ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Συνεχή Γλωσσικά Μοντέλα Με Σημασιολογική Και Συντακτική Πληροφορία

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΔΗΜΗΤΡΗ ΚΑΓΙΑΡΑ

Επιβλέπων : Βασίλης Διγαλάκης
Καθηγητής Π.Κ

Χανιά, Οκτώβριος 2013

Ευχαριστίες

Θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες στον επιβλέποντα καθηγητή μου κ. Βασίλη Διγαλάκη για την πολύτιμη βοήθεια και υποστήριξη που μου προσέφερε κατά τη διάρκεια εκπόνησης της προπτυχιακής μου διατριβής, καθώς επίσης και για το έναυσμα που μου έδωσε ώστε να ασχοληθώ με τα γλωσσικά μοντέλα. Καθώς και τον κ. Βασίλη Διακολουκά, υπεύθυνο του τομέα τηλεπικοινωνιών, για την άριστη συνεργασία, καθοδήγηση και κατανόηση που επέδειξε για την ολοκλήρωση της παρούσας προπτυχιακής διατριβής. Επιπρόσθετα, τις θερμότερες ευχαριστίες μου οφείλω να εκφράσω στα μέλη της οικογένειάς μου, στους φίλους μου και τους συμφοιτητές μου για τη ψυχολογική στήριξη που μου παρείχαν σε όλη τη διάρκεια των σπουδών μου.

Περίληψη

Τα συνεχή γλωσσικά μοντέλα βασίζονται στην πληροφορία συνύπαρξης διαδοχικών λέξεων (Co-occurrence). Στη διατριβή αυτή, ασχολούμαστε με την εκπαίδευση συνεχών γλωσσικών μοντέλων, κάνοντας χρήση της σημασιολογικής και της συντακτικής τους πληροφορίας.

Η σημασιολογική πληροφορία των λέξεων αντλείται μέσα από κάποιες μετρικές ομοιότητας. Οι μετρικές ομοιότητας που χρησιμοποιούμε είναι οι : Cosine Similarity, Jaccard Similarity, Dice Coefficient Similarity, Overlap Coefficient Similarity, Normalized Google Distance. Επίσης, η συντακτική πληροφορία είναι αρκετά χρήσιμη για την εκπαίδευση και τη δόμηση ενός γλωσσικού μοντέλου.

Επιπλέον, στην εργασία μας ερευνούμε αρκετούς τρόπους για την συνένωση της σημασιολογικής και της συντακτικής πληροφορίας με την πληροφορία συνύπαρξης των διαδοχικών λέξεων. Αυτό επιτυγχάνεται είτε με την συνένωση των μετρικών, είτε με γραμμικό συνδυασμό των μετρικών.

Τα αποτελέσματα μας έδειξαν ότι, κάνοντας χρήση περισσότερης πληροφορίας για την περιγραφή των λέξεων, το μοντέλο μας γίνεται αποδοτικότερο. Τα συνεχή γλωσσικά μοντέλα έχουν εφαρμογή σε διαφόρους τομείς όπως στην αναγνώριση ομιλίας, την αυτόματη διόρθωση και την πρόβλεψη κειμένου και τη διόρθωση υπαγόρευσης.

Λέξεις Κλειδιά: Cosine Similarity, Jaccard Similarity, Overlap coefficient similarity, Dice similarity, Normalized Google Distance, Co-occurrence, Syntax, Gaussian Mixture Model (GMM), Tied Gaussian Mixture Model (TGMM), Expectation Maximization (EM), Singular Value Decomposition (SVD), Linear Discriminant Analysis, Perplexity.

ABSTRACT

The continuous language models are based on the co-occurrence information of consecutive words. In this thesis, we include semantic and syntactic information in the training process of the continuous language models.

The semantic information of words was extracted using similarity metrics, such as: Cosine Similarity, Jaccard Similarity, Dice Coefficient Similarity, Overlap Coefficient Similarity, Normalized Google Distance. The syntactic information was obtained using a part of speech tagger and process to be training and development of a language model.

Moreover, in our thesis we investigate several ways of combining semantic and syntactic information with the co-occurrence of successive words. This is achieved either by concatenation, or through a linear combination of the metrics.

The results show that when adding more information in the training we can obtain more efficient language models. Continuous language models are used in several application domains such as speech recognition, automatic text correction and dictation.

Keywords: Cosine Similarity, Jaccard Similarity, Overlap coefficient similarity, Dice similarity, Normalized Google Distance, Co-occurrence , Syntax, Gaussian Mixture Model (GMM), Tied Gaussian Mixture Model (TGMM), Expectation Maximization (EM), Singular Value Decomposition (SVD), Linear Discriminant Analysis, Perplexity.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΙΝΑΚΑΣ ΣΧΗΜΑΤΩΝ

ΛΙΣΤΑ ΠΙΝΑΚΩΝ

1

Εισαγωγή

1.1 Γλωσσικά Μοντέλα σε Διακριτό Χώρο

Γνωρίζουμε πως τα στατιστικά μοντέλα χρησιμοποιούν στατιστικές τεχνικές για την εκτίμηση τους. Αναθέτουν, δηλαδή, πιθανότητες σε κάθε λέξη του κειμένου που έχουμε προς εκπαίδευση. Μία από τις κυρίαρχες τεχνολογίες είναι τα N-gram μοντέλα. Ένα N-gram μοντέλο θεωρεί κάθε λέξη ως μία διακριτή μεταβλητή. Το μοντέλο αυτό είναι χρήσιμο για πολλές εφαρμογές όπως η αναγνώριση ομιλίας, η οπτική αναγνώριση χαρακτήρων, η αυτόματη μετάφραση, ακόμη και η διόρθωση υπαγόρευσης. Γενικά, το N-gram μοντέλο είναι αποτελεσματικό, όταν υπάρχει ένα ικανοποιητικό σύνολο δεδομένων για μία συγκεκριμένη εργασία.

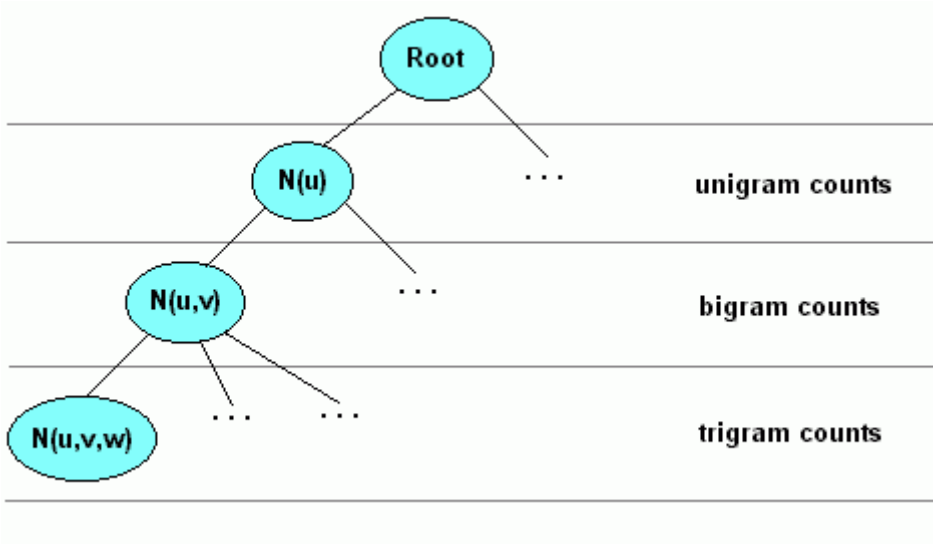
Επιπλέον, γνωρίζουμε ότι, για τη σωστή μοντελοποίηση ενός γλωσσικού μοντέλου με βάση το λεξικό, θα πρέπει για κάθε λέξη να γνωρίζουμε τις προηγούμενες λέξεις που την ακολουθούν.

Τα στατιστικά γλωσσικά μοντέλα που περιγράφουν την πιθανότητα εμφάνισης μίας ακολουθίας λέξεων P (W) για μία ακολουθία λέξεων $W_n = w_1 w_2 \dots w_n$.

Κάνοντας χρήση τον κανόνα αλυσίδας των πιθανοτήτων το P (W) γίνεται:

$$\begin{aligned} P(W) &= P(w_1, w_2, w_3, \dots, w_N) \\ &= P(w_1)P(w_2|w_1)P(w_3|w_1, w_2) \dots P(w_n|w_1, w_2, \dots, w_{n-1}) \\ &= \prod_{i=1}^{n} P(w_i|w_1, w_2, \dots, w_{i-1}) \end{aligned} \qquad 1.1.1$$

Όπου $P(w_i|w_1, w_2, \dots, w_{i-n})$ είναι η πιθανότητα εμφάνισης της λέξης w_i δεδομένο ότι έχει εμφανιστή η ακολουθία $(w_1 w_2 \dots w_{\iota-1})$. Συνεπώς, η επιλογή του w_i εξαρτάται από το ιστορικό εμφάνισης λέξεων. Για ένα λεξιλόγιο μεγέθους V υπάρχουν V_{i-1} διαφορετικές ιστορίες και έτσι για να καθοριστεί πλήρως το $P(w_i|w_1, w_2, \dots, w_{i-n})$ θα πρέπει να εκτιμηθούν V_i τιμές. Στην πραγματικότητα, οι πιθανότητες $P(w_i|w_1, w_2, \dots, w_{i-n})$είναι αδύνατον να εκτιμηθούν ακόμη και για μέτριες τιμές του i, δεδομένου ότι οι περισσότερες ιστορίες είναι μοναδικές ή έχουν εμφανιστεί μόνο μερικές φορές. Μία πρακτική λύση στο παραπάνω πρόβλημα είναι να περιορίσουμε τη μνήμη του Markov.



Σχήμα 1.1 : N-grams

1.2 Σχεδιασμός των γλωσσικών μοντέλων

Ο σκοπός του γλωσσικού μοντέλου είναι να παράγει ένα μηχανισμό για τον υπολογισμό της πιθανότητας εμφάνισης κάποιας λέξης w_k, σε μία έκφραση δεδομένων των προηγουμένων λέξεων, $W_1^{k-1} = [w_k, \dots, w_{k-1}]$. Ένας απλός και αποδοτικός τρόπος για να επιτευχθεί αυτός ο υπολογισμός είναι η χρήση των Μαρκοβιανών N-οστής τάξης (N-grams), στις οποίες θεωρούμε ότι η λέξη w_k εξαρτάται μόνο από τις προηγούμενες N-1 λέξεις.

$$P(w_k|W^{k-1}) = P\left(w_k\middle|W_{k-n-1}^{k-1}\right) \qquad 1.2.1$$

Οι N-grams μπορούν να υπολογιστούν από απλές συχνότητες εμφάνισης και να αποθηκευτούν σε πίνακες, Για παράδειγμα όταν N=3, δηλαδή χρησιμοποιούν συνδυασμούς τριών λέξεων, τότε έχουμε:

$$P(w_k|w_{k-1}, w_{k-2}) = \frac{t(w_{k-2}, w_{k-1}, w_k)}{b(w_{k-2}, w_{k-1})} \qquad 1.2.2$$

όπου $t(w_{k-2}, w_{k-1}, w_k)$ είναι ο αριθμός των εμφανίσεων της ακολουθίας λέξεων w_{k-2}, w_{k-1}, w_k στα δεδομένα εκπαίδευσης και $b(w_{k-2}, w_{k-1})$ είναι ο αριθμός των εμφανίσεων της ακολουθίας λέξεων w_{k-2}, w_{k-1}.

Εύκολα μπορεί να παρατηρήσει κάποιος ότι απαιτείται ένας πολύ μεγάλος αριθμός από τέτοιες ακολουθίες τριών λέξεων, ακόμη και για ένα λεξιλόγιο με 10.000 καταχωρήσεις για να περιγράφει το σύνολο των λέξεων μίας εφαρμογής. Συγκεκριμένα υπάρχουν V^3 ενδεχόμενες ακολουθίες για λεξικό με V λέξεις. Επειδή, όμως, ορισμένες από αυτές δεν έχουν επαρκή αριθμό εμφανίσεων, το αποτέλεσμα της σχέσης (1.2.2) δεν θα μπορεί να θεωρηθεί αξιόπιστο, καθώς είναι δυνατό να υφίσταται οξύ πρόβλημα αραιότητας δεδομένων εκπαίδευσης

Η λύση που εφαρμόζεται στη βιβλιογραφία είναι η εφαρμογή διαφόρων smoothing τεχνικών στις διακριτές κατανομές ώστε να μην μηδενίζουν οι πιθανότητες για της λέξης με μικρή ή μηδενική συχνότητα εμφάνισης στα δείγματα εκπαίδευσης.

1.3 Οργάνωση κειμένου

Στο δεύτερο κεφάλαιο, θα δείξουμε πως γίνεται η ανάπτυξη των συνεχή γλωσσικών μοντέλων. Για την καλύτερη εκπαίδευση των συνεχή γλωσσικών μοντέλων χρησιμοποιούμε κάποιες μετρικές ομοιότητα καθώς και συντακτική πληροφορία που αναλύονται στο τρίτο κεφάλαιο. Στο τέταρτο κεφάλαιο, περιγράφουμε τα πειράματα και εκθέτουμε τα συμπεράσματα μας. Στο πέμπτο κεφάλαιο, συνοψίζουμε λέγοντας τι κάναμε γενικά σε όλη την διατριβή μας και παρουσιάζουμε κάποιες τεχνικές που μπορούν να πραγματοποιηθούν στο μέλλον για τη βελτίωση της έρευνας μας .

