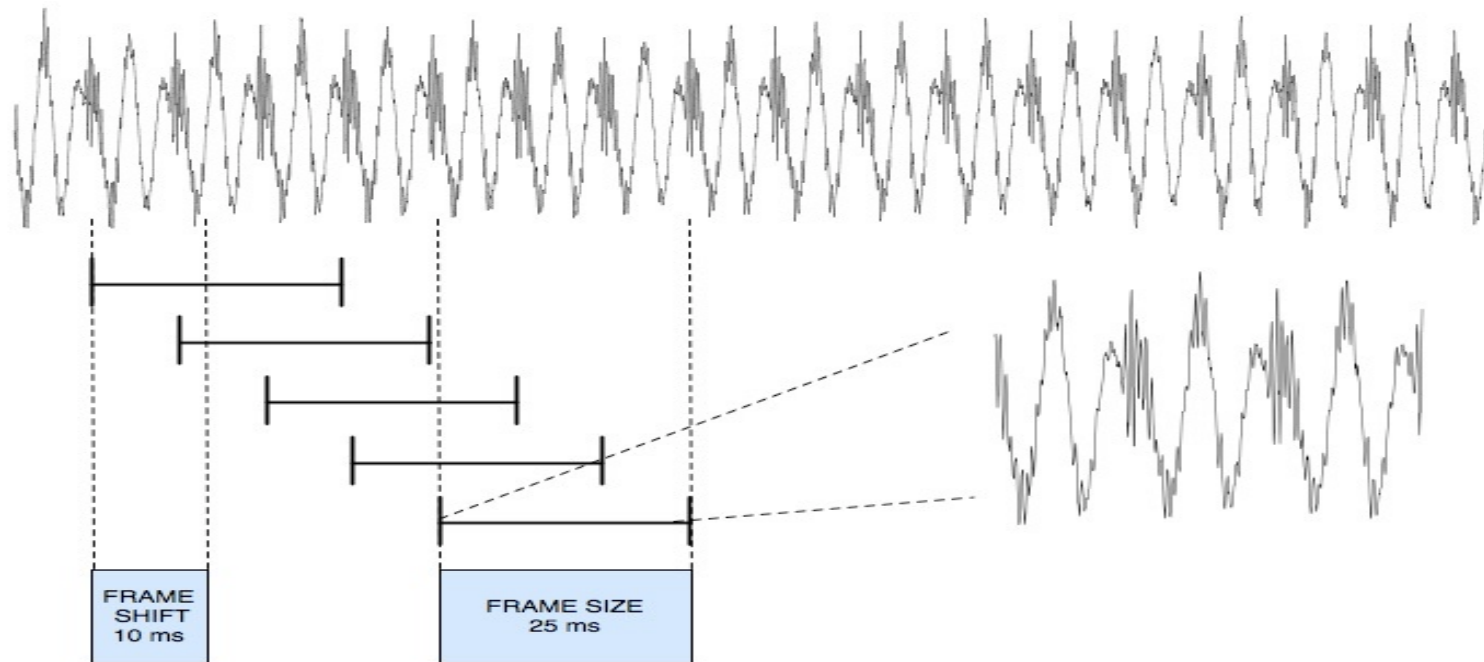


Σχήμα από τις διαφάνειες των Jurafsky & Martin (2008), προερχόμενο από τον B. Pellom.

- **Μέτρηση του αναλογικού σήματος (πίεση αέρα) ανά τακτά χρονικά διαστήματα ($10\text{Hz} = 10$ φορές ανά sec).**
 - Απαιτείται **συχνότητα δειγματοληψίας τουλάχιστον διπλάσια** από τη **μέγιστη συχνότητα (συνιστώσα) του σήματος**.
 - **Ομιλία:** $< \sim 10\text{ KHz}$, άρα δειγματοληψία $\geq 20\text{ KHz}$.
 - **Τηλεφωνία:** $< 4\text{ KHz}$, άρα δειγματοληψία $\geq 8\text{ KHz}$.
- **Οι μετρήσεις αποθηκεύονται ως ακέραιοι.**
 - Συνήθως των 8 bit (-128 ως 127) ή 16 bit (-32.768 ως 32.767).

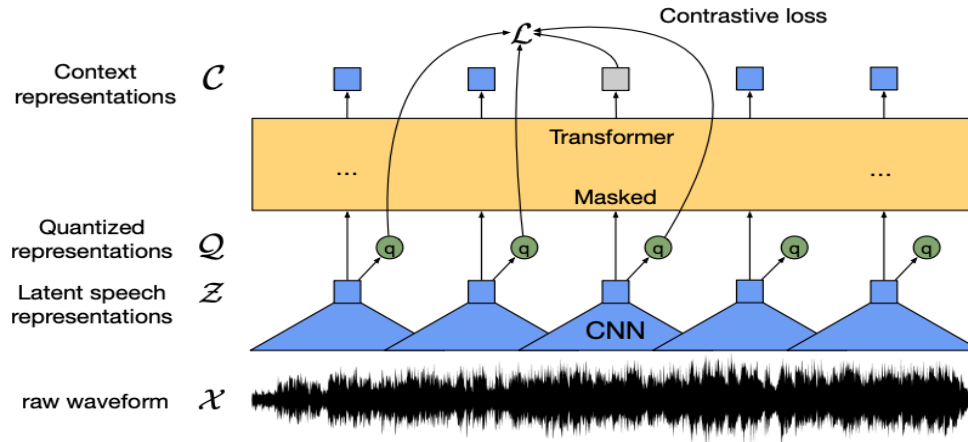
Τμήματα (frames)

Σχήμα από τις διαφάνειες των
Jurafsky & Martin (2008).



- Εξάγουμε **επικαλυπτόμενα τμήματα (frames)** του σήματος.
 - **Σέρνουμε ένα «παράθυρο»** κατά μήκος του σήματος.
- **Κάθε τμήμα** συχνά παριστάνεται από ένα **διάνυσμα**.
 - Παραδοσιακά **39 MFCC features** (βασισμένα σε μετασχηματισμό Fourier).
 - Πιο πρόσφατα **διανύσματα** που παράγονται με **προ-εκπαιδευμένους Transformers** (π.χ. **wav2vec, HuBERT**).

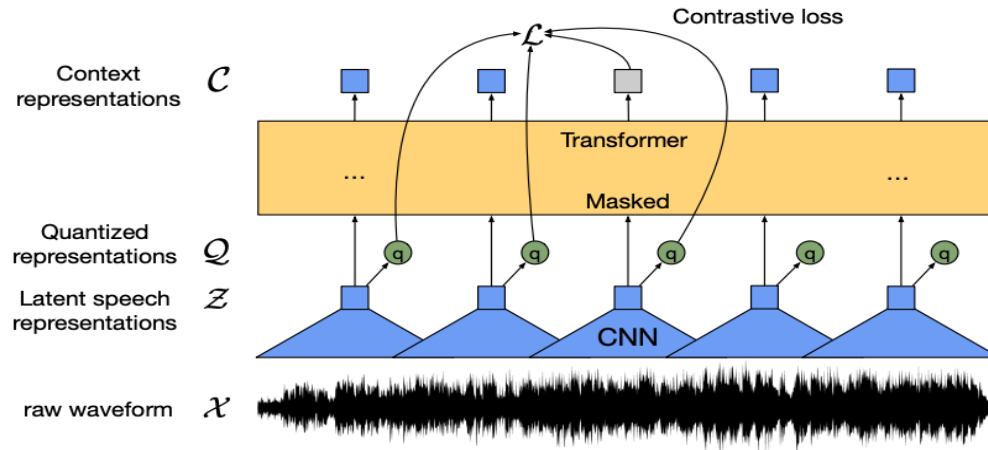
wav2vec



Σχήμα από το άρθρο των Baevsky κ.ά., «wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations», NeurIPS 2020 (<https://arxiv.org/abs/2006.11477>).

- Ένα **CNN** παράγει ένα **διάνυσμα** (z) για **κάθε τμήμα** (frame).
 - Η **είσοδος στο CNN** είναι οι (διακριτές) **τιμές ήχου ενός τμήματος** (διάνυσμα x). Σκεφτείτε τις σαν μια **μονοδιάστατη μικρή εικόνα** με ένα **κανάλι εισόδου**.
 - Με n **συνελικτικά φίλτρα** $\rightarrow$ **διάνυσμα n χαρακτηριστικών** (z) για κάθε τμήμα x .
- Για **κάθε διάνυσμα τμήματος** (z) που προκύπτει παίρνουμε και το **κοντινότερο διάνυσμα** (q) από ένα **codebook**.
 - Το **codebook** περιέχει **σταθερό αριθμό διανυσμάτων**, τα οποία **μαθαίνουμε**.
 - Ακριβέστερα χρησιμοποιούνται **πολλαπλά codebooks**. **Συνενώνουμε** τα διανύσματα που προκύπτουν (για το τμήμα) από κάθε codebook και τα περνάμε από ένα **dense layer** για να πάρουμε το q .

wav2vec – συνέχεια

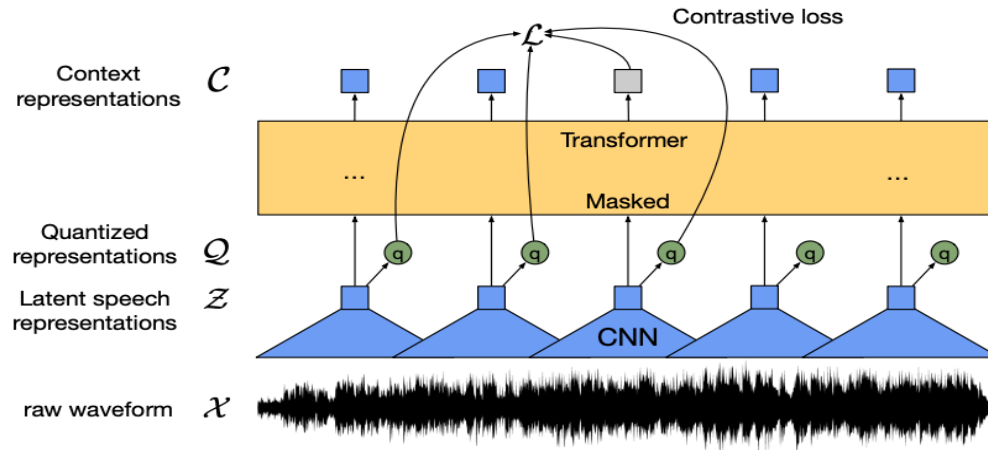


Σχήμα από το άρθρο των Baevsky κ.ά., «wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations», NeurIPS 2020 (<https://arxiv.org/abs/2006.11477>).

- Τα **διανύσματα των τμημάτων** (z) που παράγει το CNN περνούν από στοιβαγμένους **κωδικοποιητές Transformer** (όπως στο BERT).
 - Έτσι παράγονται **νέα διανύσματα τμημάτων** (c) που είναι «γνωρίζουν» και τα υπόλοιπα τμήματα (**context-aware**).
- Κατά την **προ-εκπαίδευση**, **κρύβουμε τυχαία τμήματα** (διανύσματα z) και απαιτούμε να «**μαντέψει**» το wav2vec τα **διανύσματα q τους από το αντίστοιχο context-aware διάνυσμα c** .
 - **Αντικαθιστούμε** στην είσοδο των Transformers τα **διανύσματα (z) για τα κρυμμένα τμήματα με ένα κοινό διάνυσμα** (σαν του [MASK] στο BERT).

wav2vec – συνέχεια

Σχήμα από το άρθρο των Baevsky κ.ά., «wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations», NeurIPS 2020 (<https://arxiv.org/abs/2006.11477>).



- Κατά την προ-εκπαίδευση, κρύβουμε τυχαία τμήματα (διανύσματα z) και απαιτούμε να «μαντέψει» το wav2vec τα διανύσματα q τους από το αντίστοιχο context-aware διάνυσμα c_t .
 - Ζητάμε από το wav2vec να επιλέξει το σωστό q διάνυσμα μεταξύ των $\tilde{q}$ διανυσμάτων όλων των κρυμμένων τμημάτων (σφάλμα L_m).

$$\mathcal{L}_m = -\log \frac{\exp(\text{sim}(c_t, q_t)/\kappa)}{\sum_{\tilde{q} \sim Q_t} \exp(\text{sim}(c_t, \tilde{q})/\kappa)}$$

- Ένα πρόσθετο σφάλμα (loss, βασισμένο στην εντροπία) φροντίζει να χρησιμοποιούνται όλα τα διανύσματα του κάθε codebook.

HuBERT

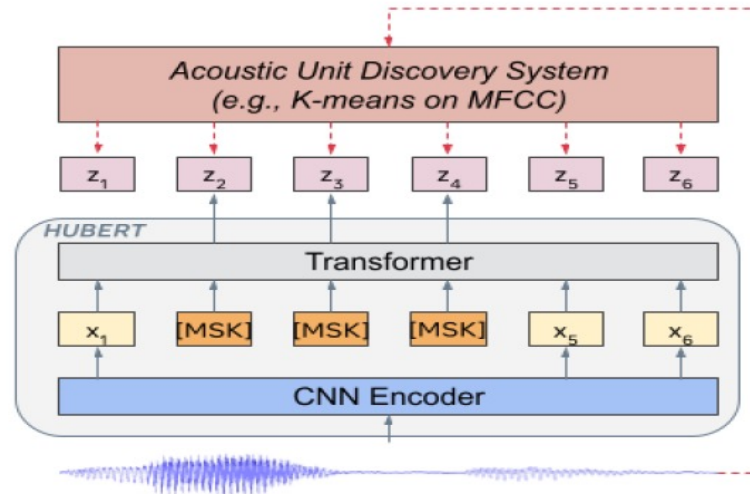


Fig. 1: The HuBERT approach predicts hidden cluster assignments of the masked frames (y_2, y_3, y_4 in the figure) generated by one or more iterations of k-means clustering.

Σχήμα και τύπος από το άρθρο των Hsu κ.ά., «HuBERT: Self-Supervised Speech Representation of Hidden Units», 2021 (<https://arxiv.org/abs/2106.07447>).

- Παρόμοιο με το wav2vec αλλά τώρα για κάθε κρυμμένο (ή μη) τμήμα (frame) απαιτούμε κατά την προ-εκπαίδευση το μοντέλο να μαντεύει τη συστάδα (cluster) στην οποία ανήκει το τμήμα.
 - Οι αρχικές συστάδες παράγονται εφαρμόζοντας τον **k-means** στα διανύσματα **MFCC** των τμημάτων όλων των δεδομένων (ήχος μόνο) προ-εκπαίδευσης.
 - Σε επόμενους κύκλους παράγουμε νέες συστάδες-στόχους εφαρμόζοντας τον **k-means** στα διανύσματα των τμημάτων προ-εκπαίδευσης που παράγει το μοντέλο του προηγούμενου κύκλου.