2

Συνεχή Γλωσσικά Μοντέλα

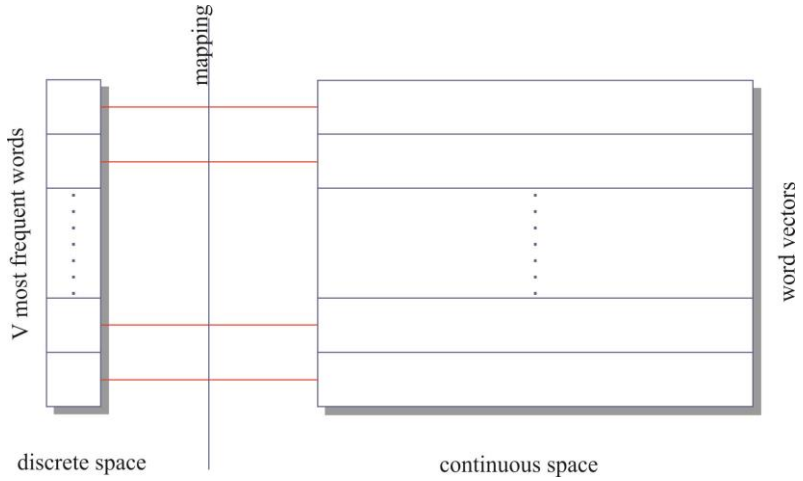
2.1 Γλωσσικά Μοντέλα σε Συνεχή Χώρο

Γνωρίζουμε πως η κυρίαρχη τεχνολογία για την εκπαίδευση ενός γλωσσικού μοντέλου είναι η μέθοδος με N-grams. Όμως αυτή η μέθοδος έχει δύο προβλήματα. Πρώτον, ένα N-gram μοντέλο έχει έναν τεράστιο αριθμό από παραμέτρους και το οποίο είναι δύσκολο να προσαρμοστεί χρησιμοποιώντας ένα σχετικά μικρό όγκο δεδομένων. Δεύτερον, θα έχουμε εμφάνιση πολλών μηδενικών πιθανοτήτων. Όποτε, για το γλωσσικό μοντέλο που θα δημιουργήσουμε, καλό θα ήταν να απαλείψουμε με κάποιο τρόπο τα μηδενικά, ώστε να πάρουμε μικρότερα διανύσματα για την περιγραφή των λέξεων.

Υπάρχουν ορισμένα σημαντικά ζητήματα σχετικά με αυτά τα μοντέλα. Αρχικά, θα πρέπει να κάνουμε κατάλληλη προβολή της κάθε «διακριτής» λέξης στο νέο συνεχή χώρο. Στη συνέχεια, θα πρέπει να ασχοληθούμε με τους χώρους μεγάλων διαστάσεων που θα δημιουργηθούν. Η μοντελοποίηση σε χώρους μεγάλων διαστάσεων η οποία αναφέρεται και ως "κατάρα των διαστάσεων", θεωρείται πολύ δύσκολη. Οπότε, η οικοδόμηση ενός LM συνεχή χώρου είναι η χαρτογράφηση της λέξης από το διακριτό χώρο στο συνεχή, καθώς και έναν ταξινομητή που θα αποφασίζει για την επομένη λέξη με βάση την ιστορία, δηλαδή να εκχωρεί μία πιθανότητα για κάθε λέξη, δεδομένης της ιστορία της.

2.2 Αντικατάσταση Λέξεων Από Συνεχή Διανύσματα

Αντικατάσταση των λέξεων από συνεχή διανύσματα γίνεται έτσι ώστε κάθε λέξη μας να μπορεί να προβληθεί στο νέο συνεχή χώρο. Παρατηρούμε ότι, έχουμε αρκετές λέξεις οι οποίες εμφανίζονται στα δεδομένα που έχουμε για εκπαίδευση, οπότε θα πρέπει με κάποιον τρόπο να τις μειώσουμε. Για να το πετύχουμε αυτό, ορίζουμε το λεξιλόγιο μας με βάση τις λέξεις που έχουν τη μεγαλύτερη συχνότητα εμφάνισης και τις υπόλοιπες τις κατηγοριοποιούμε στην κλάση με το όνομα <unk>. Για τον προσδιορισμό της χαρτογράφησης θα πρέπει να λάβουμε υπόψη μας τη συχνότητα , τη σημασία και τη σχέση κάθε λέξης με τις άλλες λέξεις.



Σχήμα 2.1 : Αντικατάσταση των λέξεων από συνεχή διανύσματα

2.2.1 Χαρακτηριστικά Διανύσματα των Λέξεων

Κάθε λέξη μπορεί αρχικά να εκπροσωπείται από ένα δείκτη - διάνυσμα w_i, έχοντας μονάδα στην i-θέση και μηδενικά στις υπόλοιπες V-1 θέσεις (όπου V είναι το μέγεθος του λεξιλογίου μας). Έτσι έχουμε ένα πινάκα διαστάσεων VxV που περιέχει όλα τα διανύσματα των λέξεων. Αυτός ο πίνακας είναι αρκετά αραιός και έχει την ακόλουθη μορφή :

w_1	1	0	0	0	0	…	0
w_2	0	1	0	0	0	…	0
w_3	0	0	1	0	0	…	0
w_4	0	0	0	1	0	...	0
w_5	0	0	0	0	1	…	0
...	...	...	...	...	...	...	...
w_v	0	0	0	0	0	...	1

Πίνακας 2.1: Δείκτες Διανυσμάτων

Αυτός ο πίνακας αποτελείται από V^2 στοιχειά. Αυτό σημαίνει ότι, καθώς το μέγεθος του λεξιλογίου αυξάνεται , το μέγεθος του πινάκα αυξάνεται εκθετικά. Το επόμενο βήμα είναι η χαρτογράφηση κάθε φορέα σε ένα διάνυσμα μικρότερων διαστάσεων, το οποίο θα αφορά τη συχνότητα εμφάνισης της λέξης και τον τρόπο με τον οποίο κάθε λέξη συμπεριφέρεται με τις άλλες.

2.2.2 Co-occurrence

Όπως αναφέρθηκε πριν, για να γίνει η κατάλληλη χαρτογράφηση της κάθε λέξης, βλέπουμε πως κάθε λέξη σχετίζεται με τις υπόλοιπες. Για αυτό δημιουργούμε έναν πινάκα που θα μας δείχνει τον αριθμό των φορών που μία λέξη ακολουθείται από μία άλλη. Δηλαδή, κάθε στοιχειό του πινάκα περιέχει την τιμή C_{ij} , όπου j είναι η λέξη που ακολουθείται από τη λέξη i, στα δεδομένα εκπαίδευσης.

C_{ij}	V_1	V_2	V_3	V_4	V_5	….	V_v
V_1	0	0	0	92	34	….	57
V_2	10	0	0	0	12	….	0
V_3	0	0	0	37	53	….	156
V_4	12	78	0	0	64	….	0
V_5	0	35	118	0	0	….	24
….	….	….	….	….	….	….	….
V_v	45	56	0	72	0	….	0

Πίνακας 2.2: Πίνακας Co-occurrence

Παρατηρούμε ότι, ο πίνακας έχει διαστάσεις VxV και περιέχει πάρα πολλά μηδενικά, γεγονός το οποίο είναι προφανές, αφού ο πίνακας περιγράφει το πόσο συχνά μία λέξη ακολουθείται από μία άλλη. Για να πετύχουμε τη μείωση διαστάσεων του παραπάνω πινάκα χρησιμοποιούμε τη μέθοδο *Singular Value Decomposition* (SVD). Τη μέθοδο SVD την περιγράφουμε παρακάτω.

2.2.3 Singular Value Decomposition (SVD)

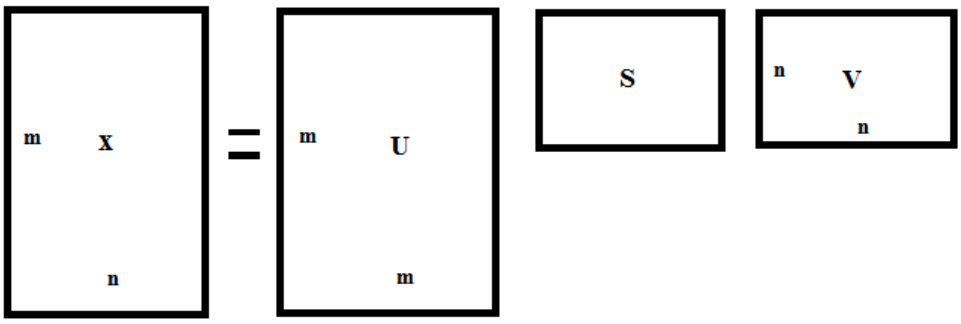
Η μέθοδος SVD είναι μια από τις πιο διαδομένες μεθόδους για τη μείωση των διαστάσεων, είναι ένας από τους ισχυρούς αλγορίθμους της γραμμικής άλγεβρας και έχει εφαρμοστεί σε πολλούς τομείς, όπως είναι η αναγνώριση προτύπων.

Η μέθοδος SVD διασπά τον πινάκα X με διαστάσεις (m x n) σε τρεις $X=USV^T$:

- Ο πίνακας U είναι ένας ορθογώνιος πίνακας (m x m), οι στήλες του οποίου είναι τα αριστερά ιδιοδιανύσματα του πίνακα X. Ισχύει ότι $U^TU = I_n$.

- Ο πίνακας V είναι ένας ορθογώνιος πίνακας (n x n), οι στήλες του οποίου είναι τα δεξιά ιδιοδιανύσματα του πίνακα X. Ισχύει ότι $V^TV = I_n$.

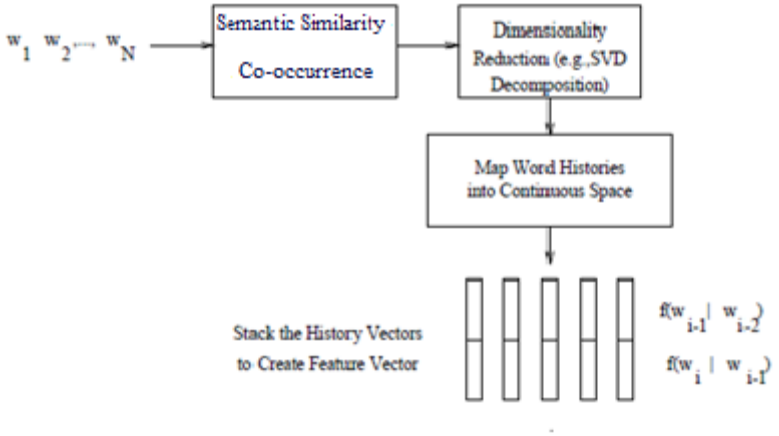
- Ο πίνακας S είναι ένας διαγώνιος πίνακας (m x n), του οποίου τα διαγώνια στοιχειά είναι οι ιδιοτιμές του πίνακα X. Ισχύει ότι $S = \text{diag}(\sigma_1, \ldots, \sigma_n)$ όπου $\sigma_1 > 0$ όταν $1 < i < r$ και $\sigma_i = 0$ όταν $i > r$



Σχήμα 2.2 : Σχηματική απεικόνιση της μέθοδου SVD

2.2.4 Παραγωγή Ιστορικών

Με βάση τα N-gram μοντέλα υπολογίζουμε την ιστορία κάθε λέξης που θα αποτελείται από τις προηγούμενες N-1 λέξεις. Ανάλογα με την τιμή του N , θα έχουμε και το αντίστοιχο μοντέλο. Αν N=1 , τότε κάθε λέξη είναι ανεξάρτητη. Για N=2, κάθε λέξη εξαρτάται από την προηγούμενη της και εάν N=3, κάθε λέξη εξαρτάται από τις προηγούμενες δύο λέξεις. Στη φυσική γλωσσά φαίνεται ότι κάθε λέξη έχει ισχυρή εξάρτηση από τις δύο προηγούμενες της, έτσι ώστε να εκπαιδεύσει trigram μοντέλα, για N=3.



Σχήμα 2.3 : Παραγωγή Ιστορικών

Τα διανύσματα ιστορικών θα τα κάνουμε χρήση στις Gaussian κατανομές που ακλουθούν παρακάτω, όταν θα αναφερθούμε στο συνεχή χώρο. Τα ιστορικά θα αποτελούνται από τα διανύσματα που περιγράφουν τις λέξεις μας και θα έχουν μήκος 100*(N-1)= 200 στοιχεία (N=3 διότι, όπως αναφέραμε παραπάνω στα γλωσσικά μοντέλα χρησιμοποιούμε trigram, το 100 προκύπτει από τις διαστάσεις του SVD με τις οποίες θα πειραματιστούμε). Παρατηρούμε ότι, τα ιστορικά θα προβάλλονται σε ένα τεράστιο διανυσματικό χώρο, οπότε θα πρέπει να βρούμε ένα τρόπο να τα μειώσουμε. Ο τρόπος αυτός είναι να τα προβάλλουμε σε μικρότερο χώρο με τη μέθοδο LDA. Δηλαδή, να πάρουμε ένα νέο διάνυσμα y_i για το οποίο θα ισχύει $y_i = B \cdot h_i$, όπου h_i είναι το διάνυσμα ιστορικών και B ο πίνακας που προκύπτει από τη μέθοδο LDA, η οποία αναλύεται στο επόμενο κεφάλαιο.

2.2.5 Linear Discriminant Analysis (LDA)

Η γραμμική διακινούσα ανάλυση (LDA) είναι μία μέθοδος που στοχεύει στην υψηλή διαχωριστικότητα μεταξύ των διαφόρων λέξεων που θα πρέπει να ταξινομηθούν. Η μέθοδος αυτή ομαδοποιεί τις λέξεις που ανήκουν στην ίδια κλάση, ενώ ταυτόχρονα διαχωρίζει αυτές που ανήκουν σε διαφορετικές κλάσεις. Για να διατυπώσουμε τη μαθηματική διαδικασία βελτίωσης, θα πρέπει να υπολογίσουμε για κάθε κλάση τα διανύσματα των μέσων τιμών και τους πινάκες συνδιακύμανσης

$$\bar{x}_v = \frac{1}{N_v}\sum_{i=1}^{N_v} x_i \qquad 2.2.1$$

$$W_v = \frac{1}{N_v}\sum_{i=1}^{N_v} (x_i - \bar{x}_v)(x_i - \bar{x}_v)^T \qquad 2.2.2$$

και για το ολοκληρωμένο πλήθος δεδομένων θα έχουμε:

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \qquad 2.2.3$$

$$T = \frac{1}{N}\sum_{i=1}^{N} (x_i - \bar{x})(x_i - \bar{x})^T \qquad 2.2.4$$

Στους παραπάνω τύπους, το συνολικό αριθμό των κρατήσεων εκφράζει το N και το N_v αντιπροσωπεύει τον αριθμό των κρατήσεων στην κατηγορία V. Φυσικά, εφόσον υπάρχουν V κλάσεις.

$$\sum_{v=1}^{V} N_v = N \qquad 2.2.5$$

Με αυτούς τους ορισμούς μπορούμε να διατυπώσουμε εύκολα το κριτήριο βελτιστοποίησης, δηλαδή :

$$\hat{B} = argmax_{B^L} \frac{\left|B_L^T \bar{T} B_L\right|}{|B_L^T W B_L|} \quad 2.2.6$$

όπου W,

$$W = \frac{1}{N}\sum_{v=1}^{V} N_v W_v \quad 2.2.7$$

Παρά το γεγονός ότι το κριτήριο αυτό μπορεί να φαίνεται δύσκολο με την πρώτη μάτια, μπορεί να γίνει πολύ εύκολα κατανοητό. Στην ουσία, ο αριθμητής είναι η διακύμανση των συγκεντρωμένων δεδομένων εκπαίδευσης στο νέο μετασχηματισμένο χώρο. Από την άλλη, ο παρανομαστής, είναι η μέση διακύμανση κάθε κατηγορίας στο μετασχηματισμένο χώρο. Το κριτήριο προσπαθεί να μεταποιήσει την απόσταση μεταξύ των κλάσεων, ελαχιστοποιώντας ταυτόχρονα το μέγεθος κάθε μίας κλάσης που αδειάζει. Άρα, καταλήγουμε σε αυτό που θέλουμε, που είναι να μην χάσουμε τη διακριτή πληροφορία των διανυσμάτων στο μετασχηματισμένο χώρο.

Στην περίπτωση μας έχουμε ένα πρόβλημα να αντιμετωπίσουμε και αυτό είναι το μεγάλο μέγεθος των ιστορικών που κάνουμε χρήση. Για να εκτιμηθεί η προβολή του πίνακα B, πρέπει να υπολογιστούν με την μέθοδο LDA οι πίνακες συνδιακύμανσης.

$$t_{1i} = \sum_{n\in i} x_n \quad 2.2.8$$

$$t_{2i} = \sum_{n\in i} x_n x_n^T \quad 2.2.9$$

Όπου i είναι κλάσεις λέξεων που δημιουργήθηκαν και x_n τα διανύσματα ιστορικών.

Αφού υπολογίσαμε όλα τα στατιστικά στοιχειά για όλα τα ιστορικά, πάμε να υπολογίσουμε τα διανύσματα μέσων τιμών για κάθε κλάση λέξεων.

$$m_i = \frac{1}{n_i}\sum_{n\in i} x_n = \frac{1}{n_i} \cdot t_{1i} \qquad 2.2.10$$

Στην συνέχεια, πρέπει να εκτιμηθεί ο πίνακας συνδιακύμανσης για όλο τον πίνακα B

$$S_B = \sum_{i=1}^{c} n_i(m_i - m\,)(m_i - m)^T \qquad 2.2.11$$

Όπου,

$$m = \frac{1}{n}\sum_{i=1}^{c} n_i m_i \qquad 2.2.12$$

Για το συγκεντρωτικό πίνακα συνδιακύμανσης για κάθε κλάση θα πρέπει πρώτα να υπολογίσουμε τους πίνακες συνδιακύμανσης για κάθε κλάση.

$$\begin{aligned} S_i &= \sum (x - m_i\,)(x - m_i)^T = \\ &= \sum \left[xx^T - xm_i^T - m_i x^T + m_i m_i^T\right] = \\ &= \sum xx^T - \ (\sum x)m_i^T - \ m_i \sum x^T + \ n_i m_i m_i^T = \\ &= \sum xx^T - \ n_i m_i m_i^T \end{aligned} \qquad 2.2.13$$

Οπότε έχουμε,

$$S_W = \sum_{i=1}^{c} S_i \qquad 2.2.14$$

Ο στόχος αυτής της μεθόδου είναι να προβάλουμε το διάνυσμα των λέξεων σε ένα νέο χώρο ο οποίος θα σχετίζεται με την ιστορία των λέξεων. Οπότε και θα έχουμε προβάλει το διάνυσμα των λέξεων y στο R^L , όπου L=50.

2.3 Μοντελοποίηση Συνέχων Γλωσσικών Μοντέλων Με GMM

Η μοντελοποίηση των λέξεων στο συνεχή χώρο γίνεται χρησιμοποιώντας μείγματα συνέχων Gaussian κατανομών. Η εκτίμηση των παραμέτρων μας γίνεται με τον αλγόριθμο Expectation-Maximization.

2.3.1 Πολυδιάστατη Gaussian Κατανομή

Στη θεωρία πιθανοτήτων και της στατιστικής, η πολυδιάστατη κανονική κατανομή ή πολυδιάστατη Gaussian κατανομή, είναι μία γενίκευση της μονοδιάστατης κανονικής κατανομής σε υψηλότερες διαστάσεις. Ένας πιθανός ορισμός είναι ότι, ένα τυχαίο διάνυσμα λέγεται ότι είναι p-διαστάσεων κανονικής κατανομής, εάν κάθε γραμμικός συνδυασμός των συνιστωσών p έχει μία μονοδιάστατη κατανομή. Και περιγράφεται από την παρακάτω εξίσωση:

$$f(x) = f(x_1, x_2, \ldots, x_p) = \left(\frac{1}{2\pi}\right)^{\frac{p}{2}} \left[\frac{1}{\det(\Sigma)}\right]^{\frac{1}{2}} \exp\left[\frac{-1}{2}(x-\mu)^t \Sigma^{-1}(x-\mu)\right]$$

Η πολυδιάστατη κανονική κατανομή ενός τυχαίου διανύσματος k-διαστάσεων $X = [X_1, X_2, .., X_k]$ μπορεί να γραφεί και στην ακόλουθη μορφή :

- $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$

με k-διαστάσεων διάνυσμα μεσών τιμών

- $\boldsymbol{\mu} = [\, E[X_1], E[X_2], \ldots, E[X_k] \,]$

και πίνακα συνδιακύμανσης k x k διαστάσεων

- $\boldsymbol{\Sigma} = [\, Cov[X_i, X_j] \,]$, $i = 1,2,\ldots,k$; $j = 1,2,\ldots,k$

Η πολυδιάστατη κανονική κατανομή περιγράφει τις μεταβλητές που τείνουν να συγκεντρώνονται γύρο από τη μέση τιμή τους. Με βάση το πολυδιάστατο κεντρικό οριακό θεώρημα, οποιαδήποτε τυχαία μεταβλητή μπορεί να περιγραφεί από την κανονική κατανομή, εάν έχει ένα μεγάλο σύνολο παρατηρήσεων. Για το λόγο αυτό, είναι χρήσιμες οι Gaussian κατανομές και χρησιμοποιούνται συχνά στη στατιστική μοντελοποίηση και στην εκπαίδευση γλωσσικών μοντέλων.

2.3.2 Γλωσσικό Μοντέλο με Χρήση της Πολυδιάστατης Gaussian Κατανομής

Μετά τη συλλογή των δεδομένων με βάση την ιστορία τους, χρησιμοποιούμε μία πολυδιάστατη Gaussian κατανομή για κάθε λέξη. Στη συνεχεία υπολογίζουμε το διάνυσμα των μεσών τιμών και τον πίνακα συνδιακύμανσης για κάθε λέξη. Η πιθανότητα του y (όπου y το ιστορικό), δεδομένης της λέξης, υπολογίζεται από την παρακάτω εξίσωση:

$P(y|w) = \mathcal{N}(y;\ \mu_w, \Sigma_w)$, όπου μ_w, Σ_w είναι το διάνυσμα μέσων τιμών και ο πίνακας συνδιακύμανσης της λέξης w.

Με τη δημιουργία του μοντέλου μας, θέλουμε να υπολογίσουμε την πιθανότητα της λέξης δεδομένης της ιστορίας της. Αυτό επιτυγχάνεται κάνοντας χρήση τον κανόνα Bays

$$P(w|y) = \frac{P(W)p(y|w)}{p(y)} = \frac{P(w)p(y|w)}{\sum_{u=1}^{V} P(u)p(y|u)} \qquad 2.3.1$$

όπου P(w) είναι η πιθανότητα εμφάνισης της λέξης στο κείμενο μας.

Οι παράμετροι που χρησιμοποιούνται για την εκπαίδευση του μοντέλου μας, είναι ο πίνακας που μας επιστρέφει η μέθοδος SVD, η προβολή του πίνακα B από τη μέθοδο LDA, τα διανύσματα μέσων τιμών και ο πίνακας συνδιακύμανσης για κάθε λέξη. Για την αξιολόγηση του μοντέλου μας, υπολογίζουμε πάνω στα δοκιμαστικά δεδομένα την λογαριθμική πιθανότητα και το perplexity.

2.3.3 Gaussian Mixture Model (GMM)

Ένα Gaussian Mixture Model (GMM) αποτελεί μία παραμετρική συνάρτηση πυκνότητας πιθανότητας, η οποία αντιπροσωπεύεται σαν ένα σταθμισμένο άθροισμα από Gaussian συστατικά. Τα GMMs χρησιμοποιούνται συνήθως σε παραμετρικά μοντέλα σαν την κατανομή των πιθανοτήτων συνέχων μετρήσεων ή για να χαρακτηρίσουν ακουστικά και γλωσσικά μοντέλα. Οι παράμετροι του GMM υπολογίστηκαν από τα δεδομένα εκπαίδευσης και χρησιμοποιήθηκαν στον επαναληπτικό αλγόριθμο Expectation-Maximization(EM) για την παροχή αποτελεσμάτων.

Ένα GMM μπορεί εύκολα να περιγραφεί από την παρακάτω εξίσωση,

$$p(x|\lambda) = \sum_{k=1}^{K} c_k \mathcal{N}\,(\mathrm{x}\,|\mu_k\,, \Sigma_k) \qquad 2.3.2$$

όπου x είναι ένα D-διαστάσεων συνέχων μεταβλητών διάνυσμα δεδομένων , c_k, k=1,..,K, είναι τα βάρη του μείγματος και $\mathcal{N}\,(\mathrm{x}\,|\mu_k\,, \Sigma_k)$, k=1,…,K, είναι οι Gaussian πυκνότητες και περιγράφονται από τον παρακάτω τύπο,

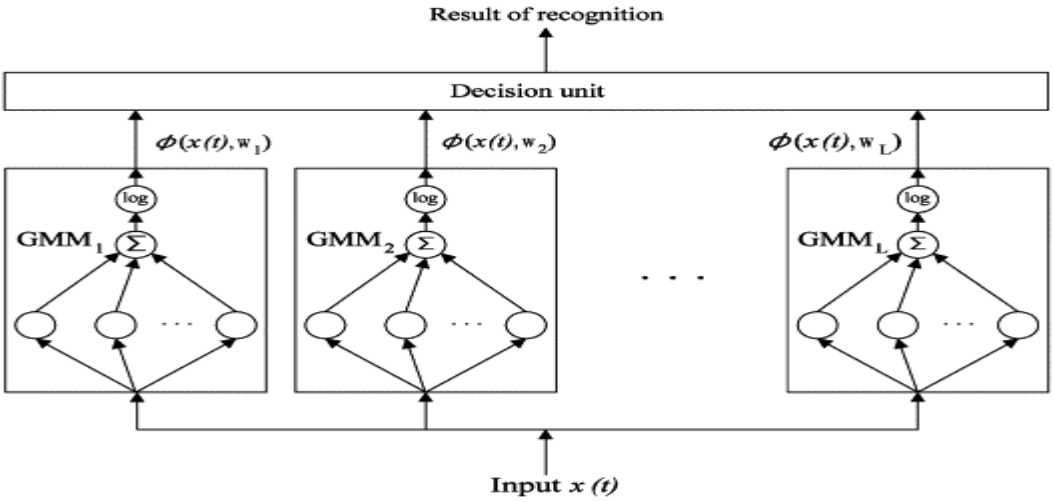
$$\mathcal{N}\,(\mathrm{x}\,|\mu_k\,, \Sigma_k) = \frac{1}{(2\pi)^{\frac{D}{2}}|\Sigma_k|^{\frac{1}{2}}} exp\left\{-\frac{1}{2}\left(\mathrm{x} - \mu_\kappa\right)' \Sigma_k^{-1}\left(\mathrm{x} - \mu_\kappa\right)\right\} \qquad 2.3.3$$

με μέση μ_ι τιμή και πίνακα συνδιακύμανσης Σ_ι. Τα βάρη των μειγμάτων θα πρέπει να ικανοποιούν τη συνθήκη των πιθανοτήτων , δηλαδή το άθροισμα τους να είναι 1. Το πλήρη GMM παραμετροποιείται από το διάνυσμα μέσης τιμής , τον πίνακα συνδιακύμανσης και τα βάρη των μιγμάτων από όλα τα συστατικά των πυκνοτήτων.