Κωδικοποιητές/αποκωδικοποιητές

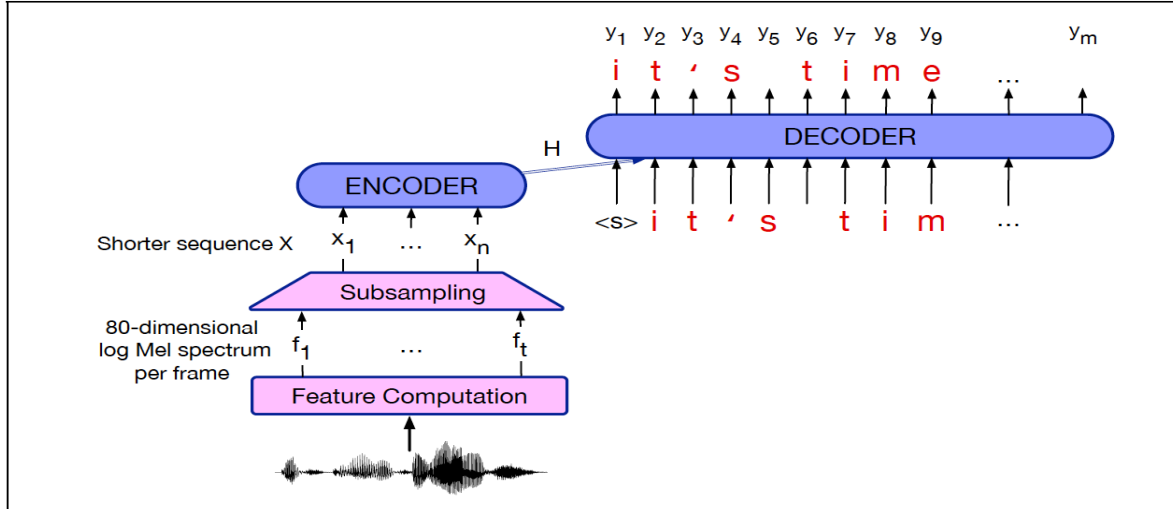


Figure 26.6 Schematic architecture for an encoder-decoder speech recognizer.

Σχήμα από το βιβλίο «Speech and Language Processing» των D. Jurafsky & J.H. Martin, 3η έκδοση (υπό προετοιμασία).
<http://web.stanford.edu/~jurafsky/slp3/>

- Ο κωδικοποιητής και ο αποκωδικοποιητής μπορούν να είναι **RNNs**, όπως είδαμε στη μηχανική μετάφραση. Συχνά είναι πια **Transformers**.
 - Εκπαιδεύονται **μαζί**, σε **ζεύγη εισόδων-εξόδων**, όπως στην μηχανική μετάφραση.
- Ο κωδικοποιητής διαβάζει μια **ακολουθία διανυσμάτων**, ένα διάνυσμα για **κάθε τμήμα ήχου (frame)** (ή λιγότερα, αν κάνουμε υπο-δειγματοληψία τους).
 - Παραδοσιακά χρησιμοποιούνταν **διανύσματα MFCC** ή παρόμοια. Πιο πρόσφατα χρησιμοποιούνται διανύσματα που παράγονται από μοντέλα σαν το **wav2vec**.
 - Συνήθως υπάρχει και ένας **μηχανισμός προσοχής**, όπως στη μηχανική μετάφραση.

Κωδικοποιητές/αποκωδικοποιητές

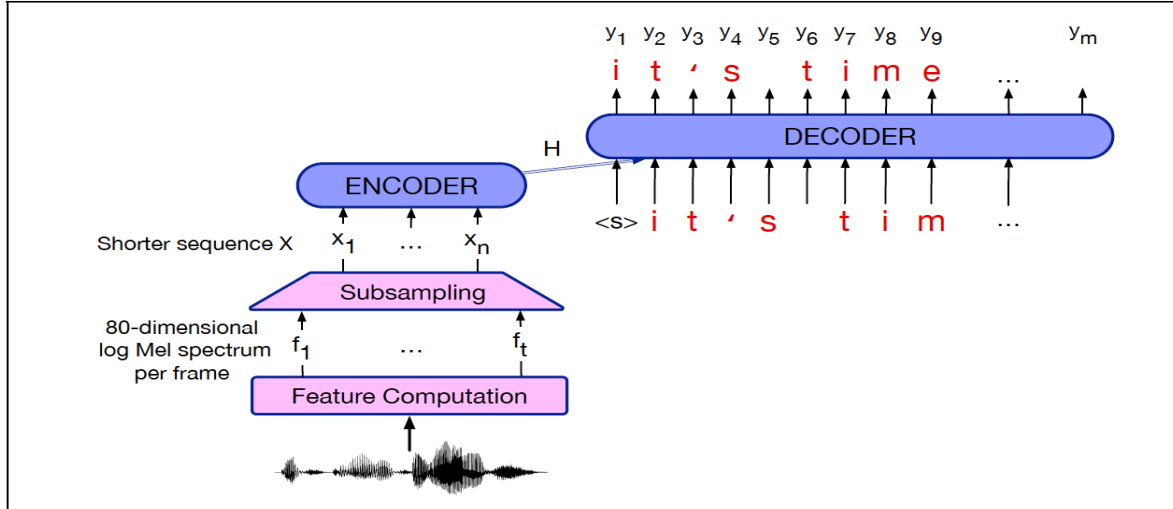


Figure 26.6 Schematic architecture for an encoder-decoder speech recognizer.

Σχήμα από το βιβλίο «Speech and Language Processing» των D. Jurafsky & J.H. Martin, 3η έκδοση (υπό προετοιμασία).
<http://web.stanford.edu/~jurafsky/slp3/>

- Ο αποκωδικοποιητής παράγει γράμμα-γράμμα το κείμενο.
 - **Κατά την εκπαίδευση**, όταν υπολογίζουμε τη νέα κατάσταση του αποκωδικοποιητή, χρησιμοποιούμε ως **προηγούμενο γράμμα το σωστό προηγούμενο (teacher forcing)**, ακόμα και αν ο αποκωδικοποιητής είχε επιλέξει άλλο (λάθος) προηγούμενο γράμμα.
 - **Σταδιακά** μπορούμε να χρησιμοποιούμε όλο και **συχνότερα το προηγούμενο γράμμα** που είχε επιλέξει ο ίδιος ο **αποκωδικοποιητής (scheduled sampling)**.
 - Μετά την εκπαίδευση, **σε κάθε κατάσταση του αποκωδικοποιητή** επιλέγουμε **λαίμαργα το γράμμα στο οποίο δίνει μεγαλύτερη πιθανότητα το μοντέλο**. **Εναλλακτικά δειγματοληψία (sampling) ή beam search**.

Κωδικοποιητές/αποκωδικοποιητές

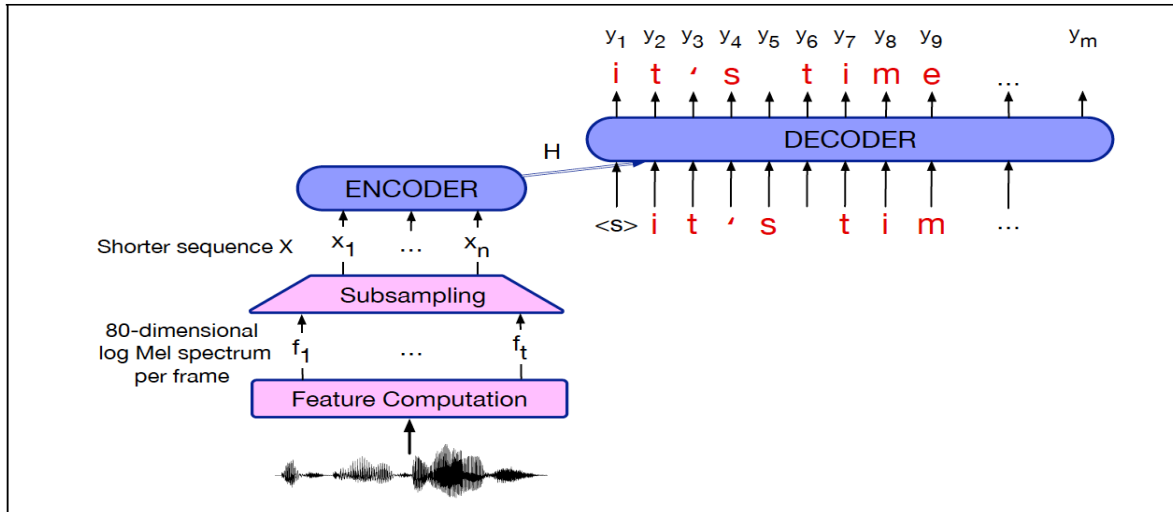


Figure 26.6 Schematic architecture for an encoder-decoder speech recognizer.

Σχήμα από το βιβλίο «Speech and Language Processing» των D. Jurafsky & J.H. Martin, 3η έκδοση (υπό προετοιμασία).
<http://web.stanford.edu/~jurafsky/slp3/>

- Μπορούμε να παράγουμε τις n πιθανότερες ακολουθίες γραμμάτων (n -best list) και μετά να τις φιλτράρουμε λαμβάνοντας υπόψιν τις πιθανότητες που τους δίνει ένα γλωσσικό μοντέλο εκπαιδευμένο σε πολύ μεγάλα σώματα κειμένων.
 - Ή λαμβάνουμε υπόψιν τις πιθανότητες και του γλωσσικού μοντέλου κατά το beam search.
 - Π.χ. προσθέτουμε την λογαριθμική πιθανότητα $p = \log(p_1) + \log(p_2) + \dots$ που δίνει ο αποκωδικοποιητής σε ένα μονοπάτι $y_1, y_2, \dots$ (που εξερευνούμε κατά το beam search) και την λογαριθμική πιθανότητα q που δίνει στο μονοπάτι $y_1, y_2, \dots$ το γλωσσικό μοντέλο ($\lambda_1 p + \lambda_2 q$).
 - Τα γλωσσικά μοντέλα «προτιμούν» σύντομες ακολουθίες (γιατί;), οπότε συνήθως χρησιμοποιούμε και έναν παράγοντα που επιβραβεύει μακρύτερες ακολουθίες γραμμάτων.

ASR με κωδικοποιητές μόνο

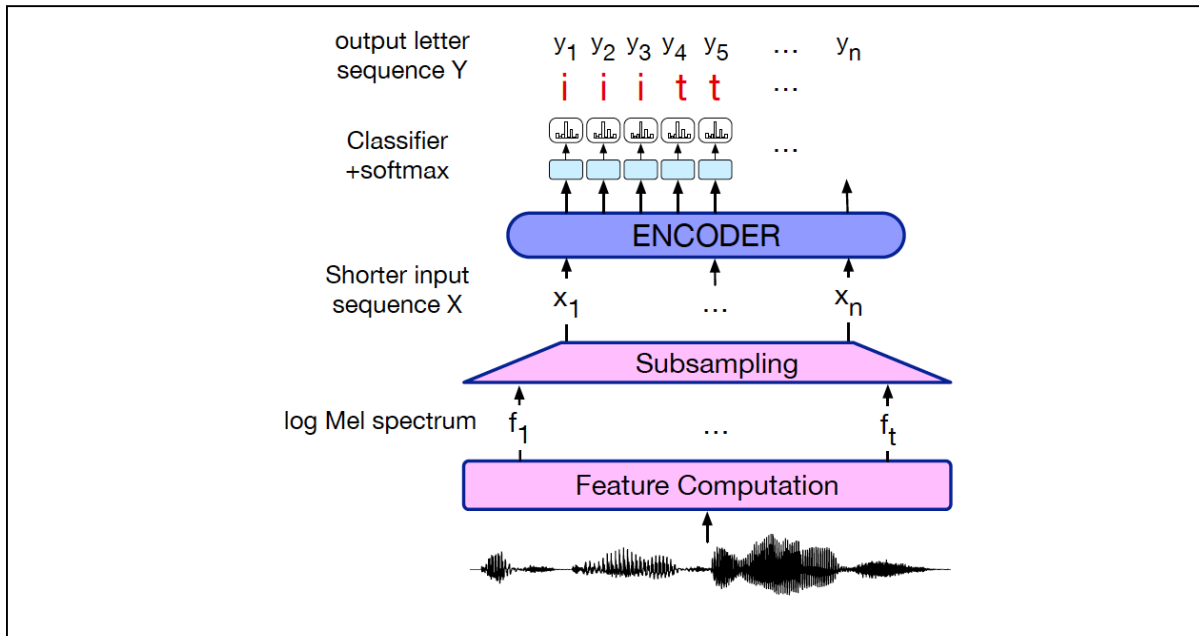


Figure 26.10 Inference with CTC: using an encoder-only model, with decoding done by simple softmaxes over the hidden state h_t at each output step.

- Από κάθε κατάσταση h_i του κωδικοποιητή, παράγουμε ένα γράμμα.
 - Π.χ. περνάμε την κατάσταση h_i από ένα **dense layer** (ή MLP) με **softmax**.
 - Δεν υπάρχει τώρα αποκωδικοποιητής.
 - Δεν μπορούμε να αλλάξουμε τη σειρά στην έξοδο αλλά δεν χρειάζεται στην αναγνώριση ομιλίας (ενώ χρειάζεται στη μηχανική μετάφραση).

Σχήματα από το βιβλίο «Speech and Language Processing» των D. Jurafsky & J.H. Martin, 3η έκδοση (υπό προετοιμασία).
<http://web.stanford.edu/~jurafsky/slp3/>

ASR με κωδικοποιητές μόνο

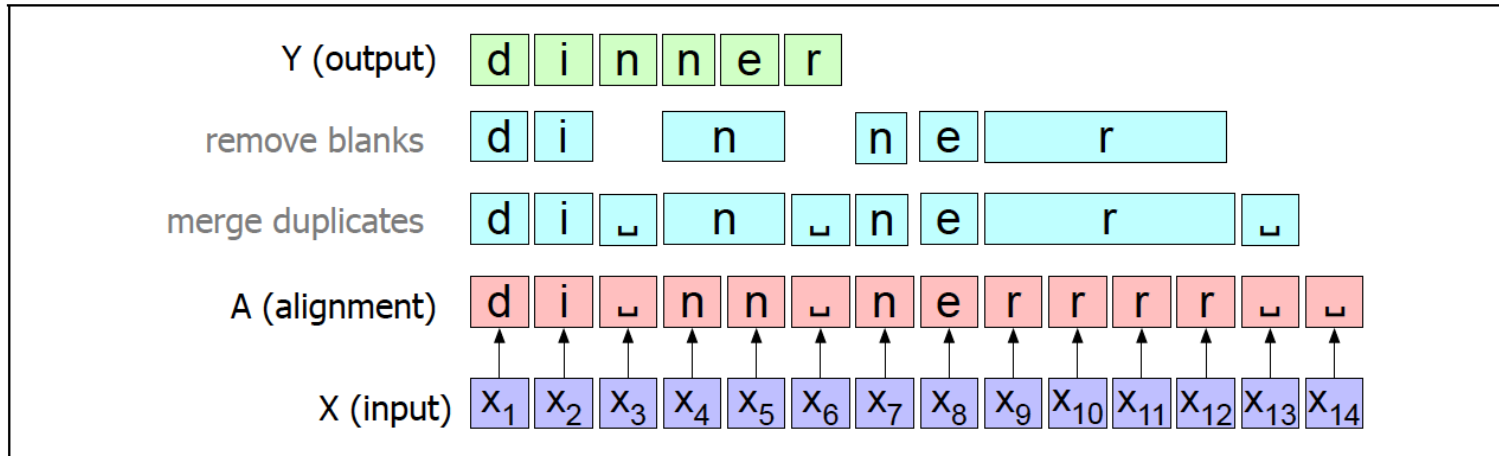


Figure 26.8 The CTC collapsing function B , showing the space blank character $_$; repeated (consecutive) characters in an alignment A are removed to form the output Y .

Σχήμα από το βιβλίο «Speech and Language Processing» των D. Jurafsky & J.H. Martin, 3η έκδοση (υπό προετοιμασία).
<http://web.stanford.edu/~jurafsky/slp3/>

- Από κάθε κατάσταση h_i του κωδικοποιητή, παράγουμε ένα γράμμα.
 - Παράγονται τόσα γράμματα όσα και τα διανύσματα στην είσοδο του κωδικοποιητή.
 - Τελικά διαγράφουμε συνεχόμενες εμφανίσεις του ίδιου γράμματος. Π.χ. αν παραχθεί «dinnnerrrr», το κάνουμε «dinner».
 - Ένας ειδικός χαρακτήρας (ή κενό) παράγεται όταν δεν πρέπει να διαγραφούν συνεχόμενες εμφανίσεις του ίδιου γράμματος (π.χ. το «di_nn_nerrrr_» $\rightarrow$ «dinner»).

ASR με κωδικοποιητές μόνο

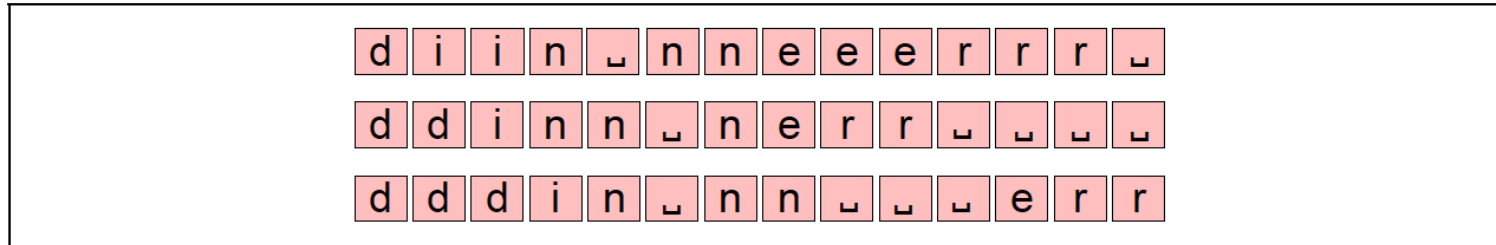


Figure 26.9 Three other legitimate alignments producing the transcript *dinner*.

- Έτσι όμως υπάρχουν **πολλαπλές ακολουθίες γραμμάτων** (πριν τη διαγραφή συνεχόμενων εμφανίσεων ίδιου γράμματος) που **όλες μπορεί να οδηγούν στη σωστή τελική ακολουθία** (μετά τη διαγραφή συνεχόμενων εμφανίσεων).
 - Η **πιθανότητα μιας τελικής ακολουθίας** ισούται με το **άθροισμα των πιθανοτήτων που δίνει το μοντέλο** σε όλες τις ακολουθίες (πριν τη διαγραφή συνεχόμενων εμφανίσεων γραμμάτων) που οδηγούν σε αυτήν.
 - **Πιο περίπλοκη η αναζήτηση** πιθανότερης τελικής ακολουθίας με **beam search**.
 - Επίσης γίνεται **πιο περίπλοκος ο υπολογισμός του σφάλματος (loss)** που δείχνει πόσο καλά τα πήγαμε. Βλ. J&M για περιγραφή του **CTC loss**.

Μέτρα αξιολόγησης

- **Λόγος λαθών λέξεων (Word Error Rate):**

- $WERR = \frac{\text{Insertions+Replacements+Deletions}}{\text{\#ReferenceWords}}$

Παράδειγμα από τις διαφάνειες των Jurafsky & Martin (2008).

Σωστή μεταγραφή (reference).

REF: portable **** PHONE UPSTAIRS last night so

HYP: portable FORM OF STORES last night so

Εξοδος συστήματος (υπόθεση).

I R R

$$WER = (1+2+0)/6 = 50\%$$

- Υπολογίζεται όπως η απόσταση Levenshtein, αλλά με κόστος 1 και για R. Το WERR μπορεί να βγει και > 1 .

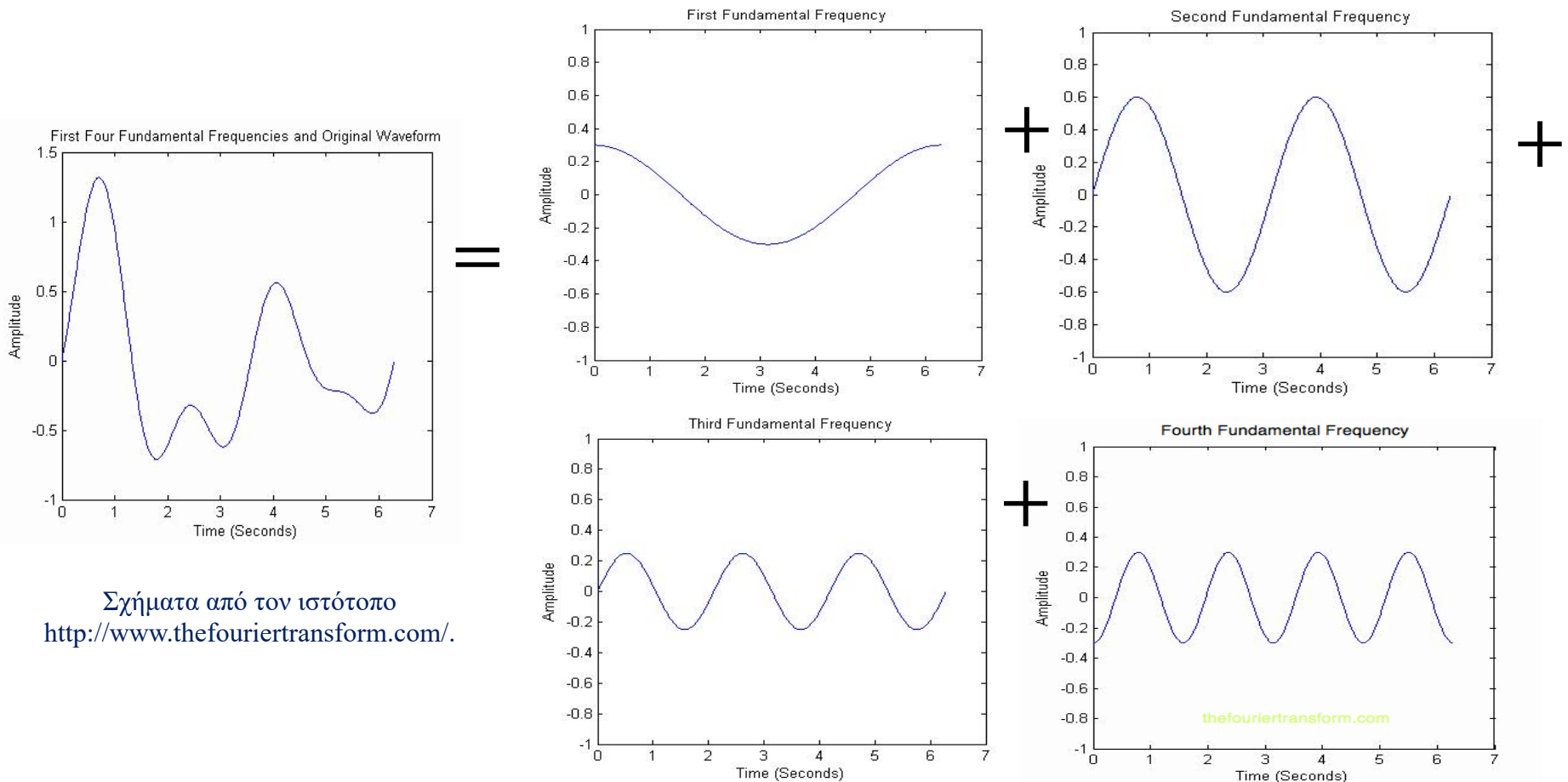
- **Λόγος λαθών προτάσεων (Sentence Error Rate):**

- Προτάσεις με ≥ 1 λάθος / πλήθος προτάσεων.

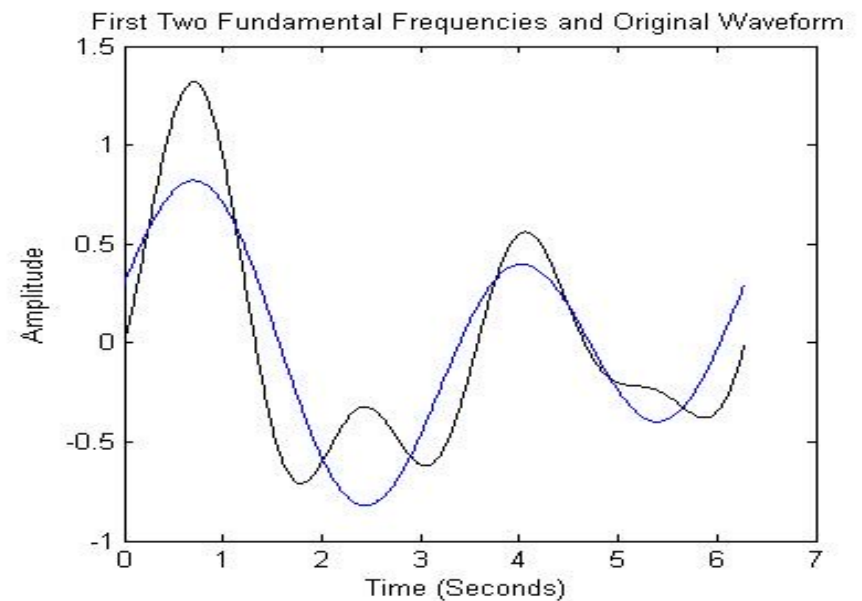
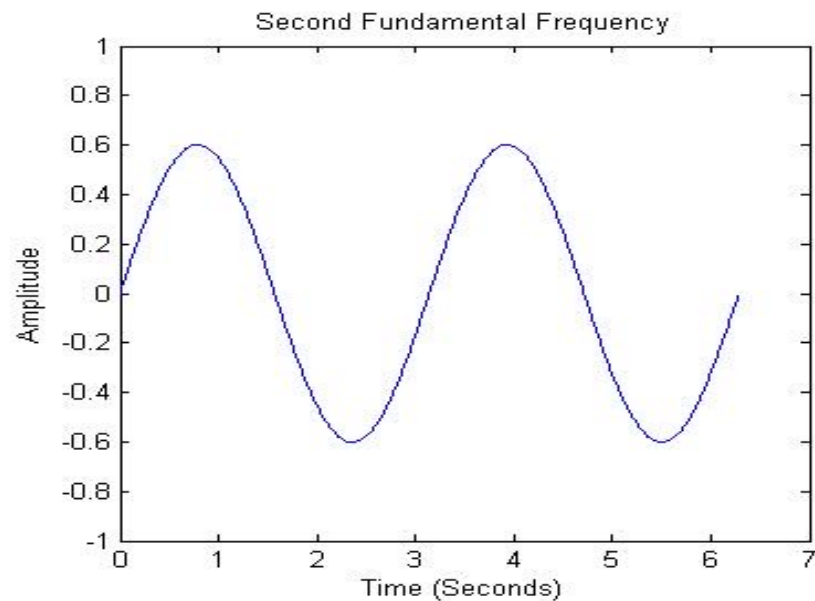
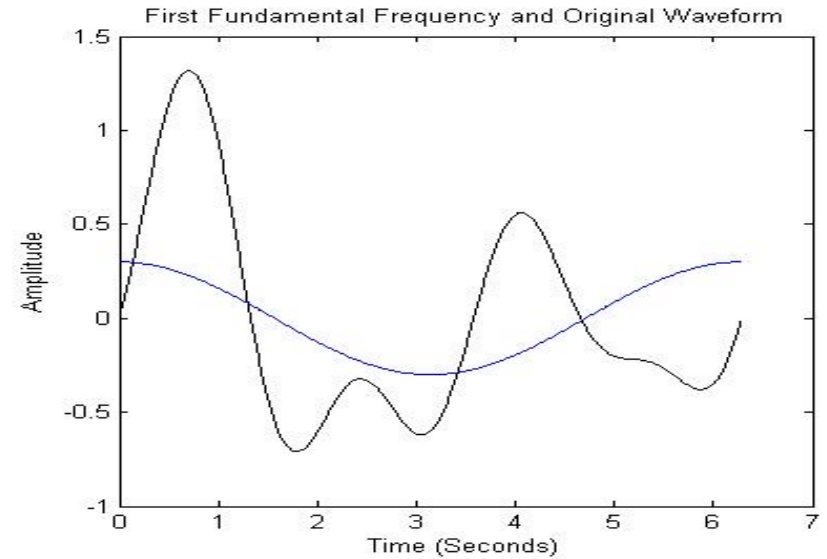
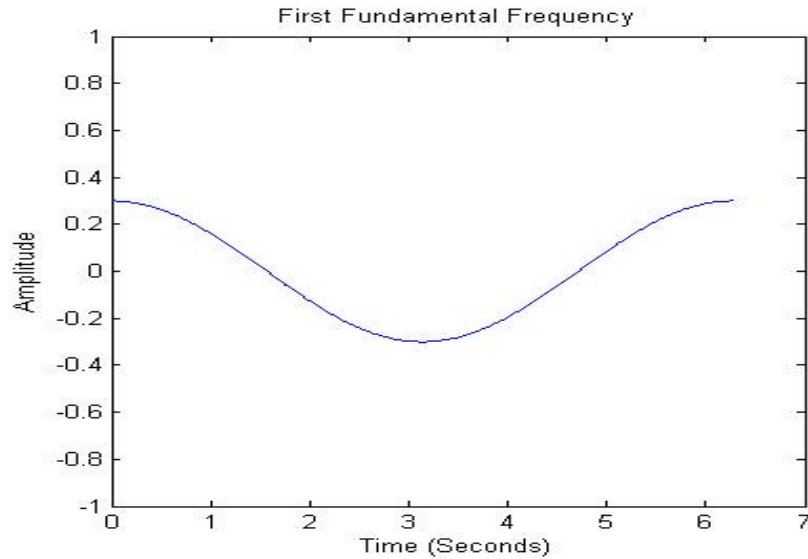
Πρόσθετες προαιρετικές διαφάνειες για
παλαιότερη τεχνολογία αναγνώρισης
ομιλίας: διανύσματα MFCC και Hidden
Markov Models (HMMs).