Κάθε GMM αποτελείται από K_w κατανομές. Μετά την αντιστοίχιση των λέξεων με την ιστορία τους, συλλέγουμε την ιστορία κάθε λέξης και δημιουργούμε τα μείγματα:

$$p(y|w) = \sum_{k=1}^{K_w} c_{w,k} \mathcal{N}\,(y, \mu_{w,k}, \Sigma_{w,k}) \qquad 2.3.4$$

όπου K_w είναι ο αριθμός των συστατικών.

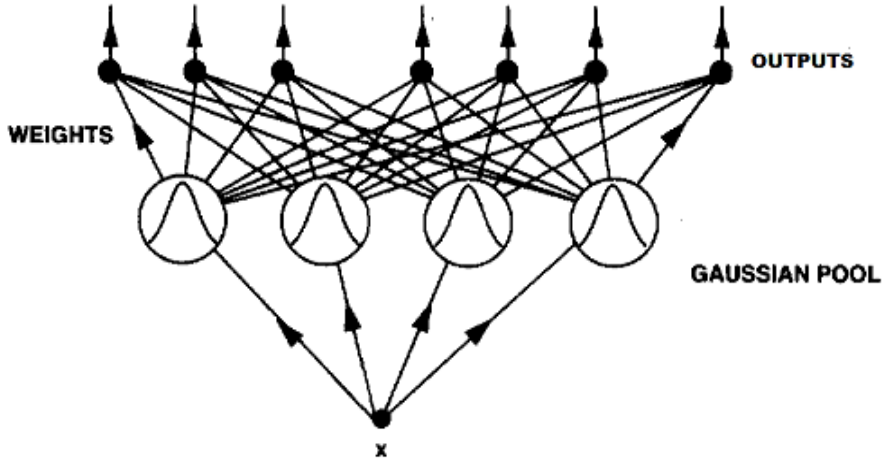


Σχήμα 2.4 : Gaussian Mixture Model

Χρησιμοποιούμε διαφορετικές τιμές του K για κάθε μείγμα στο οποίο θα κάνουμε εφαρμογή για την εκπαίδευση του μοντέλου μας. Το GMLM έχει σαν ορίσματα τις παρακάτω παραμέτρους : τον πίνακα που προκύπτει από την μέθοδο SVD , τον πίνακα B που προκύπτει από την μέθοδο LDA, τους φορείς των μέσων τιμών και τους πίνακες συνδιακύμανσης. Για την εκτίμηση του μοντέλου μας, χρησιμοποιούμε Bayes κανόνες και υπολογίζουμε το perplexity που εξηγούμε παρακάτω.

2.3.4 Tied Gaussian Mixture Model (T-GMM)

Τα GGMs χρησιμοποιούν διαφορετικά σύνολα κατανομών για κάθε μεταβλητή και κάθε ένα είναι παραμετροποιημένο από το διάνυσμα μέσων τιμών και ένα πίνακα συνδιακύμανσης. Αυτό έχει σαν συνέπεια την αύξηση, τη συνεχόμενη αύξηση των μεταβλητών, άρα και των παραμέτρων. Υπάρχει μία ειδίκευση των GGM, αντί να κάνουμε χρήση των διαφορετικών κατανομών για κάθε λέξη, να δημιουργήσουμε ένα κοινό σύνολο των κατανομών, που για κάθε λέξη θα έχει διαφορετικό βάρος.



Σχήμα 2.5 : Gaussian pool

Οι πίνακες συνδιακύμανσης, Σ_i, μπορεί να είναι πλήρους βαθμού ή διαγώνιοι. Επιπλέον, οι παράμετροι μπορεί να είναι κοινοί μεταξύ των Gaussian συστατικών, όπως να έχουν όλα κοινό πίνακα συνδιακύμανσης, τότε θα έχουμε tied GMM. Η επιλογή της διαμόρφωσης του μοντέλου συχνά καθορίζεται από την ποσότητα των διαθέσιμων στοιχείων για την εκτίμηση των παραμέτρων των GMMs και το πώς θα χρησιμοποιηθούν στο γλωσσικό μοντέλο μας. Το σημαντικό με τα GMMs είναι ότι μπορούν να περιγράφουν συστατικά GMMs που έχουν πλήρεις πίνακες διακυμάνσεων από ένα σετ GMMs που μπορούν να έχουν διαγώνιους πίνακες διακυμάνσεων. Γενικά, τα GMM είχαν χρήση σε βιομετρικά συστήματα, αλλά κυρίως σε συνεχή συστήματα αναγνώρισης ομιλιών διότι έχουν την ικανότητα να εκπροσωπούν μεγάλη κατηγορία των κατανομών του δείγματος.

Η συνάρτηση που μας περιγράφει την πιθανότητα της ιστορίας, δεδομένης της λέξης για T-GMM, είναι η ακόλουθη :

$$p\ (y\ |w) = \sum_{k=0}^{K_w} c_{w,k} \mathcal{N}\ (h;\ \mu_{w_k} \Sigma_k) \qquad 2.3.5$$

Τα T-GMM χρησιμοποιούνται στην αναγνώριση προτύπων, σε στατιστικά γλωσσικά μοντέλα και σε ακουστικά μοντέλα. Το πλεονέκτημα τους είναι, ότι για μεγάλο όγκο δεδομένων χρησιμοποιούν όσο το δυνατόν λιγότερες παραμέτρους και η εκπαίδευση του μοντέλου είναι ποιοτικότερη και ταχύτερη.

Το GMLM προτείνεται για να ξεπεραστούν τα μειονεκτήματα της μεθόδου N-gram, όπως είναι η γενίκευση και η προσαρμοστικότητα. Παρά το γεγονός ότι η μέθοδος αυτή έχει ένα μειονέκτημα όσο αφόρα το ποσό των παραμέτρων που χρησιμοποιούνται, είναι αρκετά χρήσιμη. Για να μειώσουμε το πλήθος των παραμέτρων, κάνουμε χρήση των T-GMM, που επιλεγεί να σμίξει τις παραμέτρους, έτσι ώστε να έχουμε καλύτερη εκτίμηση των αποτελεσμάτων με λιγότερες παραμέτρους. Ακόμη, πρέπει να παρατηρήσουμε ότι, ορισμένα στοιχειά της ιστορίας έχουν πολύ μικρές διακυμάνσεις, κάτι το οποίο οδηγεί σε υπολογιστικά προβλήματα. Δεσμεύοντας παραμέτρους όπως τους πίνακες διακύμανσης ή ολόκληρα μίγματα, τότε θα ξεπεράσουμε το παραπάνω πρόβλημα.

Στο δικό μας μοντέλο θα κάνουμε δέσμευση όπως στους πίνακες συνδιακύμανσης, δηλαδή για κάθε λέξη για όλα τα συστατικά των Gaussian, θα έχουμε κοινούς πίνακες συνδιακύμανσης. Όσον αφόρα τις παραμέτρους για την εκπαίδευση του TMLM μοντέλου, αυτές είναι ίδιες με το GMLM.

3

Σημασιολογική και Συντακτική Πληροφορία

3.1 Μετρικές Σημασιολογικής Ομοιότητας

Υπάρχει μία άποψη η οποία υποστηρίζει ότι αν μία λέξη περιγράφεται από το ίδιο περιεχόμενο με μία άλλη λέξη, τότε κατά πασά πιθανότητα θα έχει και την ίδια σημασιολογία με την άλλη λέξη. Πάνω σε αυτή την θεωρία στηρίζονται σχεδόν όλες οι μετρικές που θα αναλύσουμε παρακάτω. Οι μετρικές που θα χρησιμοποιήσουμε για τον υπολογισμό της σημασιολογίας των λέξεων χρησιμοποιούν σύνολα με λέξεις κλειδιά για των υπολογισμό τους.

Για των υπολογισμό των συνόλων με τις λέξεις κλειδιά χρησιμοποιήσαμε ένα φράγμα στην τιμή που μας δίνει την πληροφορία συνύπαρξης διαδοχικών λέξεων (Co-occurrence). Στην ουσία δημιουργήσαμε κλάσεις που περιείχαν για κάθε λέξη τις λέξεις που την ακολουθούσαν τις περισσότερες φορές. Έτσι δημιουργήσαμε κλάσεις που τις ονομάσαμε $K|w|$ και περιείχαν τις λέξεις κλειδιά για κάθε λέξη. Με βάσει το παρακάτω παράδειγμα είναι εύκολο να κατανοήσουμε πως προκύπτουν οι κλάσεις με τις λέξεις κλειδιά.

<s> doublequote did we sell dollars questionmark </s>

<s> what dollars questionmark doublequote barks mr hariri upon his return to the room **period** </s>

<s> he glares at mr razian **comma** chairman of the billionaires two lebanese banks **period** </s>

<s> the aide stares at the floor **period** </s>

<s> doublequote the currency has stabilized because i became prime minister **comma** doublequote mr hariri snaps **period** </s>

<s> doublequote theres no other reason **period** doublequote </s>

<s> credit the man **comma** his money **comma** or both **period** </s>

<s> rafic hariri has lifted lebanon **comma** if not to its feet **comma** at least to its knees **period** </s>

<s> the civil war between muslims and christians ended in nineteen ninety **comma** thanks to forty thousand syrian troops who still havent gone home **period** </s>

<s> mr hariri has launched a ten hyphen year **comma** twelve billion dollars reconstruction program **comma** which is attracting foreign investment and aid **period** </s>

<s> last year **comma** the capital inflow helped the economy grow seven percent as the inflation rate fell below ten percent for the first time in a decade **period** </s>

<s> the work ahead remains enormous **period** </s>

Παραπάνω βρίσκονται οι προτάσεις που έχουν παρθεί από τα δεδομένα μας. Με έντονο χρώμα είναι οι λέξεις που θα χρησιμοποιήσουμε για να βρούμε τις λέξεις κλειδιά. Αρά w_1 είναι η λέξη commaκαι w_2είναι η λέξη period.

Οι λέξεις κλειδιά K|w_1| για την λέξη w_1είναι : razian, minister, lebanon, feet, ninety, year, year.

Οι λέξεις κλειδιά K|w_2| για την λέξη w_2είναι : room, banks, floor, snaps, reason, both, knees, home, aid, decade, enormous.

3.1.1 Cosine Similarity

Η cosine similarity μετρική βασίζεται κατά κύριο λόγο στο περιεχόμενο των λέξεων. Για παράδειγμα, αν έχουμε δύο λέξεις για τις οποίες γνωρίζουμε από ποιες λέξεις εμφανίζονται πιο συχνά, τότε με τον παρακάτω τύπο μπορούμε να υπολογίσουμε στατιστικά πόσο μοιάζουν μεταξύ τους.

$$C(w_1, w_2) = \frac{\left|K|w_1| \cap K|w_2|\right|}{sqrt\left(\left|K|w_1|\right|\right) x\ sqrt\left(\left|K|w_2|\right|\right)} \qquad 3.1.1$$

- Όπου w_1, w_2 είναι οι λέξεις που θέλουμε να ελέγξουμε την ομοιότητά τους
- $K|w_1|, K|w_2|$ είναι κλάσεις με τις λέξεις που εμφανίζονται πιο συχνά ώστε να δηλώνουν το περιεχόμενο των λέξεων που θα συγκρίνουμε.

Παρατηρούμε ότι, η cosine similarity ελέγχει κατά ποσό τα διανύσματα τα οποία περιέχουν το περιεχόμενο κάθε λέξης, συγκλίνουν έτσι ώστε να μας δώσει σαν αποτέλεσμα την ομοιότητα τους.