Μετασχηματισμός Fourier

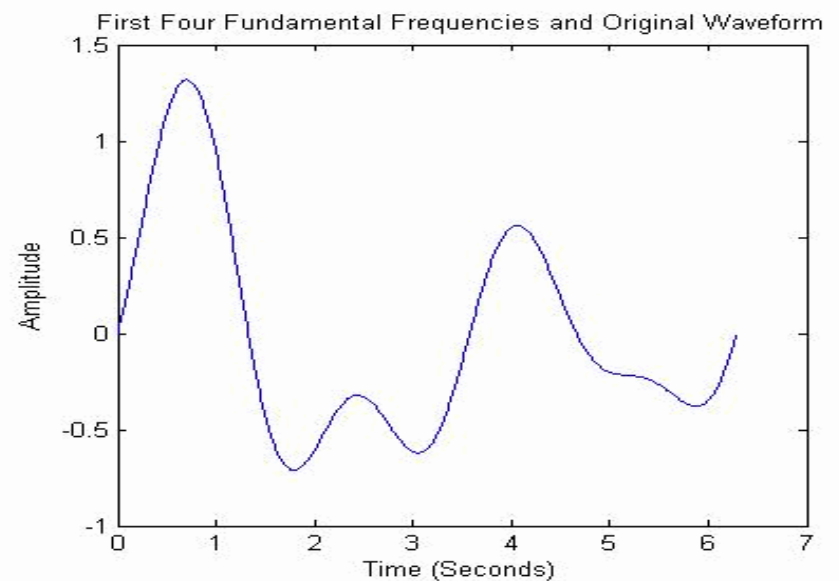
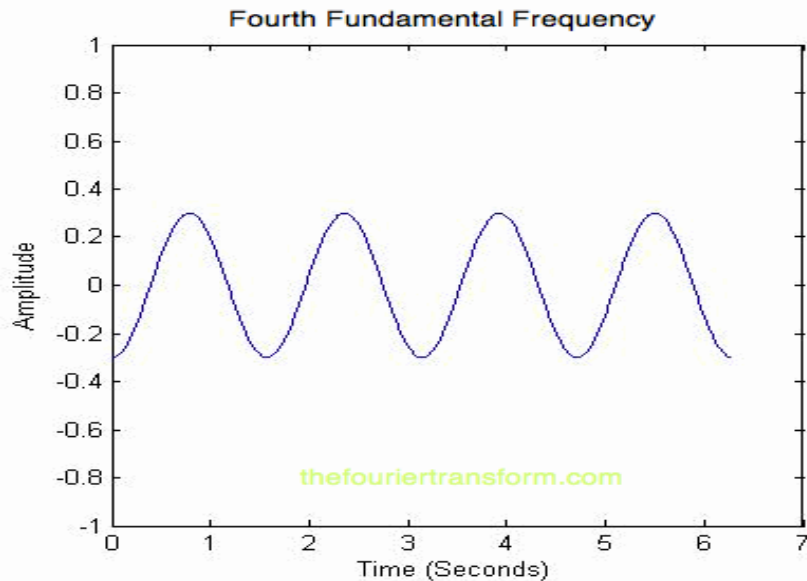
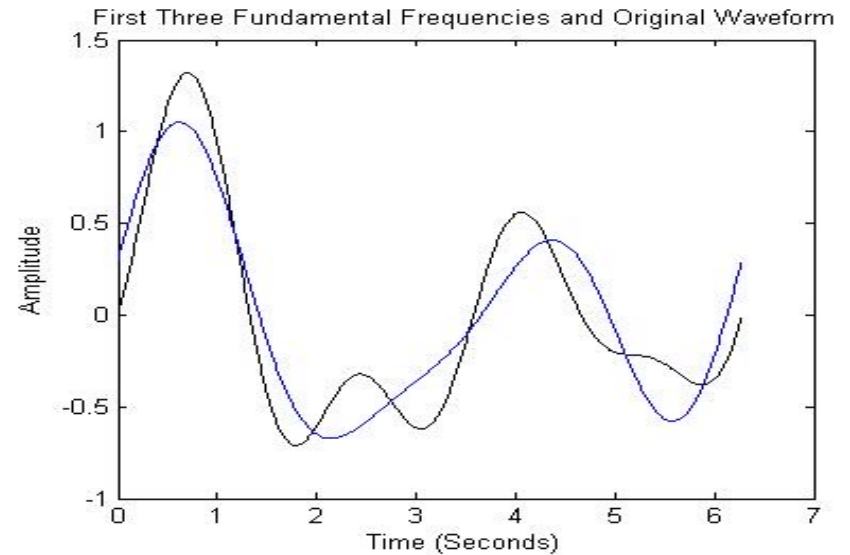
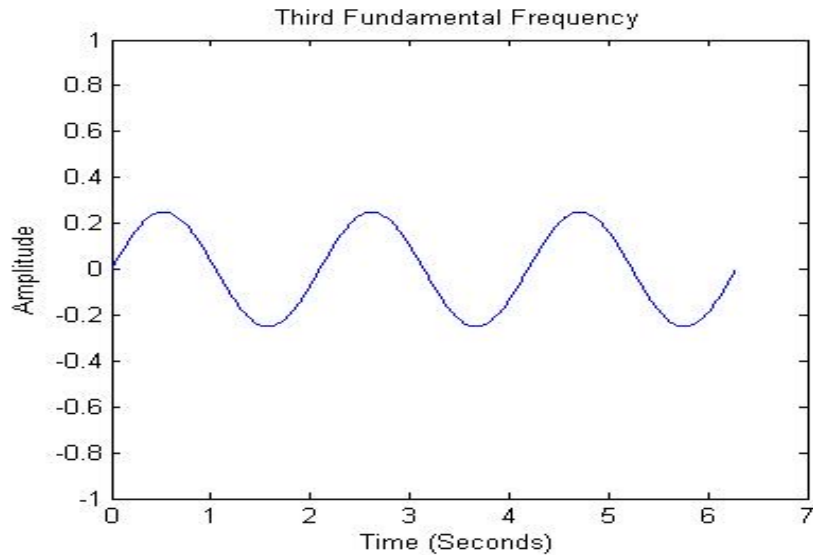
- Μπορούμε να σκεφτούμε **κάθε ήχο** (ή σήμα) ως **άθροισμα πολλών** (γενικά άπειρων) **ημιτονοειδών**.



Μετασχηματισμός Fourier

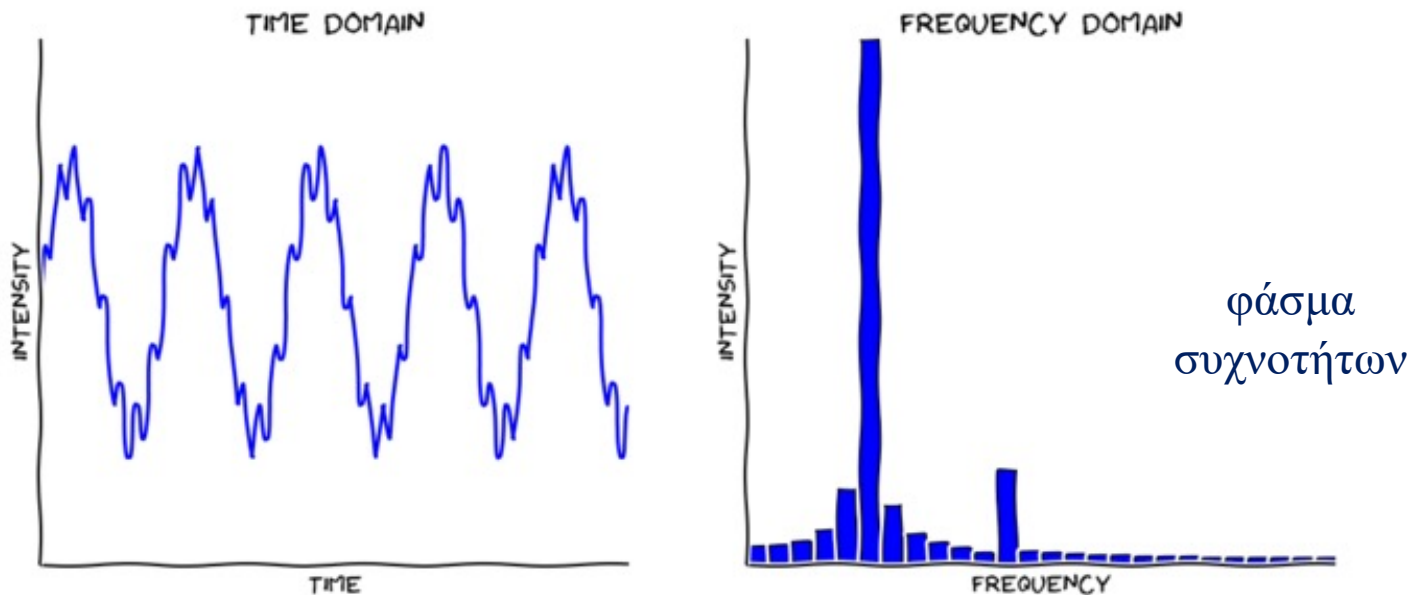


Μετασχηματισμός Fourier



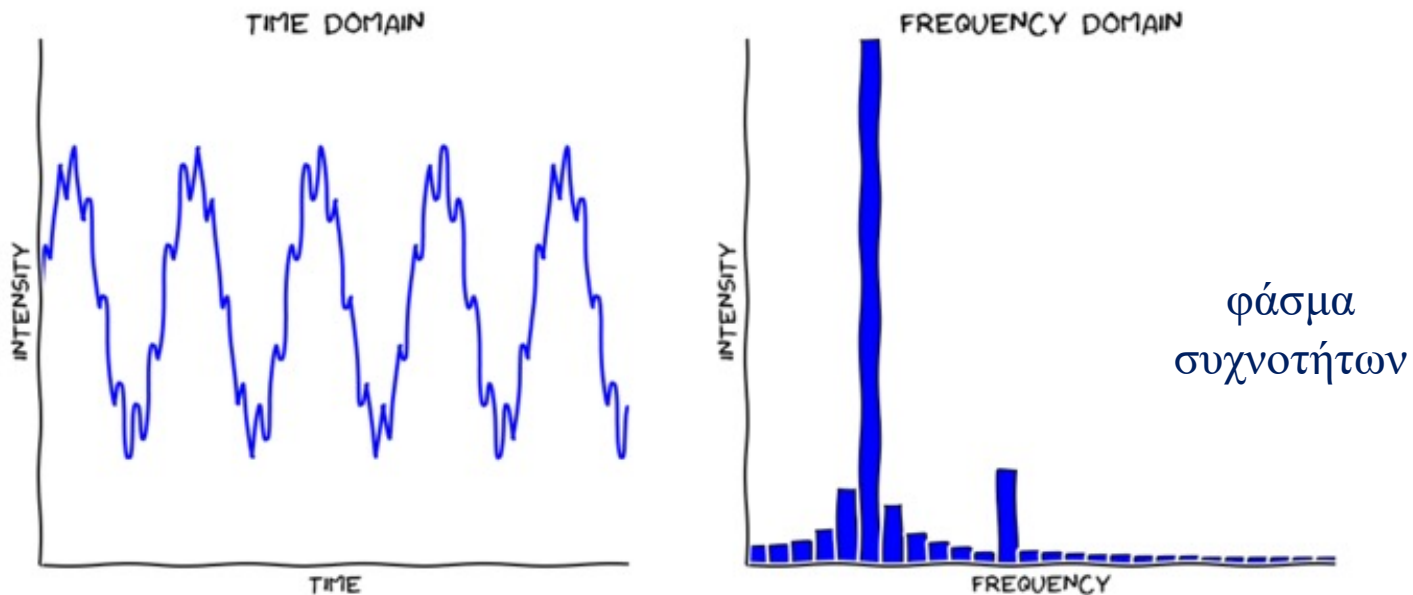
Μετασχηματισμός Fourier

- Μετατρέπει το αρχικό σήμα $f(t)$ (συνάρτηση του χρόνου t) σε μιγαδική συνάρτηση $\hat{f}(\xi)$ της συχνότητας (ξ).
 - $\hat{f}(\xi) = \int_{-\infty}^{+\infty} f(t) \cdot e^{-2\pi \cdot i \cdot t \cdot \xi} dt$ ($e^{i \cdot \theta} = \cos \theta + i \cdot \sin \theta$)
 - Το μέτρο του μιγαδικού $|\hat{f}(\xi)|$ δείχνει πόσο συμμετέχει η συχνότητα ξ στο αρχικό σήμα.



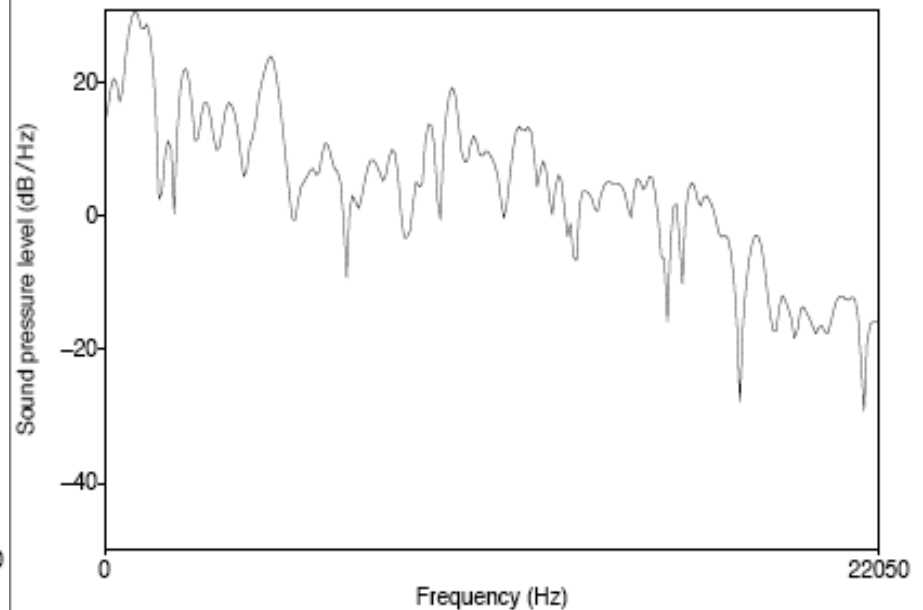
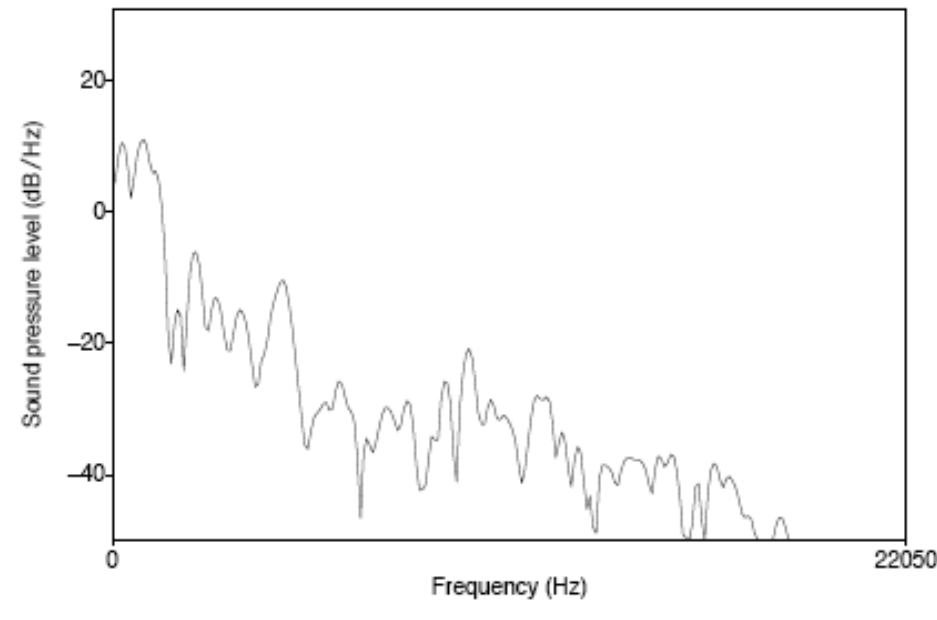
Διακριτός μετασχηματισμός Fourier (DFT)

- Για διακριτό σήμα $x[0], \dots, x[N - 1]$ και N διακριτές συχνότητες ξ :
 - $\hat{x}(\xi) = \sum_{n=0}^{N-1} x[n] \cdot e^{\frac{-2\pi \cdot i \cdot n \cdot \xi}{N}}$ ($e^{i \cdot \theta} = \cos \theta + i \cdot \sin \theta$)
 - Αν $N = 2^m$ (δύναμη του 2), μπορούμε να χρησιμοποιήσουμε τον **αλγόριθμο FFT** (Fast Fourier Transform).