Για παράδειγμα έχουμε τις λέξεις comma και period. Η λέξη comma έχει για λέξεις κλειδιά τις : lebanon, feet, year, year. Ενώ η λέξη period έχει για λέξεις κλειδιά τις : home, aid, feet.

	lebanon	feet	year	home	aid
$K\|w_1\|$	1	1	2	0	0
$K\|w_2\|$	0	1	0	1	1

$$C(w_1, w_2) = \frac{1*0+1*1+2*0+1*0+1*0+1*0}{\sqrt{1^2+1^2+2^2+0^2+0^2}\sqrt{0^2+1^2+0^2+1^2+1^2}}$$
$$= \frac{1}{3\sqrt{2}} = 0.236$$

Cos_{ij}	V_1	V_2	V_3	V_4	V_5	….	V_v
V_1	1	0	0	0	0	….	0
V_2	0	1	0	0	0	….	0.030
V_3	0	0	1	0	0.523	….	0
V_4	0.321	0	0	0	1	….	0
V_5	0	0	1	0	1	….	0
….	….	….	….	….	….	….	….
V_v	0	0.600	0	0	0	….	1

Πίνακας 3.1: Πίνακας Cosine Similarity

Στον παραπάνω πινάκα παρατηρούμε τις αποστάσεις που υπάρχουν ανάμεσα στις λέξεις. Όταν η απόσταση ισούται με μονάδα, τότε η πληροφορία που μας δίνεται είναι πως οι δύο λέξεις θα έχουν και την ίδια σημασία. Γενικά είναι εύκολο να παρατηρηθεί ότι, οι πίνακες είναι αρκετά αραιοί και περιέχονται σε αυτούς παρά πολλά μηδενικά. Τα στοιχειά που περιέχει κάθε πίνακας εξαρτώνται από το λεξικό το οποίο κάνουμε χρήση για την εκπαίδευση του μοντέλου. Στη δίκη μας περίπτωση ο πίνακας είναι διαστάσεων 2700x2700.

3.1.2 Jaccard Similarity

Ομοίως, η Jaccard Similarity μετρική βασίζεται κατά κύριο λόγο στο περιεχόμενο των λέξεων. Μονό που η Jaccard δεν έχει τόσο αυστηρό ορισμό. Ουσιαστικά, όπως παρατηρείτε και από τον παρακάτω τύπο μας δίνει καθαρά ποσοστιαία απόδοση.

$$\mathrm{J}(w_1, w_2) = \frac{\left|K|w_1| \cap\ K|w_2|\right|}{\left|K|w_1|\right| + \left|K|w_2|\right| - \left|K|w_1| \cap\ K|w_2|\right|} \qquad 3.1.2$$

- Όπου w_1, w_2 είναι οι λέξεις που θέλουμε να ελέγξουμε την ομοιότητα τους
- $K|w_1|, K|w_2|$ είναι κλάσεις με τις λέξεις που εμφανίζονται πιο συχνά, ώστε να δηλώνουν το περιεχόμενο των λέξεων που θα συγκρίνουμε.

Για παράδειγμα έχουμε τις λέξεις comma και period. Η λέξη comma έχει για λέξεις κλειδιά τις : lebanon, feet, year, year. Ενώ η λέξη period έχει για λέξεις κλειδιά τις : home, aid, feet.

	lebanon	feet	year	home	aid
$K\|w_1\|$	1	1	2	0	0
$K\|w_2\|$	0	1	0	1	1

$$J(w_1, w_2) =$$
$$= \frac{1*0+1*1+2*0+1*0+1*0}{(1+1+2)+(1+1+1)-(1*0+1*1+2*0+1*0+1*0)}$$
$$= \frac{1}{6} = 0.167$$

J_{ij}	V_1	V_2	V_3	V_4	V_5	….	V_v
V_1	1	0	0	0	0.400	….	0
V_2	0	1	0	0	0	….	0.730
V_3	0	0	1	0	0	….	0
V_4	0	0.789	0	0	1	….	0
V_5	0	0	1	0	1	….	0
….	….	….	….	….	….	….	….
V_v	0	0	0	0	0	….	1

Πίνακας 3.2: Πίνακας Jaccard Similarity

Στον παραπάνω πινάκα παρατηρούμε τις αποστάσεις που υπάρχουν ανάμεσα στις λέξεις. Όταν η απόσταση ισούται με μονάδα, τότε η πληροφορία που μας δίνεται είναι πως οι δύο λέξεις θα έχουν και την ίδια σημασία. Γενικά, εύκολα παρατηρούμε ότι, οι πίνακες είναι αρκετά αραιοί και περιέχονται σε αυτούς παρά πολλά μηδενικά. Τα στοιχειά που περιέχει κάθε πίνακας εξαρτώνται από το λεξικό το οποίο κάνουμε χρήση για την εκπαίδευση του μοντέλου. Στη δίκη μας περίπτωση ο πίνακας είναι διαστάσεων 2700x2700

3.1.3 Dice Coefficient Similarity

Όσον αφόρα τη dice coefficient similarity, ισχύει ακριβώς το ίδιο με τις παραπάνω μετρικές, δηλαδή βασίζεται κατά κύριο λόγο στο περιεχόμενο των λέξεων. Παρατηρώντας και τον παρακάτω τύπο θα καταλάβουμε πως αυτή η μετρική δίνει βάρος καθαρά στην τομή την οποία θα έχουν τα διανύσματα με τα περιεχόμενα των λέξεων.

$$\mathrm{D}(w_1, w_2) = 2\frac{\left|K|w_1| \cap K|w_2|\right|}{\left|K|w_1|\right| + \left|K|w_2|\right|} \qquad 3.1.3$$

- Όπου w_1, w_2 είναι οι λέξεις που θέλουμε να ελέγξουμε την ομοιότητα τους
- $K|w_1|, K|w_2|$ είναι κλάσεις με τις λέξεις που εμφανίζονται πιο συχνά, ώστε να δηλώνουν το περιεχόμενο των λέξεων που θα συγκρίνουμε.

Για παράδειγμα έχουμε τις λέξεις comma και period. Η λέξη comma έχει για λέξεις κλειδιά τις : lebanon, feet, year, year. Ενώ η λέξη period έχει για λέξεις κλειδιά τις : home, aid, feet.

	lebanon	feet	year	home	aid
$K\|w_1\|$	1	1	2	0	0
$K\|w_2\|$	0	1	0	1	1

$$D(w_1, w_2) = 2\frac{1*0 + 1*1 + 2*0 + 1*0 + 1*0}{(1+1+2)+(1+1+1)} = \frac{2}{7} = 0.286$$

D_{ij}	V_1	V_2	V_3	V_4	V_5	….	V_v
V_1	1	0	0	0	0.001	….	0
V_2	0.234	1	0	0	0	….	0.030
V_3	0	0	1	0	0.768	….	0
V_4	0.123	0.789	0	0	1	….	0
V_5	0	0	1	0	1	….	0
….	….	….	….	….	….	….	….
V_v	0	0.609	0	0	0	….	1

Πίνακας 3.3: Πίνακας Dice Coefficient Similarity

Στον παραπάνω πινάκα παρατηρούμε τις αποστάσεις που υπάρχουν ανάμεσα στις λέξεις. Όταν η απόσταση ισούται με μονάδα, τότε η πληροφορία που μας δίνεται είναι πως οι δύο λέξεις θα έχουν και την ίδια σημασία. Γενικά είναι εύκολο να παρατηρηθεί ότι, οι πίνακες είναι αρκετά αραιοί και περιέχονται σε αυτούς παρά πολλά μηδενικά. Τα στοιχειά που περιέχει κάθε πίνακας εξαρτώνται από το λεξικό το οποίο κάνουμε χρήση για την εκπαίδευση του μοντέλου. Στη δίκη μας περίπτωση ο πίνακας είναι διαστάσεων 2700x2700.

3.1.4 Overlap Coefficient Similarity

Και σε αυτή την περίπτωση η βασική ιδέα είναι ίδια με αυτή που αναφέρθηκε παραπάνω, μονό που εδώ έχουμε μία άλλη προσέγγιση. Ουσιαστικά, πάλι δίνουμε βαρύτητα στην τομή, αλλά τώρα κοιτάμε μονό το μεγαλύτερο διάνυσμα. Παρακάτω παρατηρούμε τον τύπο που περιγράφει την Overlap coefficient similarity.

$$O(w_1, w_2) = \frac{\left|K|w_1| \cap K|w_2|\right|}{\min(\left|K|w_1|\right|, \left|K|w_2|\right|)} \qquad 3.1.4$$

- όπου w_1, w_2 είναι οι λέξεις που θέλουμε να ελέγξουμε την ομοιότητα τους

- $K|w_1|, K|w_2|$ είναι κλάσεις με τις λέξεις που εμφανίζονται πιο συχνά ώστε να δηλώνουν το περιεχόμενο των λέξεων που θα συγκρίνουμε.

Για παράδειγμα έχουμε τις λέξεις comma και period. Η λέξη comma έχει για λέξεις κλειδιά τις : lebanon, feet, year, year. Ενώ η λέξη period έχει για λέξεις κλειδιά τις : home, aid, feet.

	lebanon	feet	year	home	aid
$K\|w_1\|$	1	1	2	0	0
$K\|w_2\|$	0	1	0	1	1

$$O(w_1, w_2) = \frac{1*0+1*1+2*0+1*0+1*0}{(1+1+1)} = \frac{1}{3} = 0.333$$

O_{ij}	V_1	V_2	V_3	V_4	V_5	….	V_v
V_1	1	0	0	0	0.001	….	0
V_2	0.500	1	0	0	0	….	0.060
V_3	0	0	1	0	0.012	….	0
V_4	0	0	0	0	1	….	0
V_5	0	0	1	0	1	….	0
….	….	….	….	….	….	….	….
V_v	0	0.300	0	0	0	….	1

Πίνακας 3.4: Πίνακας Overlap Coefficient Similarity

Στον παραπάνω πινάκα παρατηρούμε τις αποστάσεις που υπάρχουν ανάμεσα στις λέξεις. Όταν η απόσταση ισούται με μονάδα, τότε η πληροφορία που μας δίνεται είναι πως οι δύο λέξεις θα έχουν και την ίδια σημασία. Γενικά είναι εύκολο να παρατηρηθεί ότι, οι πίνακες είναι αρκετά αραιοί και περιέχονται σε αυτούς παρά πολλά μηδενικά. Τα στοιχειά που περιέχει κάθε πίνακας εξαρτώνται από το λεξικό το οποίο κάνουμε χρήση για την εκπαίδευση του μοντέλου. Στη δίκη μας περίπτωση ο πίνακας είναι διαστάσεων 2700x2700.

3.1.5 Normalized Google Distance (NGD)

Η normalized Google distance είναι μια μετρική η οποία βασίζεται στη συχνότητα εμφάνισης των λέξεων. Πιο αναλυτικά, η μετρική αυτή μας δείχνει την ομοιότητα ανάμεσα σε δύο λέξεις, υπολογίζοντας το πόσο συχνά εμφανίζεται η μία λέξη μαζί με την άλλη λέξη μέσα στην ίδια πρόταση (για τα δικά μας dataset). Η normalized Google Distance περιγράφεται από τον παρακάτω τύπο και έχει προσαρμοστεί για τα δικά μας δεδομένα.

$$G(w_1, w_2) = \frac{\max\{\log f(w_1), logf(w_2)\} - \log f(w_1, w_2)}{\text{logM} - \min\{\log f(w_1), logf(w_2)\}} \qquad 3.1.5$$

- Όπου $f(w_1), f(w_2)$ είναι η συχνότητα εμφάνισης των λέξεων w_1, w_2 αντίστοιχα

- $f(w_1, w_2)$ είναι η συχνότητα εμφάνισης w_1, w_2 μαζί στην ίδια πρόταση.

- ***M*** είναι ο αριθμός των snippet που έχει η Google στη διάθεση της .Για τα δικά μας δεδομένα είναι 1000000.

NGD_{ij}	V_1	V_2	V_3	V_4	V_5	….	V_v
V_1	1	0.436	0.323	0.792	0.001	….	0
V_2	0.234	1	0	0	0.980	….	0.030
V_3	0	0	1	0.457	0.768	….	0
V_4	0.123	0.789	0	1	0	….	0
V_5	0	0.678	1	0	1	….	0.321
….	….	….	….	….	….	….	….
V_v	0	0.609	0	0	0	….	1

Πίνακας 3.5: Πίνακας Normalized Google Distance

Στον παραπάνω πινάκα παρατηρούμε τις αποστάσεις που υπάρχουν ανάμεσα στις λέξεις. Όταν η απόσταση ισούται με μονάδα, τότε η πληροφορία που μας δίνεται είναι πως οι δύο λέξεις θα έχουν και την ίδια σημασία. Γενικά, εύκολα παρατηρούμε ότι, οι πίνακες είναι αρκετά αραιοί και περιέχονται σε αυτούς παρά πολλά μηδενικά. Τα στοιχειά που περιέχει κάθε πίνακας εξαρτώνται από το λεξικό το οποίο κάνουμε χρήση για την εκπαίδευση του μοντέλου. Στη δίκη μας περίπτωση ο πίνακας είναι διαστάσεων 2700x2700.

Είναι προφανές ότι, ο πίνακας ο οποίος προκύπτει από τη Normalized Google Distance περιέχει και τα λιγότερα μηδενικά. Στην ενότητα των πειραμάτων περιγράφεται αναλυτικά η χρήση των παραπάνω πινάκων.

3.2 Συντακτική Ανάλυση

Για την πλήρη περιγραφή μίας λέξης το καλύτερο που μπορούμε να κάνουμε είναι να ορίσουμε και το συντακτικό τύπο της. Για να βρούμε το συντακτικό τύπο μίας λέξης υπάρχουν πολλοί κανόνες οι οποίοι θα πρέπει να τεθούν. Σημαντικό ρολό παίζει η θέση στην οποία βρίσκεται η λέξη. Πιο αναλυτικά, πρέπει να δούμε όλη τη λέξη και να κάνουμε το συντακτικό της δέντρο. Το συντακτικό της δέντρο είναι η πλήρης περιγραφή μίας πρότασης συντακτικά. Υπάρχουν δύο τρόποι για την ανάγνωση και την υλοποίηση του συντακτικού δέντρου. Ο πρώτος τρόπος είναι να ξεκινήσουμε από πάνω προς τα κάτω. Δηλαδή, να πάρουμε αρχικά την πρόταση-έννοια, να εξετάσουμε σε τι φράσεις χωρίζεται και στην συνέχεια να εξετάσουμε μέσα στις φράσεις ποια μέρη του λογού υπάρχουν. Ο δεύτερος τρόπος είναι να ξεκινήσουμε από τα μέρη του λόγου, δηλαδή κάθε λέξη ποιον συντακτικό τύπο έχει

και να πάμε προς τα πάνω για να εξετάσουμε σε ποιες φράσεις περιέχονται και έτσι να καταλήξουμε στην πρόταση.

Η γραμματική επισημείωση σε ένα κείμενο είναι η διαδικασία κατά την οποία αναγνωρίζεται κάθε λεκτική μονάδα του κειμένου ως μέρος του λόγου και σημειώνεται η ιδιότητα αυτή δίπλα στη λεκτική μονάδα. Έτσι, η έξοδος ενός προγράμματος που υλοποιεί αυτή τη διαδικασία (Part of speech tagger) είναι οι λεκτικές μονάδες του κειμένου και από ένα tag για την κάθε μία που προσδιορίζει αυτή την ιδιότητά της. Με τον όρο tag γίνεται αναφορά σε μία ακολουθία χαρακτήρων που συμβολίζουν τα μέρη του λόγου. Υπάρχουν πολλά διαφορετικά συστήματα ορισμού των μερών του λόγου, άρα και συμβολισμού τους. Για παράδειγμα, υπάρχουν διαφορετικοί τρόποι αντιμετώπισης σημείων στίξης, ειδικών λέξεων κλπ. Πρέπει να σημειωθεί ότι τα tags δεν προσδιορίζουν μέρος του λόγου όπως ο όρος αυτός αναφέρεται ως όρος της γλωσσολογίας. Αποσκοπούν σε μία πιο γενική αναγνώριση του συντακτικού και γραμματικού ρόλου που έχει η κάθε λεκτική μονάδα στην πρόταση, ώστε να είναι πιο εύκολα διαχειρίσιμη για περεταίρω ανάλυση του κειμένου.

Η γραμματική επισημείωση χρησιμοποιείται συνήθως ως ένα πρώτο επίπεδο επεξεργασίας κειμένου, το οποίο θα χρησιμεύσει σε πιο σύνθετες εργασίες που ανήκουν στον τομέα της επεξεργασίας φυσικής γλώσσας, όπως αναλυτικότερη συντακτική ανάλυση, σημασιολογική ανάλυση, μεταφράσεις κλπ.. Ένας σημαντικός παράγοντας που συνέβαλε στην πρόοδο για την γραμματική επισημείωση και στη δημιουργία αντίστοιχων εργαλείων επισημείωσης με αρκετά μεγάλη ακρίβεια, είναι η ύπαρξη μεγάλων σωμάτων δεδομένων, κάθε ένα από τα οποία ονομάζεται επισημειωμένο σύνολο δεδομένων (tagged corpus), δηλαδή μεγάλου όγκου κειμένων κάθε λέξη των οποίων ακολουθείται από μία ετικέτα (tag) γραμματικής επισημείωσης. Η δημιουργία αυτών των σωμάτων έγινε κυρίως με μη αυτοματοποιημένες διαδικασίες με συμβολή ειδικών στον τομέα της γλωσσολογίας και με περιορισμένη χρήση κανόνων από στατικούς αυτόματους συντακτικούς αναλυτές.

Μέχρι σήμερα, η ακρίβεια των εργαλείων επισημείωσης έχει φτάσει μέχρι και το 96%, ποσοστό που αν και είναι αρκετά ικανοποιητικό, μπορεί να δημιουργήσει προβλήματα σε δεύτερο και τρίτο επίπεδο επεξεργασίας μετά το τέλος της επισημείωσης. Η δυσκολία επίτευξης μεγάλης ακρίβειας στη γραμματική επισημείωση οφείλεται σε δύο ανασταλτικούς παράγοντες. Ο πρώτος έχει να κάνει με την ύπαρξη λέξεων με διφορούμενα νοήματα. Τέτοιες λέξεις που μπορούν να έχουν παραπάνω από μία σημασιολογικές ερμηνείες είναι πολύ δύσκολο να αναγνωριστούν από τον υπολογιστή, καθώς θα μπορούσαν να εμφανιστούν με διαφορετικά tags σε διαφορετικές προτάσεις. Για τον άνθρωπο, γνωρίζοντας τη σημασιολογία της εκάστοτε λέξης αλλά και τα συμφραζόμενα της, η επιλογή της σωστής ετικέτας είναι αρκετά πιο εύκολη.

Γενικά, όμως, η αναλυτική χρήση της συντακτικής ανάλυσης δεν μπορεί να μας βοηθήσει, διότι έχουμε να διαχειριστούμε μεγάλο όγκο δεδομένων και πολύ μεγάλο λεξιλόγιο για την εκπαίδευση του γλωσσικού μοντέλου μας. Οπότε, στο λεξιλόγιο μας διπλά από τη λέξη απλά βάλαμε και το συντακτικό της τύπο. Επίσης, κάναμε

έναν πίνακα ο οποίος περιέχει όλους τους συντακτικούς τύπους. Αυτό το κάναμε για να αντιστοιχίσουμε το συντακτικό τύπο με αριθμό, έτσι ώστε να μπορούμε να τον κάνουμε χρήση στον κορμό τον οποίο έχουμε δημιουργήσει. Οι πινάκες μας λοιπόν έχουν την παρακάτω μορφή :

Λεξιλόγιο με συντακτικό τύπο

over	IN
time	NN
into	IN
dollar	NN
share	NN
shares	NNS
could	MD
can	MD
sixty	NN
such	JJ

Πινάκας Συντακτικών Τύπων

PRP$	1
VBG	2
VBD	3
VBN	4
VBP	5
WDT	6
JJ	7
WP	8
VBZ	9
DT	10
NN	11
TO	12
PRP	13
RB	14
NNS	15
LS	16
VB	17
WRB	18
CC	19
CD	20
EX	21
IN	22
MD	23
JJS	24
JJR	25

POS	Description
PRP$	Possessive pronoun
VBG	Verb, gerund or present participle
VBD	Verb, past tense
VBN	Verb, past participle
VBP	Verb, present tense, other 3rd person singular
WDT	Wh-determiner
JJ	Adjective
WP	Wh-pronoun
VBZ	Verb, present tense, other 3rd person singular
DT	Determiner
NN	Noun, singular or mass
TO	
PRP	Personal pronoun
RB	Adverb
NNS	Noun plural
LS	List item maker
VB	Verb
WRB	Wh-verb
CC	Coordinating conjunction
CD	Cardinal number
EX	Existential
IN	Preposition or subordinating conjunction
MD	Modal
JJS	Adjective, superlative
JJR	Adjective, comparative

<u>Λεξιλόγιο και συντακτικός τύπος με βάση τον παραπάνω πίνακα</u>

over 22
time 11
into 22
dollar 11
share 11
shares 15
could 23
can 23
sixty 11
such 7

Για να βρούμε το συντακτικό τύπο κάθε λέξης, κάναμε χρήση την έτοιμη βιβλιοθήκη NLTK. Ουσιαστικά, δώσαμε σαν όρισμα το λεξιλόγιο το οποίο χρησιμοποιούμε για την εκπαίδευση του μοντέλου μας και αυτό μας επέστρεψε ένα πίνακα όπως είναι ο πρώτος, που μας έδινε τη λέξη και το συντακτικό τύπο.

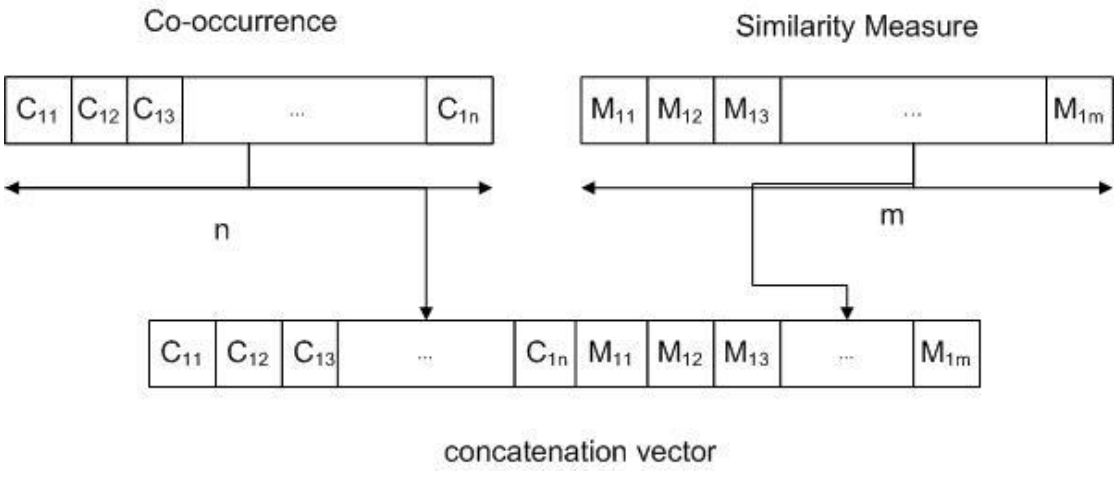
3.3 Μίξη Διανυσμάτων

Εφόσον πλέον έχουμε δημιουργήσει διανύσματα περιγραφής των λέξεων που βασίζονται σε μετρικές ομοιότητας, είναι εύκολο να δημιουργήσουμε με κάποιους τρόπους και διανύσματα που θα περιγραφούν τη λέξη μας με βάση την πληροφορία συνύπαρξης διαδοχικών λέξεων (Co-occurrence), τη σημασιολογική πληροφορία, καθώς και τη συντακτική πληροφορία.

Υπάρχουν δύο τρόποι με τους οποίους μπορεί να επιτευχθεί η μίξη των διανυσμάτων. Ο πρώτος τρόπος είναι να κάνουμε συνένωση των διανυσμάτων διαδοχικά, ενώ ο δεύτερος τρόπος είναι να πάρουμε το γραμμικό συνδυασμό των διανυσμάτων που θέλουμε.

3.3.1 Συνένωση Διανυσμάτων

Αρχικά, βάσει των μετρικών που έχουμε για την περιγραφή των λέξεων μας, δημιουργούμε διανύσματα που περιγράφουν τη λέξη μας. Τα διανύσματα θα έχουν το ίδιο μέγεθος, το οποίο και θα εξαρτάται από το μέγεθος του λεξιλογίου μας. Στον παρακάτω σχήμα μπορεί εύκολα να γίνει κατανοητό το πώς γίνεται η συνένωση των διανυσμάτων.



Σχήμα 3.1 : Συνένωση Διανυσμάτων

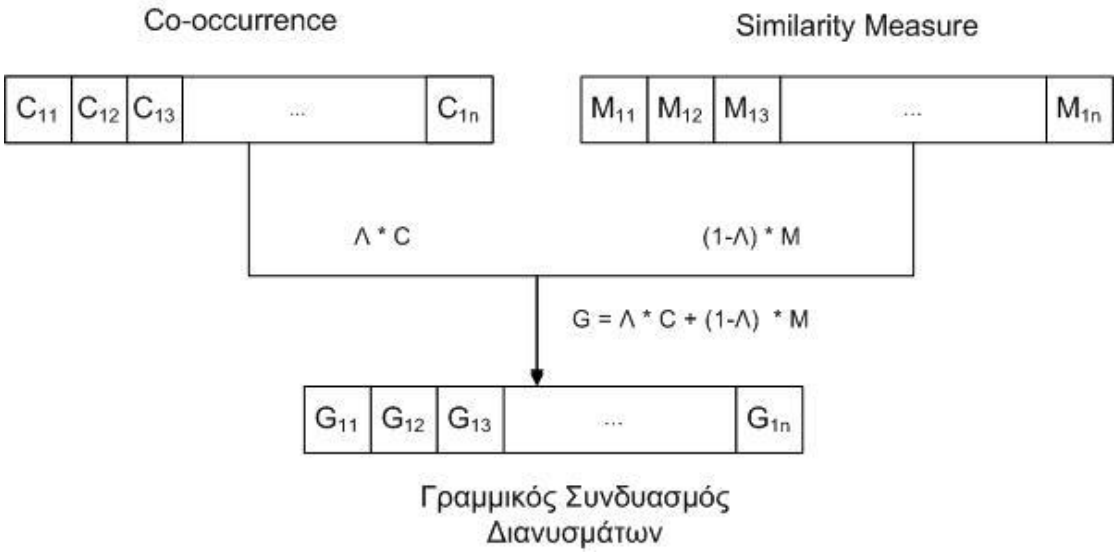
Στην ουσία δημιουργούμε ένα νέο διάνυσμα το οποίο θα έχει το διπλάσιο μέγεθος από τα αρχικά διανύσματα. Θα περιέχει όλη την πληροφορία των αρχικών διανυσμάτων. Πλέον η λέξη μας θα περιγράφεται από τη χρήση δύο μετρικών. Έτσι δεν θα έχουμε λέξεις που θα περιγράφονται από μηδενικά διανύσματα. Αυτό θα μας δώσει σαν αποτέλεσμα το μοντέλο μας να εκπαιδεύεται καλυτέρα. Στο κεφάλαιο των πειραμάτων περιγράφονται αναλυτικά όλοι οι παράμετροι που χρησιμοποιούμε.