Προέμφαση

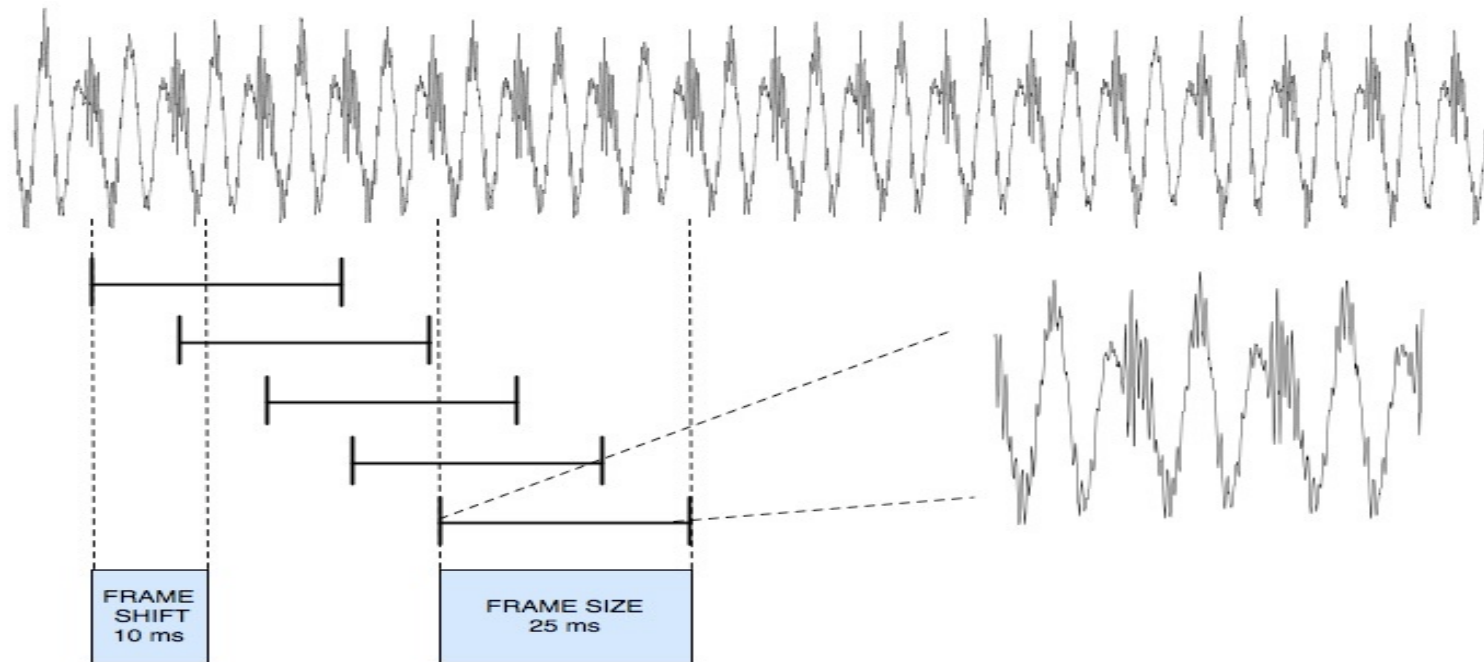
Σχήματα από τις διαφάνειες των
Jurafsky & Martin (2008).



- **Ενισχύουμε τις υψηλότερες συχνότητες της ομιλίας.**
 - Χρησιμοποιώντας **υψιπερατό φίλτρο**.
 - **Βοηθά** τη σωστή αναγνώριση ομιλίας.

Παράθυρα

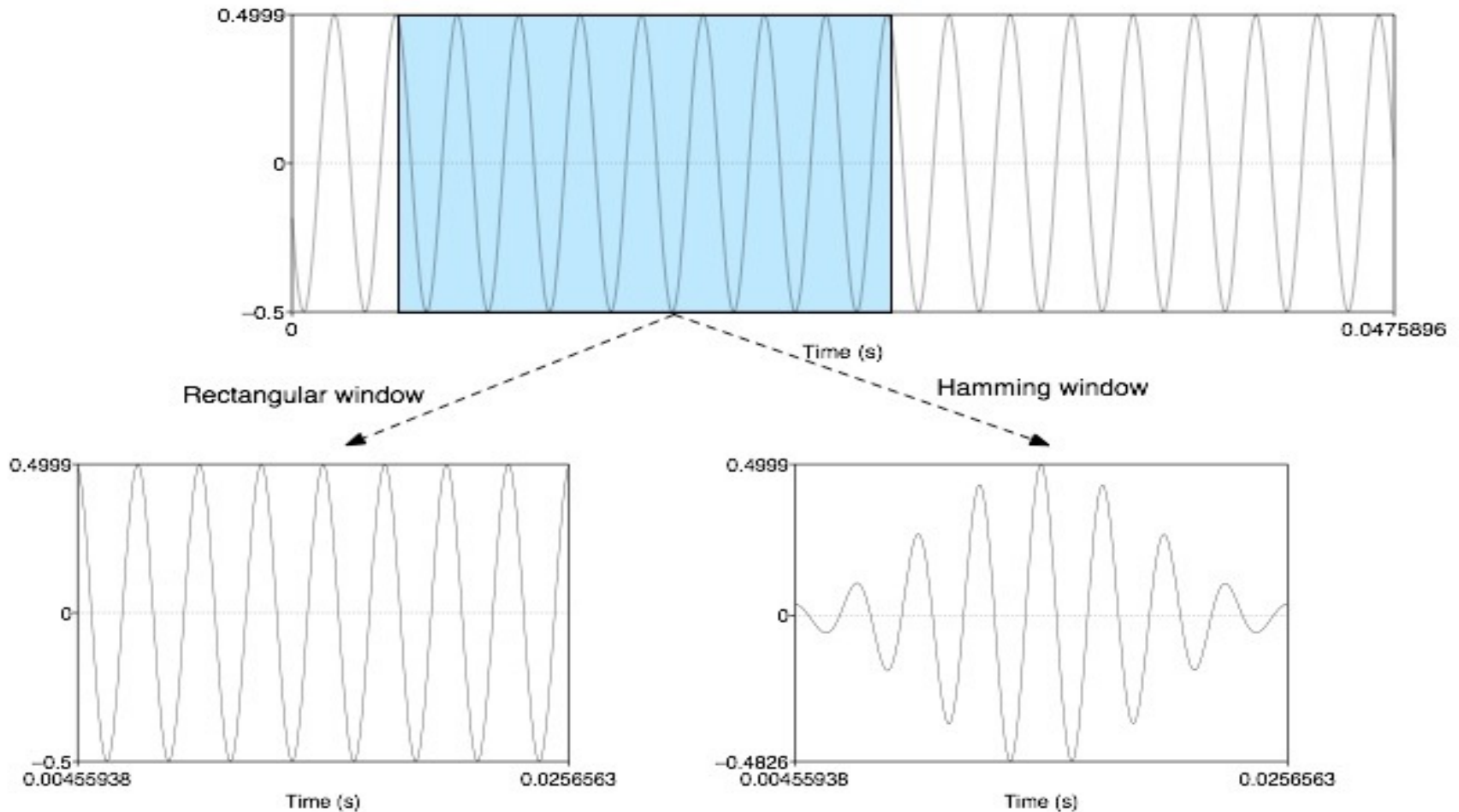
Σχήμα από τις διαφάνειες των
Jurafsky & Martin (2008).



- Εξάγουμε **επικαλυπτόμενα τμήματα (frames)** του σήματος.
 - **Σέρνουμε** ένα «παράθυρο» κατά μήκος του σήματος.
 - **Πολλαπλασιάζουμε** κάθε τιμή του (διακριτού) **σήματος** με την αντίστοιχη τιμή της **συνάρτησης του παραθύρου** (βλ. επόμενες διαφάνειες).
- **Κάθε τμήμα** συχνά παριστάνεται από **διάνυσμα 39 αριθμών**.
 - **39 MFCC features** (βλ. παρακάτω).

Παράθυρα

Σχήμα από τις διαφάνειες των
Jurafsky & Martin (2008).



- Το παράθυρο **Hamming** δίνει έμφαση στο κέντρο του τμήματος.
 - Βοηθά επίσης να αποφύγουμε **ασυνέχειες** στα άκρα των παραθύρων.

Παράθυρα

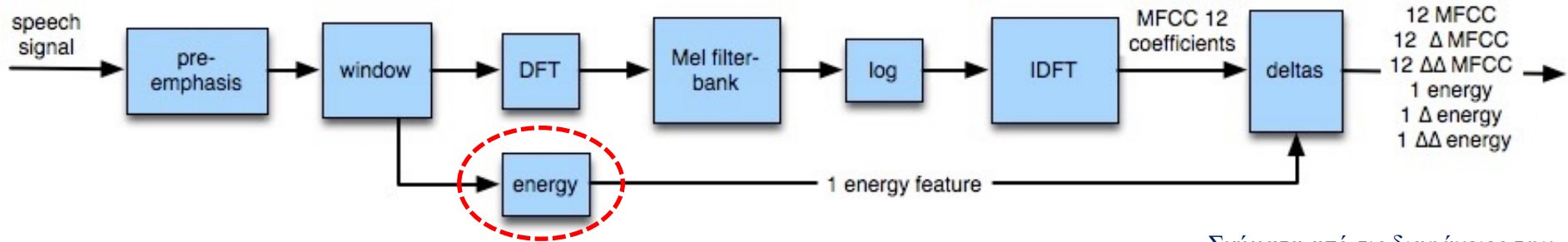
- **Τετράγωνο παράθυρο:**

$$w[n] = \begin{cases} 1 & 0 \leq n \leq L - 1 \\ 0 & \text{otherwise} \end{cases}$$

- Παράθυρο **Hamming**:

$$w[n] = \begin{cases} 0.54 - 0.46 \cos\left(\frac{2\pi n}{L-1}\right) & 0 \leq n \leq L - 1 \\ 0 & \text{otherwise} \end{cases}$$

Ενέργεια του τμήματος

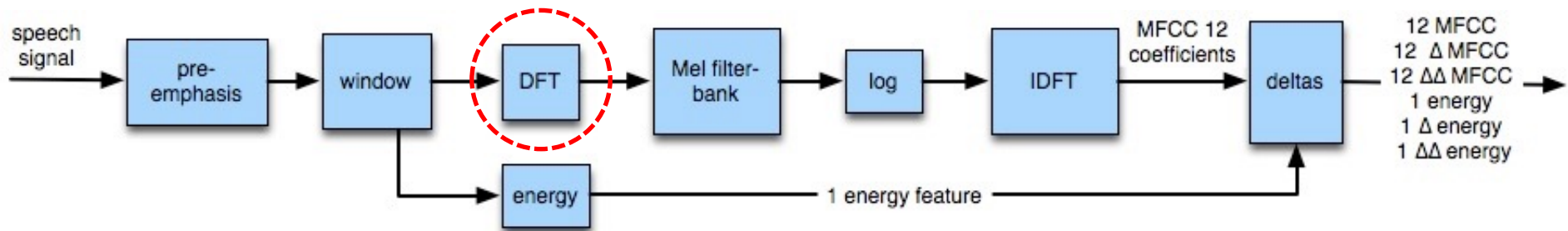


Σχήματα από τις διαφάνειες των Jurafsky & Martin (2008).

- Από **κάθε τμήμα** (εφαρμογή παραθύρου) εξάγουμε **39 αριθμούς** (τιμές ιδιοτήτων **MFCC**).
- Η τιμή μιας από τις **ιδιότητες MFCC** είναι η **ενέργεια** του τμήματος.

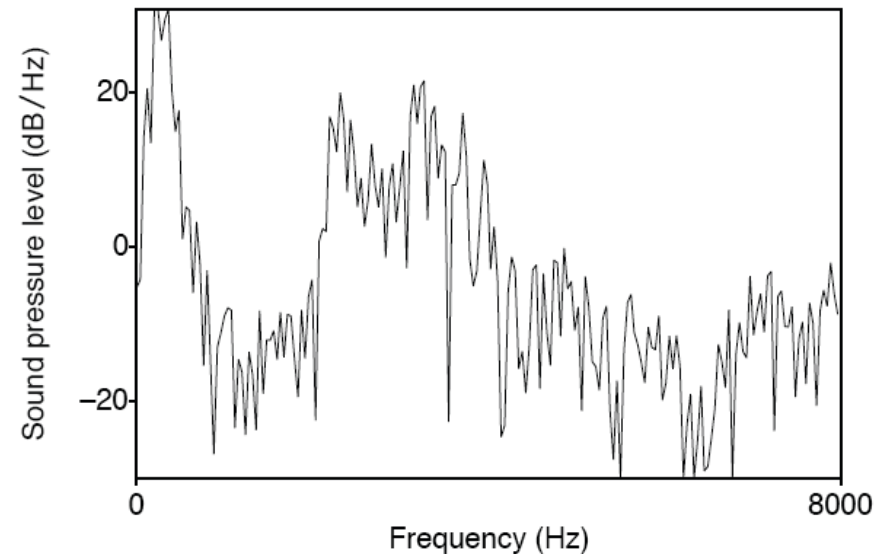
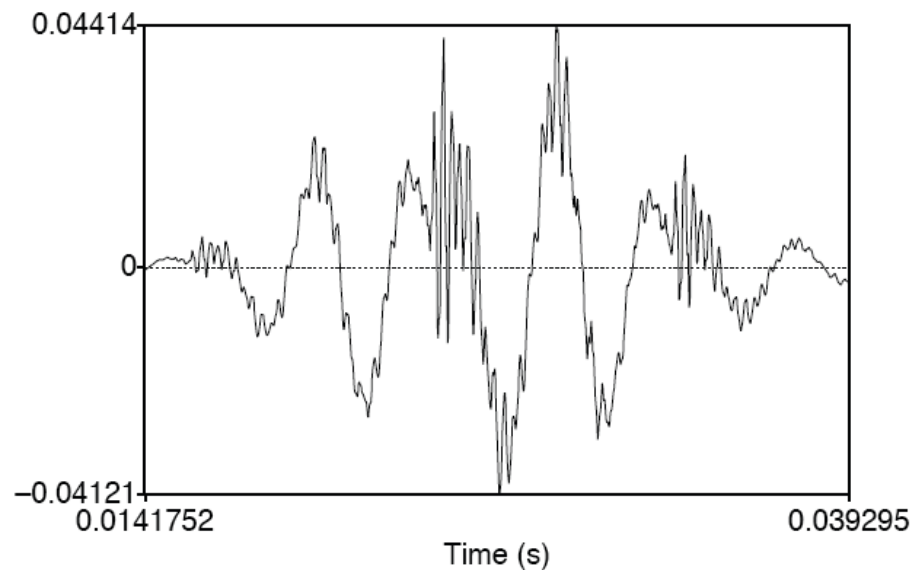
$$Energy = \sum_{n=0}^{L-1} x^2[n]$$

Μετ/μός Fourier κάθε τμήματος



Σχήματα από τις διαφάνειες των Jurafsky & Martin (2008).