3.3.2 Γραμμικός Συνδυασμός Διανυσμάτων

Με τη συνένωση των διανυσμάτων το νέο διάνυσμα θα περιέχει ίδια πληροφορία και για τις δύο μετρικές. Εμείς όμως θέλουμε να εξετάσουμε αν κάποια μετρική έχει μεγαλύτερο βάρος κατά τη δημιουργία του μεικτού διανύσματος, κάτι το οποίο θα έχει σαν αποτέλεσμα το μοντέλο μας να αποδίδει καλυτέρα. Στην ουσία το νέο διάνυσμα περιγράφεται από τον παρακάτω τύπο.

$$G = \Lambda * C + (1 - \Lambda) * M \qquad 3.3.1$$

Όπου C είναι η πληροφορία που παίρνουμε από τον Co-occurrence , M είναι μία μετρική ομοιότητας από αυτές που αναφέραμε και περιγράψαμε παραπάνω και Λ είναι το βάρος το οποίο δίνουμε στην κάθε μετρική. Στο παρακάτω σχήμα περιγράφεται αναλυτικά η διαδικασία δημιουργίας του νέου διανύσματος.



Σχήμα 3.2 : Γραμμικός Συνδυασμός Διανυσμάτων

4

Πειράματα & Αποτελέσματα

Στο κεφάλαιο αυτό θα περιγράψουμε αναλυτικά τα πειράματα τα οποία κάναμε και θα δείξουμε όλες τις παραμέτρους και όλες τις μεθόδους με τις όποιες πειραματιστήκαμε. Τέλος, θα εκθέσουμε τα συμπεράσματα μας, ανάλογα με τα αποτελέσματα που πήραμε.

4.1 Τρόπος Εκτίμησης του Συνεχούς Γλωσσικού Μοντέλου

Τα γλωσσικά μοντέλα χρησιμοποιούν για την εκτίμηση τους τις πιθανότητες που δίνουν οι ακολουθίες λέξεων. Για μία ακολουθία λέξεων με N λογία, η P(W) έχει πληροφορία σχετικά με την ακρίβεια και την πιθανότητα των ακολουθιών. Για να αποφασίσουμε την ποιότητα του μοντέλου, μπορούμε να κάνουμε χρήση των P(W). Η εντροπία και το perplexity είναι δύο βασικά μέτρα από το πεδίο της θεωρίας πληροφοριών που χρησιμοποιούνται για την αξιολόγηση ενός γλωσσικού μοντέλου.

Η πιο κοινή μονάδα μέτρησης για την αξιολόγηση ενός γλωσσικού μοντέλου είναι το ποσοστό σφάλματος κατά την αναγνώριση των λέξεων, το οποίο απαιτεί τη συμμέτοχη ενός συστήματος αναγνώρισης ομιλίας. Εναλλακτικά, μπορούμε να μετρήσουμε την πιθανότητα ότι το γλωσσικό μοντέλο μας μπορεί να θέσει ακολουθίες λέξεων και να συγκρίνει κατά πόσο αυτές μοιάζουν, χωρίς τη συμμετοχή των συστημάτων αναγνώρισης ομιλιών. Αυτό επιτυγχάνετε με την πολλαπλή χρήση της εντροπίας και ονομάζεται perplexity.

4.1.1 Εντροπία

Λαμβάνοντας υπόψη μας ότι έχουμε ένα γλωσσικό μοντέλο που θέτει πιθανότητα P(W) σε μία ακολουθία λέξεων W, μπορούμε να αντλήσουμε έναν αλγόριθμο συμπίεσης που κωδικοποιεί το κείμενο, χρησιμοποιώντας για κάθε W -log $P(W)$ bits. Η πολλαπλή χρήση της εντροπίας H(W) ενός μοντέλου $P(w_i|w_{i-n+1} \ldots w_{i-1})$ σε δεδομένα W, με μία αρκετά μεγάλη ακολουθία λέξεων , μπορεί απλά να προσαρμοστεί ως :

$$H(W) = -\frac{1}{N} log_2 P(W) \qquad 4.1.1$$

όπου $N_{\mathbf{W}}$ είναι αριθμός των λέξεων που χρησιμοποιήθηκαν από τα κείμενα W.

4.1.2 Perplexity

Το perplexity PP(W) ενός γλωσσικού μοντέλου με πιθανότητες P(W), ορίζεται ως το αντίστροφο της (γεωμετρικής) μέσης πιθανότητας που ανατίθεται από το μοντέλο μας για κάθε λέξη κατά την εκτίμηση μοντέλου. Πρόκειται για ένα μέτρο εκτίμησης, που σχετίζεται με την πολλαπλή εντροπία και είναι γνωστό ως Perplexity.

$$PP(W) = 2^{H(W)} \qquad 4.1.2$$

4.1.3 Τροποποίηση με βάσει το Μοντέλο

Εφόσον το μοντέλο μας έχει εκπαιδευτεί, για την εκτίμηση των δεδομένων μας θα χρησιμοποιήσουμε το perplexity. Αρχικά, πρέπει να υπολογίσουμε την πολλαπλή εντροπία. Για να συμβεί αυτό όμως, πρέπει πρώτα να υπολογίσουμε τη λογαριθμική πιθανότητα για όλα τα δεδομένα μας. Οπότε:

$$\log P(test\_data) = \log[P(S_1) \cdot P(S_2) \cdots P(S_T)] \qquad 4.1.3$$

για κάθε πρόταση θα έχουμε ,

$$\begin{aligned} &\log P(S_k) = \\ &= \log[P(w_1| < s >< s >)P(w_2| < s > w_1) \cdots P(w_n|w_{n-2}w_{n-1})] \\ &= \log[P(w_1| < s >< s >)] + \ \cdots \ + \log[P(w_n|w_{n-2}w_{n-1})] \end{aligned} \qquad 4.1.4$$

Κάνοντας χρήση τον κανόνα του Bays θα έχουμε:

$$\log[P(w_k|w_{k-2}w_{k-1})] = \log\frac{P(w_k)P(w_{k-2}w_{k-1}\,|w_k)}{\sum_V P(u)P(w_{k-2}w_{k-1}\,|u)} \qquad 4.1.5$$

Το perplexity υπολογίζεται από τον παρακάτω τύπο:

$$PPL = e^{\frac{-logprob}{(T+W-OOVS)}} \qquad 4.1.6$$

Όπου T είναι ο αριθμός των προτάσεων, W είναι ο συνολικός αριθμός των λέξεων που περιέχονται στο κείμενο και OOVS είναι οι λέξεις που δεν περιέχονται στο λεξικό μας.

4.2 Βάση Δεδομένων και Προεργασία

Στην διατριβή μας χρησιμοποιήσαμε δεδομένα από την Wall Street Journal, για την εκπαίδευση και την εκτίμηση του γλωσσικού μοντέλου μας. Συγκεκριμένα, χρησιμοποιήσαμε άρθρα που χρονολογούνται από το 1994 και κατά κύριο λόγο αναφέρονται σε οικονομικά και πολιτικά γεγονότα. Τα αρχεία δεδομένων είναι της μορφής WS94_*VPZ και κάθε πρόταση έχει την ακόλουθη μορφή:

<p.wsj94_001.0012.3>
<s.wsj94_001.0012.3.2>
D. D. T. is a highly persistent chemical that moves up the food chain, COMMA and it accumulates in the fatty tissue of humans .PERIOD
</s>
</p>

Στην πρώτη γραμμή παρατηρούμε ότι, παίρνουμε πληροφορίες για την πρόταση όπως από ποιο έγγραφο και από ποια παράγραφο την πήραμε, ενώ η δεύτερη γραμμή μας δείχνει από ποια πρόταση της παραγράφου την πήραμε. Άρα, έχουμε μία πρόταση που την πήραμε από το έγγραφο 001.0012, την τρίτη παράγραφο και ήταν η δεύτερη πρόταση κατά σειρά. Επίσης, τα δεδομένα δεν είναι κανονικοποιημένα και δεν έχουν επέλθει σε λεκτική ανάλυση. Αυτό σημαίνει ότι, τα σημεία στίξης ακολουθούνται από τη λέξη του σήματος, όπως παρατηρούμε με το "COMMA". Επιπλέον, υπάρχουν πολλές λανθασμένες λέξεις. Όποτε, θα πρέπει να επεξεργαστούμε τα δεδομένα πριν τα διαχωρίσουμε σε δεδομένα εκπαίδευσης και δεδομένα εκτίμησης.

Επεξεργασία Δεδομένων:

- Διαγραφή όλων των επικεφαλίδων, όπως <p.wsj94_001.0012.3> και <s.wsj94_001.0012.3.2>
- Μετατροπή όλων των δεδομένων σε πεζά
- Θα απορρίπτονται λανθασμένες λέξεις όπως «kknow», «tthey", "aare"
- Αφαίρεση όλων των αριθμών
- Κρατάμε τις λέξεις που περιγράφουν τα σημεία στίξης και απαλείφουμε τα σημεία στίξης
- Ταξινομούμε όλες τις προτάσεις ανάλογα με το μήκος τους
- Τοποθετούμε τα σύμβολα <s> και </ s> στην αρχή και στο τέλος της πρότασης
- Όταν χρησιμοποιούμε λεξικό, το οποίο περιέχει τις 2700 λέξεις με την μεγαλύτερη συχνότητα εμφάνισης, τότε οι υπόλοιπες λέξεις τοποθετούνται σε μία κλάση με το όνομα <unk> και αντικαθίστανται μέσα στα κείμενα με το συμβολισμό <unk>
- Για τη διευκόλυνση μας περιγράφουμε τις λέξεις με αριθμούς οι οποίοι μας δείχνουν στο λεξικό που έχουμε δημιουργήσει ποιες είναι οι λέξεις. Για παράδειγμα , *<s> percentage gains for <unk> ended march thirty first comma nineteen ninety three semicolon assets as of december thirty first comma nineteen ninety two assets a <unk> fee on shares held for a year or less <\s>*, ***η πρόταση παίρνει την μορφή:*** *4 691 680 14 0 385 322 69 75 1 44 54 29 90 441 22 6 901 69 75 1 44 54 18 441 8 0 1920 19 118 458 14 8 48 57 295 3*
- Τέλος, παρακάτω παρατηρούμε ένα πίνακα που μας δείχνει το μέγεθος των δεδομένων που θα κάνουμε χρήση

WS94	Train data	Test data
Αριθμός Προτάσεων	150000	6000
Αριθμός Λέξεων	4447740	164574

Πίνακας 4.1: Δεδομένα εκπαίδευσης και εκτίμησης

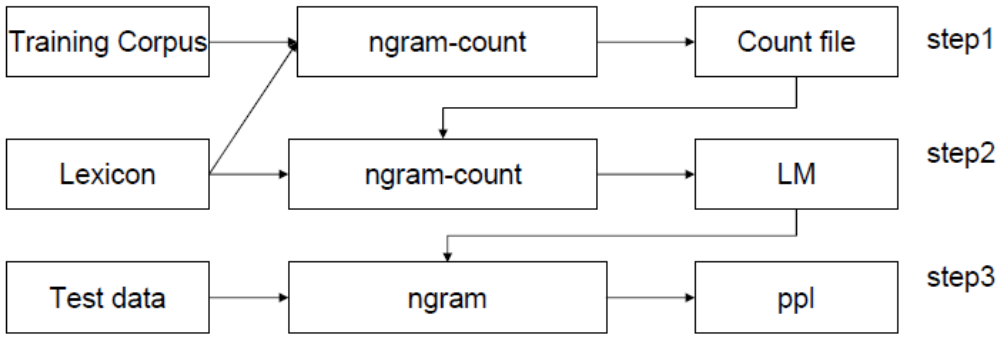
4.3 *Baseline Πειραμάτων*

Αρχικά, για baseline θα πάρουμε τα αποτελέσματα που μας δίνει το SRILM toolkit, το οποίο όμως μας δίνει αποτελέσματα με βάση το διακριτό χώρο. Οπότε, προφανώς θα είναι δύσκολο να το ξεπεράσουμε, αλλά μπορούμε τουλάχιστον να το προσεγγίσουμε. Δεύτερον, υπάρχουν πολλές εργασίες πάνω σε αυτό και κάνουν χρήση μόνο το Co-occurrence για να περιγράψουν μία λέξη. Τα αποτελέσματα των εργασιών αυτών τα ξεπεράσαμε, κάνοντας χρήση κάποιους συνδυασμούς μετρικών για την καλύτερη περιγραφή των λέξεων.