- Κατόπιν εφαρμόζουμε **DFT** στο τμήμα.



Μετασχηματισμός σε φάσμα mel

- Η ακοή δεν είναι το ίδιο ευαίσθητη στις συχνότητες.
 - Λιγότερο ευαίσθητη σε συχνότητες $\geq 1 \text{ KHz}$.

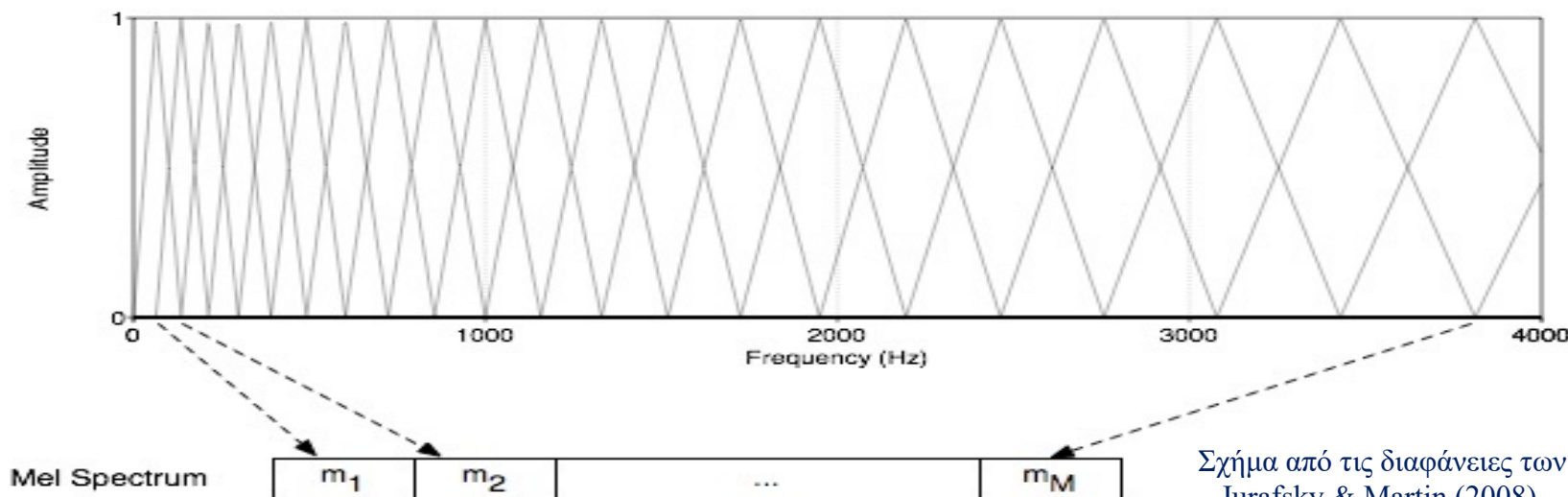
- Συστοιχία φίλτρων mel:

$$Y_t[m] = \sum_{k=1}^N W_m[k] |X_t[k]|^2$$

k : DFT bin number $(1, \dots, N)$

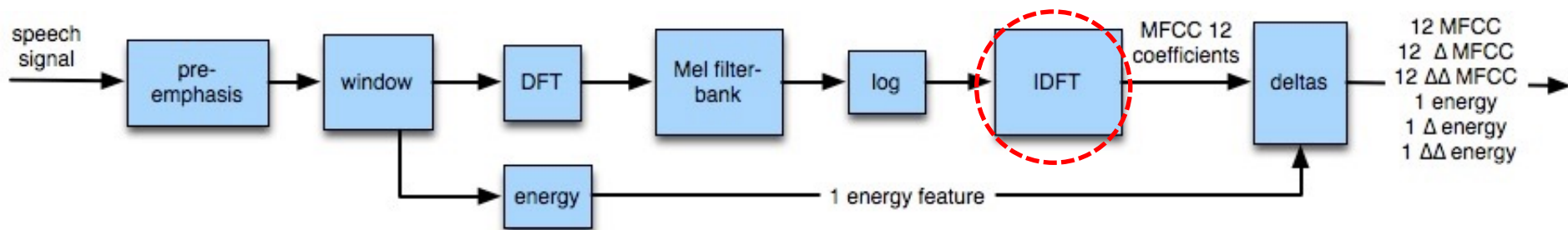
m : mel-filter bank number $(1, \dots, M)$

- Κάθε φίλτρο δρα ως τριγωνικό παράθυρο πάνω στο φάσμα.
- Τα φίλτρα είναι πιο αραιά στις μεγαλύτερες συχνότητες.



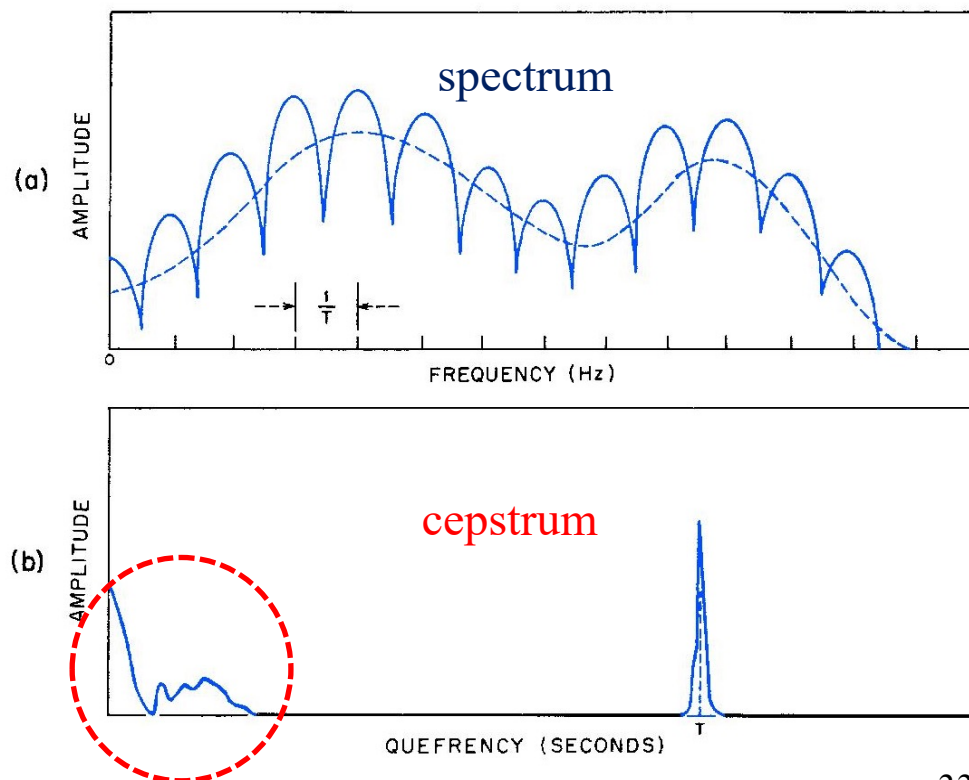
Σχήμα από τις διαφάνειες των
Jurafsky & Martin (2008).

Υπολογισμός ιδιοτήτων MFCC

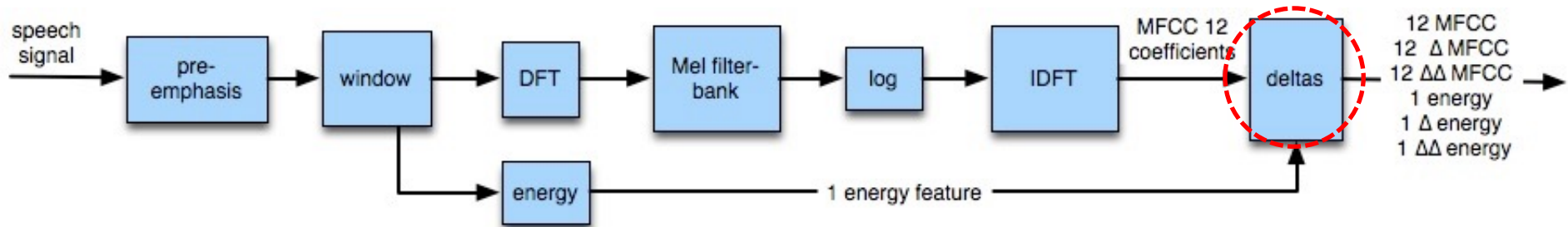


Σχήματα από τις διαφάνειες των Jurafsky & Martin (2008).

- Σκεφτόμαστε κατόπιν τα $\log(m_1), \dots, \log(m_M)$ σαν σήμα.
 - Εφαρμόζουμε DFT στο φάσμα (spectrum), για να βρούμε τις μικρές του «συχνότητες», που είναι πιο χρήσιμες στην αναγνώριση φωνής.
 - Ακριβέστερα, εφαρμόζουμε ανάστροφο DFT (IDFT), γιατί πάμε από το πεδίο συχνοτήτων πίσω στο πεδίο του χρόνου.
 - Κρατάμε τις 12 αριστερότερες τιμές του νέου «φάσματος» (cepstrum).



Μεταβολές (Δ και $\Delta\Delta$)

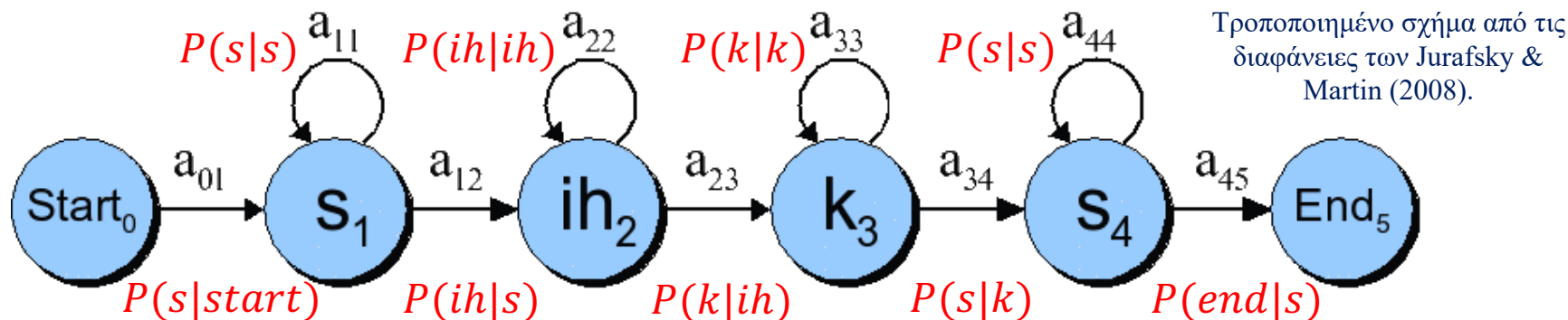


Σχήματα από τις διαφάνειες των Jurafsky & Martin (2008).

- Προσθέτουμε μεταβολές (Δ):
 - Απλούστερη περίπτωση: πόσο άλλαξε η ενέργεια από το προηγούμενο τμήμα (frame) και πόσο άλλαξε κάθε μία από τις άλλες 12 τιμές MFCC. Συνήθως πιο περίπλοκοι υπολογισμοί.
- Προσθέτουμε μεταβολές μεταβολών ($\Delta\Delta$):
 - Στην απλούστερη περίπτωση: πόσο άλλαξε το Δ της ενέργειας, το Δ κάθε μιας από τις 12 τιμές MFCC κλπ.
- Συνολικά 39 τιμές ανά τμήμα.
 - Κάθε τμήμα παριστάνεται από ένα διάνυσμα 39 αριθμών.

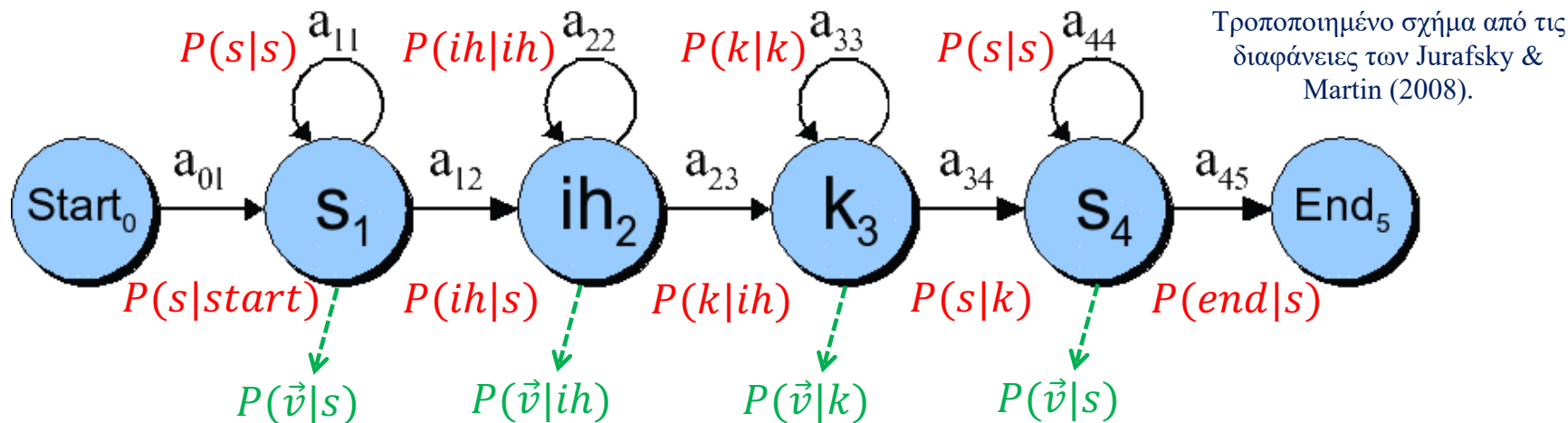
Κρυφά Μαρκοβιανά Μοντέλα

(Hidden Markov Models, HMMs)



- Θεωρούμε ότι ο ομιλητής παράγει την ακολουθία τμημάτων (διανυσμάτων MFCC) ακολουθώντας ένα μονοπάτι.
 - Οι **καταστάσεις** αντιστοιχούν σε «φώνους» (phones).
 - Π.χ. το «six» προφέρεται [s ih k s].
- Σε κάθε βήμα πηγαίνει σε νέα κατάσταση (ή μένει στην ίδια) με τις κόκκινες πιθανότητες.
 - Μπορεί π.χ. να πει [s s ih ih ih k s s] ή [s s s ih ih ih ih ih k s s s].

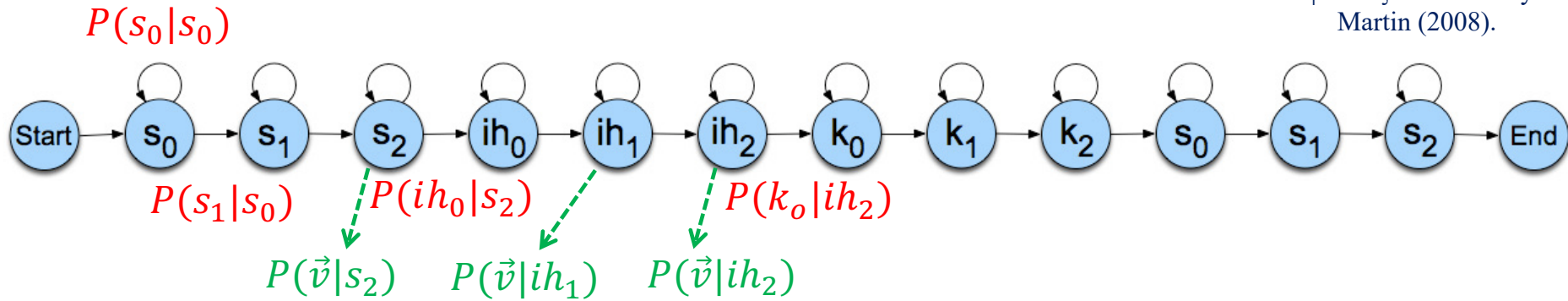
Κρυφά Μαρκοβιανά Μοντέλα



- Όποτε πηγαίνει (ή μένει) σε μια **κατάσταση**, ο ομιλητής παράγει ένα **τμήμα** (διάνυσμα MFCC) σύμφωνα με **κατανομή πιθανοτήτων** που **εξαρτάται** από την **κατάσταση**.
 - Τα **διανύσματα** δεν αντιστοιχούν 1-1 με τις **καταστάσεις**.
 - **Διαφορετικά διανύσματα** μπορεί να παραχθούν από την **ίδια κατάσταση** σε **διαφορετικές επισκέψεις** της κατάστασης.
 - Οι **πράσινες πιθανότητες** («εκπομπής») δείχνουν **πόσο πιθανό** είναι να παραχθεί **κάθε διάνυσμα** στη **συγκεκριμένη κατάσταση**.

HMM με υπο-φώνους

Τροποποιημένο σχήμα από τις διαφάνειες των Jurafsky & Martin (2008).



- Συνήθως χρησιμοποιούνται **τρεις διαφορετικές καταστάσεις** (υπο-φώνοι) **ανά φώνο**, αντί για μία κατάσταση ανά φώνο.
 - Γιατί ο **ίδιος φώνος** συχνά παράγει **διαφορετικά διανύσματα MFCC** στην **αρχή**, τη **μέση** και το **τέλος** της **προφοράς** του.
- **Δεν παράγονται** διανύσματα MFCC στις **start** και **end**.
- Στην **αναγνώριση ομιλίας**, κάθε κατάσταση του HMM συνήθως έχει **μεταβάσεις** μόνο προς μια **δεξιότερη κατάσταση** ή την **ίδια κατάσταση**.
 - Σε **άλλες εφαρμογές** των HMMs **δεν ισχύει** πάντα αυτό.

HMM για αριθμούς

Lexicon

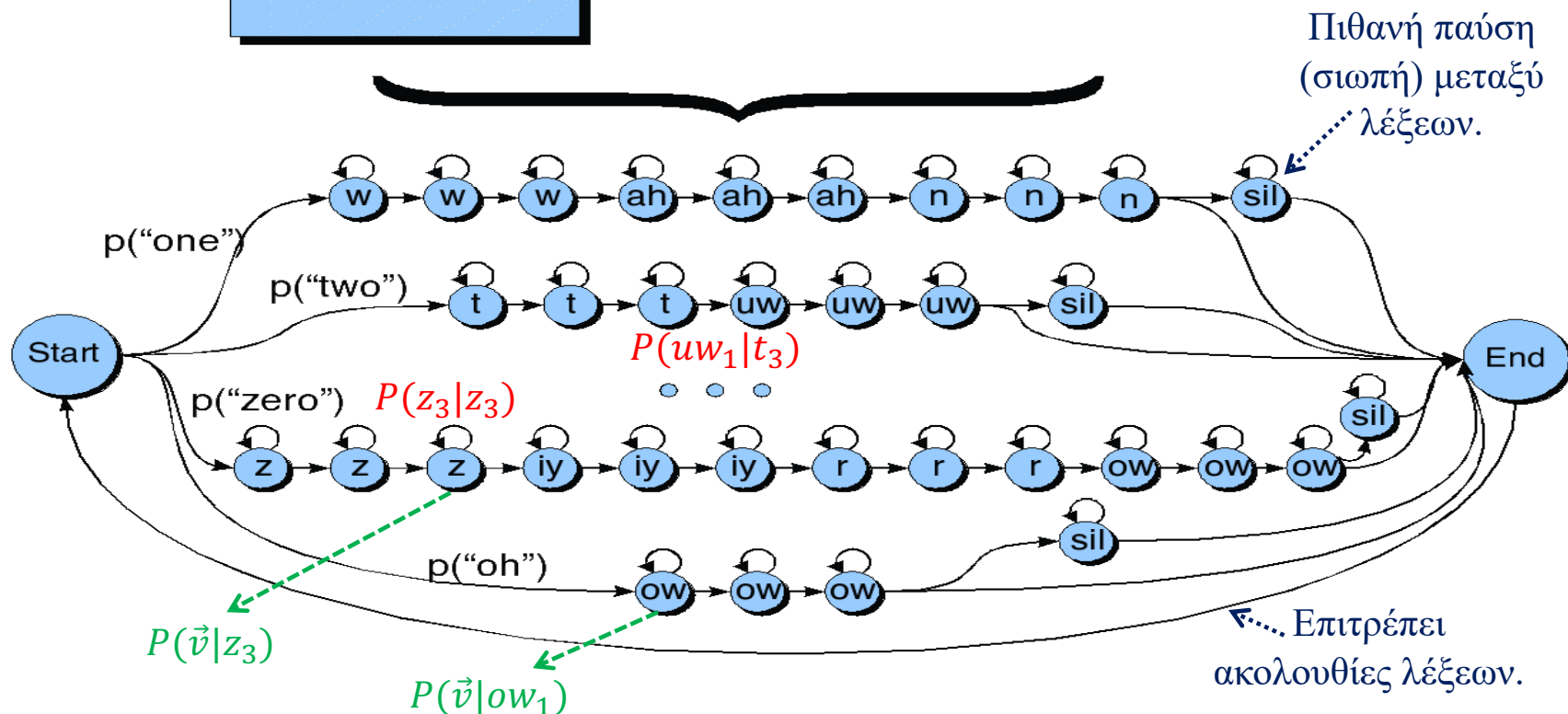
one	w ah n
two	t uw
three	th r iy
four	f ao r
five	f ay v
six	s ih k s
seven	s eh v ax n
eight	ey t
nine	n ay n
zero	z iy r ow
oh	ow

Βλ. π.χ.
<http://www.speech.cs.cmu.edu/cgi-bin/cmudict>

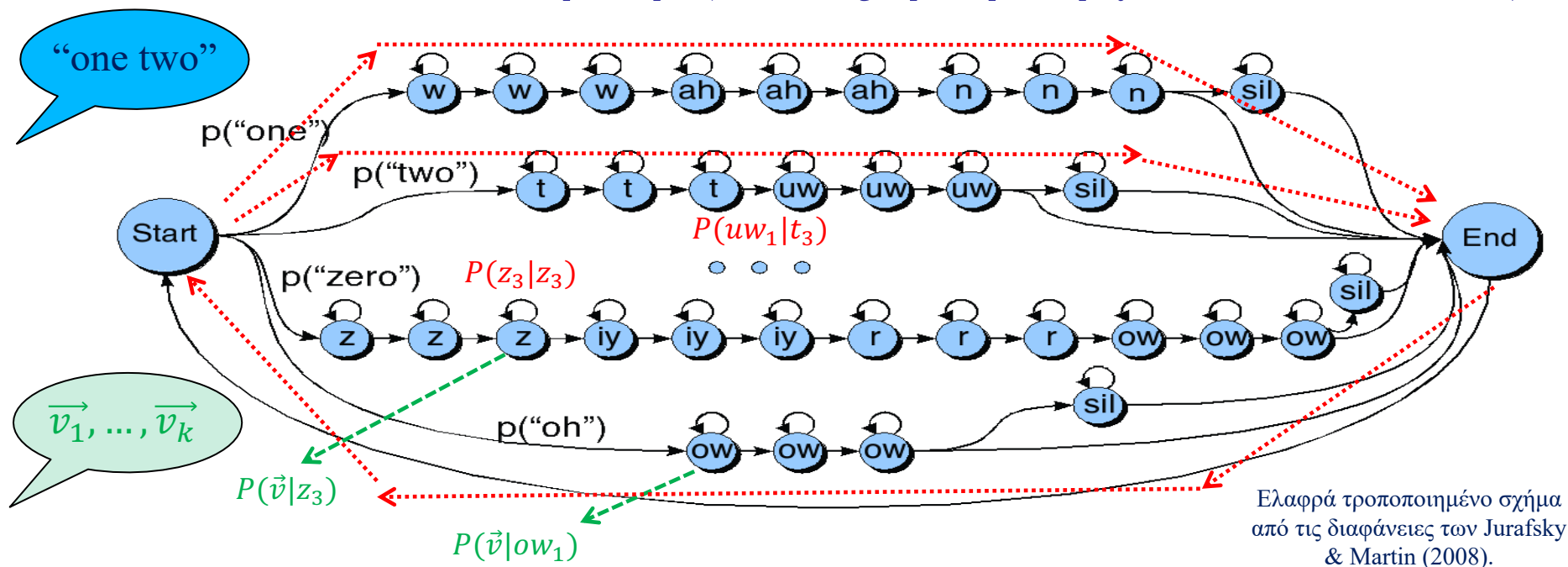
Phone HMM



Ελαφρά τροποποιημένο σχήμα
 από τις διαφάνειες των Jurafsky
 & Martin (2008).



Αποκωδικοποίηση (αναζήτηση μονοπατιού)



- Ψάχνουμε το πιθανότερο μονοπάτι που μπορεί να παρήγαγε την παρατηρούμενη ακολουθία διανυσμάτων MFCC.
 - Ουσιαστικά την **πιθανότερη ακολουθία καταστάσεων**, άρα και λέξεων.
 - Δεν ξέρουμε άμεσα ποιο μονοπάτι χρησιμοποιήθηκε γιατί **δεν υπάρχει 1-1 αντιστοιχία** μεταξύ καταστάσεων και παρατηρούμενων διανυσμάτων.
 - Το μονοπάτι είναι «κρυμμένο» από τον παρατηρητή.

Αποκωδικοποίηση (αναζήτηση μονοπατιού)

- Παρατηρούμενη ακολουθία διανυσμάτων MFCC:

$$\vec{v}_1^k = \langle \vec{v}_1, \vec{v}_2, \dots, \vec{v}_k \rangle$$

- Μια οποιαδήποτε ακολουθία καταστάσεων ίσου μήκους:

$$s_1^k = \langle s_1, s_2, \dots, s_k \rangle$$

- Θέλουμε την (κρυφή) ακολουθία καταστάσεων που είναι πιθανότερο να οδήγησε στην ακολουθία διανυσμάτων:

$$\hat{s}_1^k = \operatorname{argmax}_{s_1^k} P(s_1^k | \vec{v}_1^k) = \operatorname{argmax}_{s_1^k} \frac{P(s_1^k) \cdot P(\vec{v}_1^k | s_1^k)}{\cancel{P(\vec{v}_1^k)}}$$

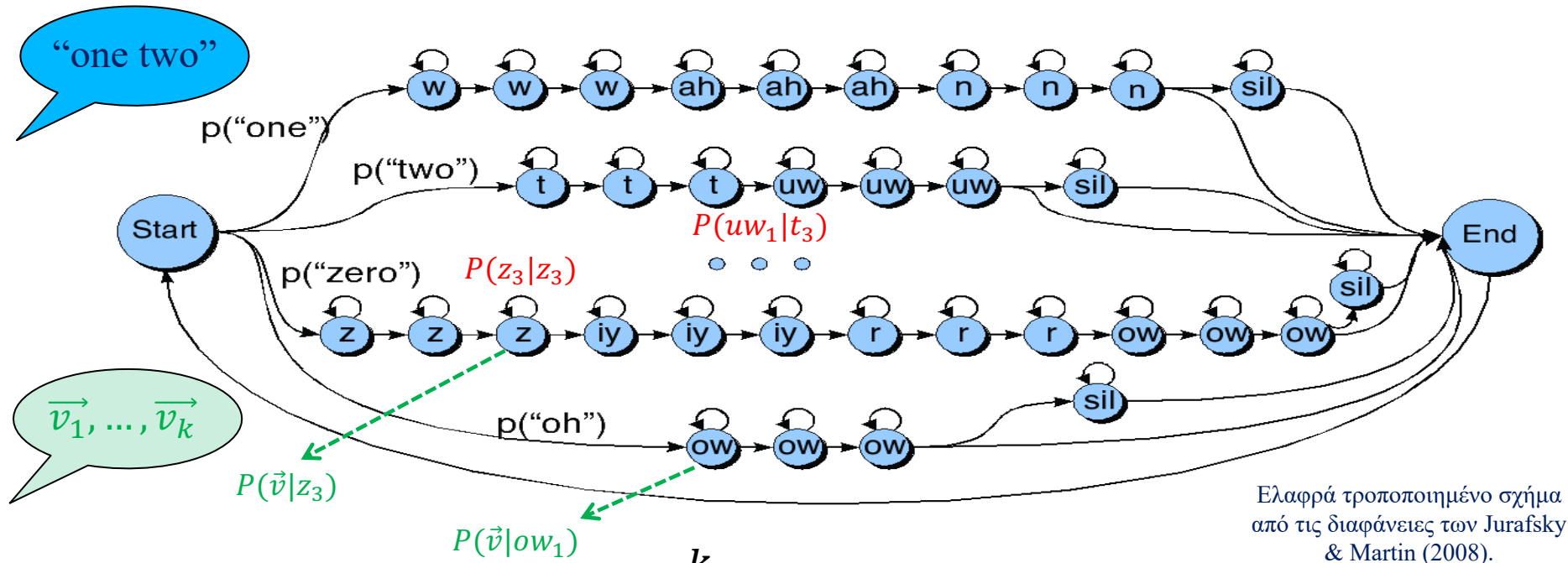
- Χρησιμοποιήσαμε τον κανόνα του Bayes.
- Ο παρονομαστής είναι ο ίδιος για κάθε s_1^k .

Η πιθανότερη ακολουθία καταστάσεων

$$\hat{s}_1^k = \underset{s_1^k}{\operatorname{argmax}} P(s_1^k | \vec{v}_1^k) = \underset{s_1^k}{\operatorname{argmax}} \boxed{P(s_1^k)} \cdot \boxed{P(\vec{v}_1^k | s_1^k)} =$$
$$\underset{s_1^k}{\operatorname{argmax}} \boxed{P(s_1) \cdot P(s_2 | s_1) \cdot P(s_3 | s_1, s_2) \cdot P(s_4 | s_1^3) \cdots P(s_k | s_1^{k-1}) \cdot}$$
$$\boxed{P(\vec{v}_1 | s_1^k) \cdot P(\vec{v}_2 | \vec{v}_1, s_1^k) \cdot P(\vec{v}_3 | \vec{v}_1, \vec{v}_2, s_1^k) \cdots P(\vec{v}_k | \vec{v}_1^{k-1}, s_1^k)}$$

- 1^η απλούστευση: $P(s_i | s_1, \dots, s_{i-1}) \cong P(s_i | s_{i-1})$
 - HMM 1^{ης} τάξης: η πιθανότητα μετάβασης στην κατάσταση s_i εξαρτάται μόνο από την προηγούμενη κατάσταση s_{i-1} .
 - Γενικότερα HMM n -στής τάξης: εξαρτάται από τις n προηγούμενες.
- 2^η απλούστευση: $P(\vec{v}_i | \vec{v}_1^{i-1}, s_1^k) \cong P(\vec{v}_i | s_i)$
 - Θεωρούμε ότι η πιθανότητα εκπομπής ενός διανύσματος $\vec{v}_i$ εξαρτάται μόνο από την κατάσταση s_i στην οποία βρισκόμαστε.

Αποκωδικοποίηση (αναζήτηση μονοπατιού)



Θεωρώντας $t_0 = \text{start}$.

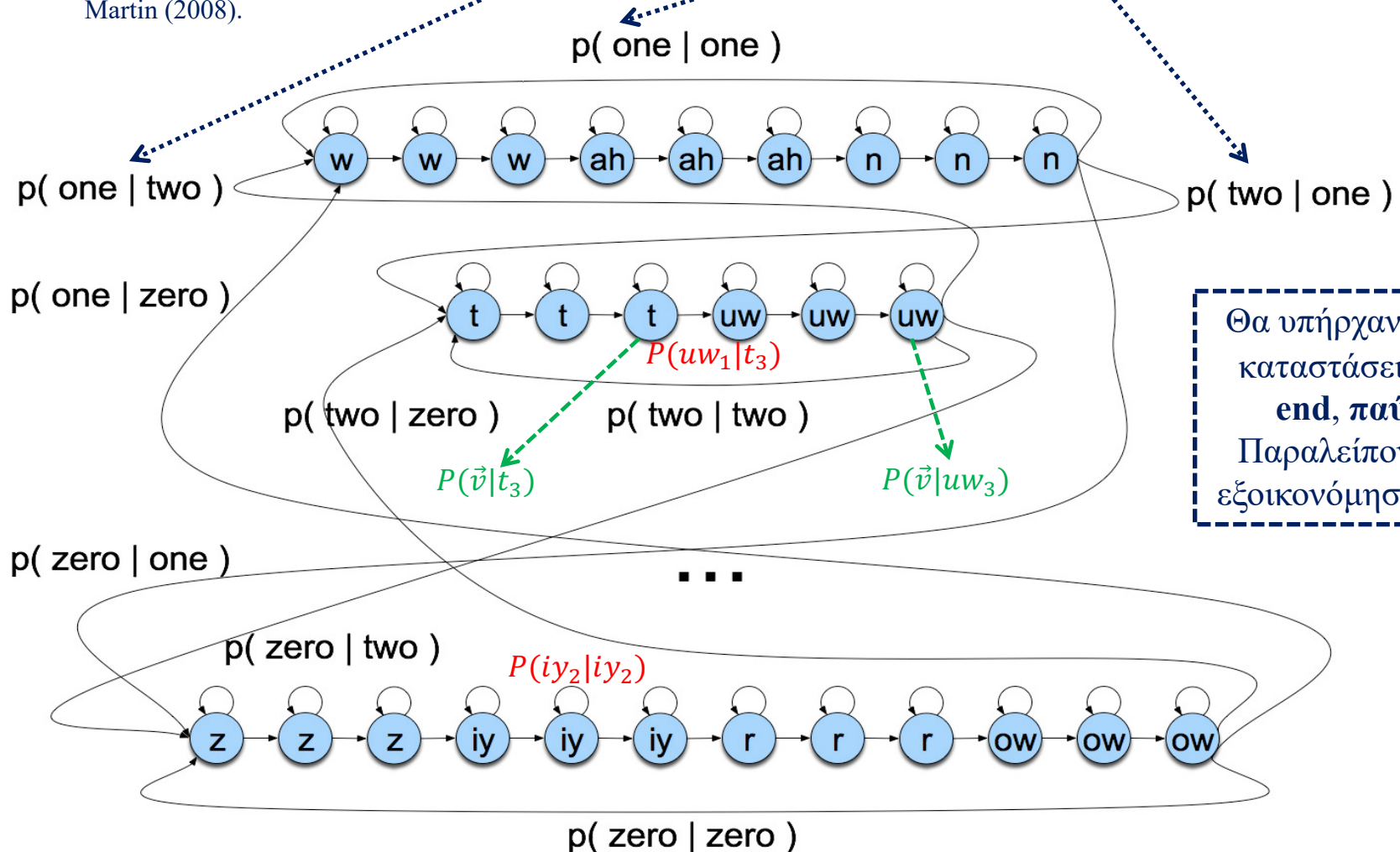
$$\hat{s}_1^k = \underset{s_1^k}{\operatorname{argmax}} \prod_{i=1}^k P(s_i | s_{i-1}) \cdot P(\vec{v}_i | s_i)$$

- Ο υπολογισμός γίνεται με **δυναμικό προγραμματισμό**.
 - Αλγόριθμος **Viterbi**. Βλ. παραπομπές.
- Εδώ **αγνοούμε** στο γινόμενο τις **μεταβάσεις** από την **end** στη **start**.

Προσθήκη γλωσσικού μοντέλου

Πριν αγνοούσαμε τις μεταβάσεις μεταξύ λέξεων. Τώρα λαμβάνουμε υπόψη **πόσο πιθανό είναι η κάθε λέξη να ακολουθεί μια άλλη.**

Τροποποιημένο σχήμα από τις διαφάνειες των Jurafsky & Martin (2008).

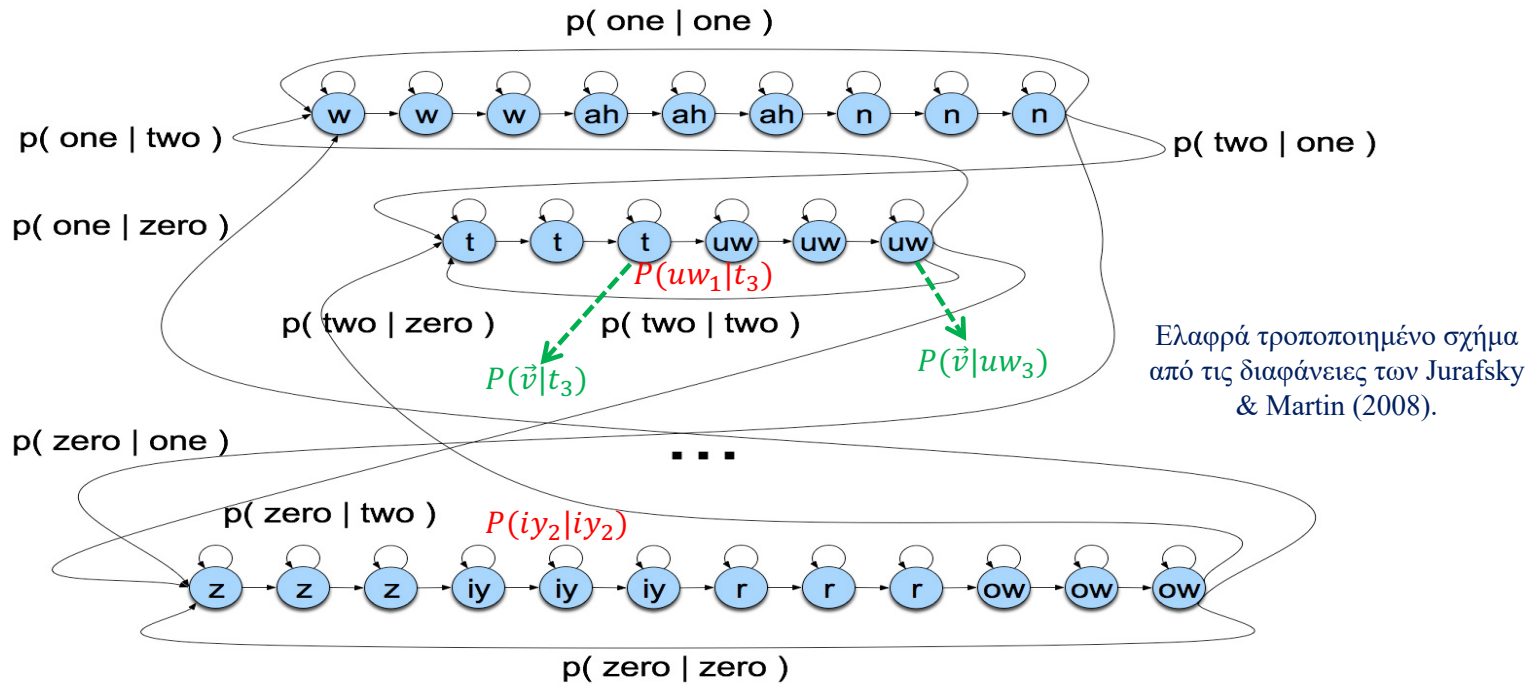


Θα υπήρχαν και εδώ καταστάσεις **start**, **end**, **παύσης**. Παραλείπονται για εξοικονόμηση χώρου.

Αποκωδικοποίηση (τόρα και με γλωσσικό μοντέλο)

“one two”

$\vec{v}_1, \dots, \vec{v}_k$



Θεωρώντας $t_0 = \text{start}$.

$$\hat{s}_1^k = \underset{s_1^k}{\operatorname{argmax}} \left(\prod_{i=1}^k P(s_i | s_{i-1}) \cdot P(\vec{v}_i | s_i) \right) \cdot \left(\prod_{j=1}^m P(w_j | w_{j-1}) \right)$$

Οι πιθανότητες των **μεταβάσεων** από το **τέλος** κάθε λέξης στην **αρχή** μιας **επόμενης**. Ουσιαστικά **γλωσσικό μοντέλο 2-γραμμάτων**.

Υποθέτουμε εδώ ότι όταν πηγαίνουμε από την τελευταία κατάσταση μιας λέξης στην πρώτη κατάσταση μιας άλλης λέξης, **δεν εκπέμπεται διάνυσμα**.

Αποκωδικοποίηση (τόρα και με γλωσσικό μοντέλο)

Γενικότερα, αν δεν έχουμε γλωσσικό μοντέλο 2-γραμμάτων, αλλά π.χ. 3-γραμμάτων.

$$\hat{s}_1^k = \operatorname{argmax}_{s_1^k} \left(\prod_{i=1}^k P(s_i | s_{i-1}) \cdot P(\vec{v}_i | s_i) \right) \cdot \text{LMScore}(w_1^m)$$

Στην πράξη δουλεύουμε με **λογαρίθμους** (αποφεύγουμε πολλαπλασιασμούς πιθανοτήτων). Επίσης δίνουμε **βάρος λ** στο γλωσσικό μοντέλο.

$$\operatorname{argmax}_{s_1^k} \sum_{i=1}^k \log P(s_i | s_{i-1}) + \log P(\vec{v}_i | s_i) + \lambda \cdot \log \text{LMScore}(w_1^m) + m \cdot C$$

Διόρθωση για να **μην προτιμώνται** προτάσεις με **λίγες μεγάλες λέξεις** (ευνοούνται από το γλωσσικό μοντέλο). $C > 0$, m το πλήθος των λέξεων.

Εκπαίδευση του HMM

- Τις **πιθανότητες μεταβάσεων** $P(s_i | s_{i-1})$ και **εκπομπής** $P(\vec{v}_i | s_i)$ τις **μαθαίνουμε** κατά την εκπαίδευση του HMM.
 - Απαιτείται **σώμα (corpus) μεταγεγραμμένων ομιλιών (εκφωνήματα και αντίστοιχο κείμενο)**.

Σχήμα από τις διαφάνειες των
Jurafsky & Martin (2008).

Transcription

Nine four oh two two

Wavefile



- Εκπαίδευση με τον αλγόριθμο εκπαίδευσης HMM **Forward-Backward** (Baum Welch), αλλά με ειδικές βελτιώσεις για ομιλία.
- Οι $P(\vec{v}_i | s_i)$ μοντελοποιούνται ως **μίγματα πολυμεταβλητών κανονικών κατανομών** (Gaussian Mixture Models, **GMM**), οπότε μαθαίνουμε τις παραμέτρους τους (μ , σ , βάρος κάθε καμπάνας).
- Πιο πρόσφατα χρησιμοποιούνται **νευρωνικά δίκτυα** (deep neural nets, **DNN**) για τις $P(\vec{v}_i | s_i)$ ή/και αντί των HMM.

Διάβασμα

- Το μεγαλύτερο μέρος της ύλης αυτής της ενότητας καλύπτεται από το κεφάλαιο 15 του βιβλίου «Speech and Language Processing» των Jurafsky & Martin, 3^η έκδοση (υπό προετοιμασία).
 - <http://web.stanford.edu/~jurafsky/slp3/>
 - Όσοι ενδιαφέρεστε ιδιαίτερα, διαβάστε και το κεφάλαιο 16, που καλύπτει τη σύνθεση ομιλίας.
 - Τα MFCC features περιγράφονται εκτενέστερα στη 2^η έκδοση (υπάρχει στη βιβλιοθήκη του ΟΠΑ).