4.3.1 SRILM toolkit

Το SRILM είναι μία συλλογή από C++ βιβλιοθήκες, που είναι ξεχωριστά εκτελέσιμα προγράμματα, τα οποία μας βοηθούν στην παραμετροποίηση και την παραγωγή στατιστικών γλωσσικών μοντέλων για την αναγνώριση φωνής και άλλων εφαρμογών. Το εργαλείο είναι χρήσιμο για τη δημιουργία γλωσσικών μοντέλων που βασίζονται σε N-grams και την αξιολόγηση τους. Ουσιαστικά, μας βοήθησε να κατανοήσουμε τι συμβαίνει σε ένα γλωσσικό μοντέλο, όταν αλλάζουμε κάποιες παραμέτρους.

Η διαδικασία μοντελοποίησης με το SRILM περιγράφεται πλήρως από το παρακάτω διάγραμμα..



Σχήμα 4.1 : SRILM toolkit

Παρακάτω αναλύονται όλες οι εντολές του SRILM toolkit που κάνουμε χρήση για την εκπαίδευση και την εκτίμηση των δεδομένων μας :

- ngram-count -vocab Lexicon.file
  -text train.txt
  -order 3
  -write train_trigram
  -unk

Αυτή η εντολή δημιουργεί ένα N-gram μοντέλο (στην περίπτωση μας θέλουμε να δημιουργήσουμε ένα trigram μοντέλο) . Όπου lexicon.file είναι το λεξικό μας, train.txt είναι τα δεδομένα προς εκπαίδευση , το 3 είναι το μέγεθος του N-gram που θα δημιουργήσουμε και unk είναι η κλάση με τις λέξεις που δεν περιέχονται στο λεξικό μας. Και παίρνουμε σαν έξοδο τα παρακάτω:

Output

<unk> 668381
<unk> <unk> 112108
loan does 1
loan negotiations will 1
from university <unk> 1
from ten 40
from ten million 1
suggested the final 1
suggested the industry 1
suggested the chicago 1
majority stake of 1
majority democrats 1

Η επομένη εντολή είναι ίδια με την προηγούμενη, απλά θα αλλάξουμε τις παραμέτρους, έτσι ώστε να δημιουργήσουμε το γλωσσικό μοντέλο μας .

- ngram-count -vocab Lexicon.file
  -read train_trigram
  -order 3
  -lm trigam.train.lm

Ομοίως με πριν, το Lexicon.file είναι το λεξικό μας , train_trigram είναι N-gram αρχείο που δημιουργήσαμε και περιγράψαμε προηγουμένως και το trigram.train.lm είναι το γλωσσικό μοντέλο που δημιουργήσαμε κατά την εκπαίδευση. Και περιέχει τα παρακάτω δεδομένα:

Output

```
\data\
ngram 1=2001
ngram 2=260999
ngram 3=246775
\1-grams:
-1.354499       </s>
-99     <bos>
-99     <eos>
-99     <s>     -1.72626
-4.109913       aircraft        -0.6429651
-4.023325       airline         -0.5559635
-3.777582       airlines        -0.6082685
-1.560787       zero zero two
-1.123479       zero zero zero
\end\
```

Τέλος, κάνουμε χρήση της τελευταίας εντολής που μας υπολογίζει το perplexity με βάση το μοντέλο που εκπαιδεύσαμε πριν για τα τεστ αρχεία.

- ngram -ppl test.txt
  -order 3
  -lm trigam.train.lm

	SRILM_bigram	SRILM_trigram
PERPLEXITY	96.048	74.110

Πίνακας 4.2: Δεδομένα εκπαίδευσης και εκτίμησης

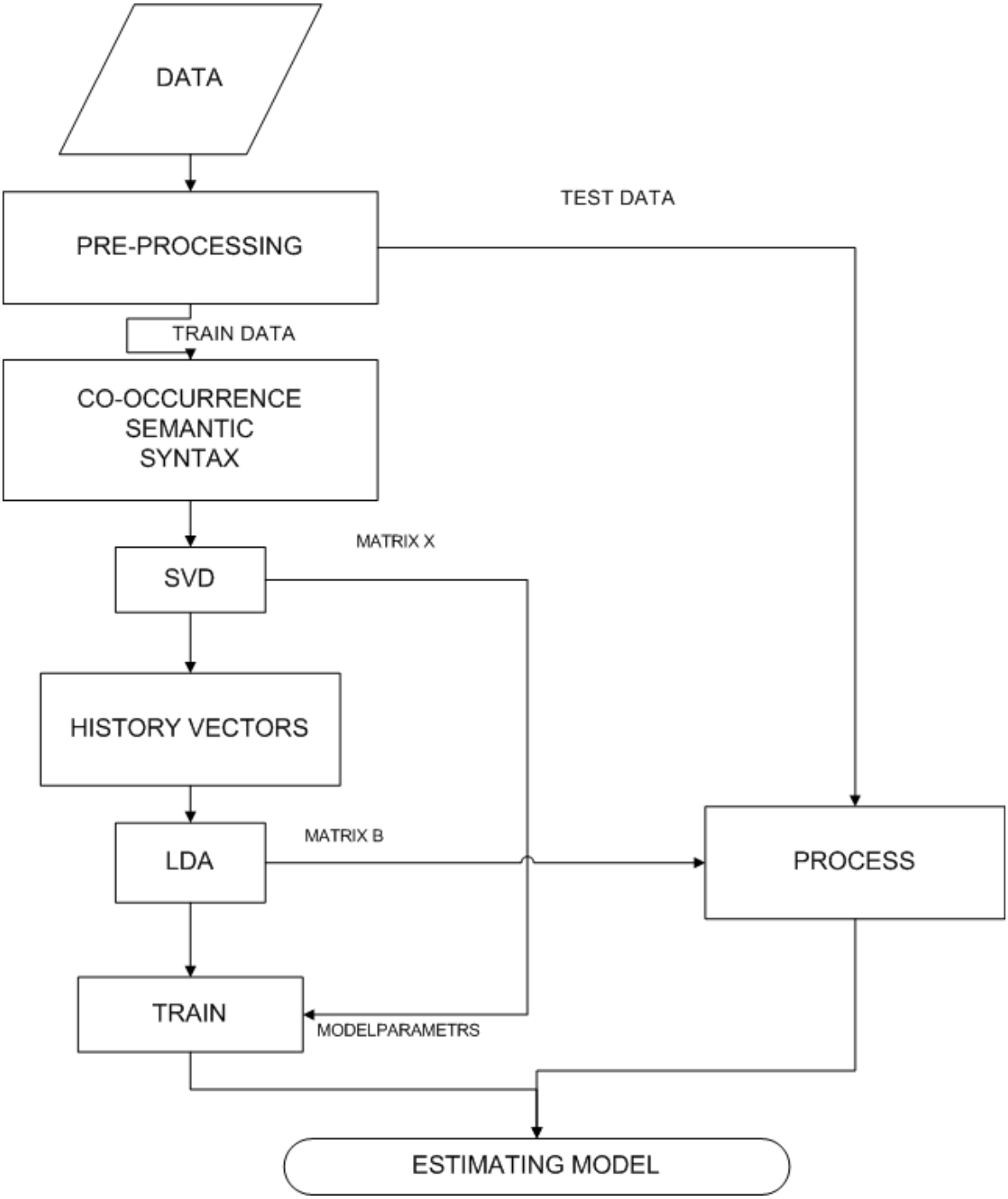
4.3.2 Baseline Συνεχούς Γλωσσικού Μοντέλου

Γενικά, έχουν πειραματιστεί αρκετοί στην εκπαίδευση γλωσσικών μοντέλων με χρήση του Co-occurrence. Οι περισσότεροι έχουν ασχοληθεί στο διακριτό χώρο, ενώ λίγοι είναι αυτοί που ασχολήθηκαν με το συνεχή χώρο. Γενικά παρατηρήσαμε ότι τα αρχικά κομμάτια του μοντέλου κρατήθηκαν σταθερά, όμως έχει γίνει χρήση μόνο του Co-occurrence και όχι μετρικών σημασιολογίας, που θα χρησιμοποιήσουμε εμείς παρακάτω. Ακόμη, δεν έχει γίνει συντακτική ανάλυση για την περιγραφή των λέξεων. Επίσης, όλα τα πειράματα είχαν σταθερή παράμετρο στη μέθοδο SVD και οι παράμετροι που άλλαξαν αφορούσαν περισσότερο τα συστατικά των Gaussian κατανομών.

Στα πειράματα που έγιναν αποδείχθηκε ότι, όσο αυξάνονται τα components των GMMs, τα αποτελέσματα που προκύπτουν είναι καλυτέρα. Επίσης, εάν κάνουμε χρήση των Tied-GMM, τα αποτελέσματα που προκύπτουν είναι ακόμη καλυτέρα και ο χρόνος εκτίμησης μειώνεται αρκετά. Εμείς για baseline θα χρησιμοποιήσουμε το καλύτερο αποτέλεσμα για τα δικά μας δεδομένα. Το κριτήριο σύγκρισης είναι το perplexity το οποίο εξηγήσαμε στο προηγούμενο κεφάλαιο πως υπολογίζεται.

Το πείραμα έγινε με χρήση του Co-occurrence πίνακα, ο οποίος είχε διαστάσεις 2700x2700 και το λεξικό μας περιέχει 2700 λέξεις. Το μέγεθος του πίνακα που επιστρέφεται με την εφαρμογή της μεθόδου SVD είναι 2700x100. Τα ιστορικά με τη χρήση της μεθόδου LDA προβάλλονται στο χώρο R^L όπου L=50. Για την εκπαίδευση του μοντέλου χρησιμοποιήσαμε GMM για κάθε λέξη του λεξικού μας που αποτελούνταν από 64 components. Και στη συνέχεια τις κάναμε T-GMM, περιέχοντας κοινό πίνακα συνδιακύμανσης για κάθε GMM. Και το perplexity το οποίο πήραμε είναι *252.336*.

Παρακάτω παρατηρούμε και τη γραφική απεικόνιση του μοντέλου που περιγράψαμε παραπάνω.



Σχήμα 4.2 : Γλωσσικό Μοντέλο

4.4 Πειράματα & Αποτελέσματα

Η αρχική ιδέα ήταν να αντικαταστήσουμε τον πίνακα Co-occurrence με έναν πίνακα ο οποίος θα μας δίνει τη σημασιολογική ομοιότητα ανάμεσα στις λέξεις και να εκτελέσουμε τον κορμό τον οποίο περιγράψαμε στα προηγούμενα κεφάλαια. Παρατηρήσαμε πως οι μετρικές ομοιότητας περιέχουν πολλά μηδενικά και δεν μας έδιναν αρκετές πληροφορίες, ώστε να μπορούμε να εκπαιδεύσουμε το γλωσσικό μοντέλο μας. Εκτός από την Normalized Google Distance, η οποία είναι μία μετρική που υπολογίζεται σε σχέση με τις συχνότητες εμφάνισης των λέξεων και μπορεί εύκολα να προσαρμοστεί στο γλωσσικό μοντέλο μας. Στην συνέχεια πειραματιστήκαμε στο να συνδυάσουμε κάπως αυτές τις μετρικές. Υπήρξαν δύο τρόποι, ο πρώτος είναι κατά την χαρτογράφηση των λέξεων να βάλουμε διαδοχικά και τα δύο διανύσματα για την περιγραφή των λέξεων, ενώ ο δεύτερος τρόπος είναι να πάρουμε ένα γραμμικό συνδυασμό των δύο μετρικών που θα χρησιμοποιήσουμε. Επίσης, επιλέξαμε στη χαρτογράφηση να βάλουμε και το συντακτικό τύπο της λέξης. Έτσι θα έχουμε την πλήρη περιγραφή της λέξης. Τέλος, πειραματιστήκαμε και με τις διαστάσεις της μεθόδου SVD και τον αριθμό των components στα GMM. Παρακάτω ακολουθεί πιο αναλυτική περιγραφή των πειραμάτων.

4.4.1 Χρήση της Normalized Google Distance(NGD)

Η Normalized Google Distance (NGD) είναι μία μετρική ομοιότητας, την οποία περιγράψαμε αναλυτικά στο δεύτερο κεφάλαιο και την οποία θα λάβουμε υπόψη μας για την περιγραφή των διανυσμάτων για κάθε λέξη του λεξικού μας. Στην περίπτωση μας κάνουμε χρήση τις 2700 λέξεις με τη μεγαλύτερη συχνότητα εμφάνισης. Δημιουργούμε έτσι ένα πίνακα 2700x2700 που θα περιέχει όλες τις τιμές της NGD. Παρατηρούμε ότι, το μήκος των διανυσμάτων για την εκπαίδευση του μοντέλου μας είναι αρκετά μεγάλο. Οπότε, κάνοντας χρήση της μεθόδου SVD θα πάρουμε ένα πίνακα 2700x100. Πλέον κάθε λέξη περιγράφεται από ένα διάνυσμα μήκους 100.

Στη συνέχεια υπολογίζουμε τα ιστορικά με βάση τα N-grams που αναφέραμε σε προηγούμενο κεφάλαιο. Και με τη χρήση της μεθόδου LDA προβάλουμε στο R^L με L=50. Κάνοντας χρήση τη Gaussian κατανομή προσπαθούμε να εκπαιδεύσουμε το γλωσσικό μοντέλο μας. Γενικά σε όλα τα πειράματα κρατάμε σταθερά τα μείγματα των Gaussian κατανομών, ώστε τα πειράματα να είναι άμεσα συγκρίσιμα. Οπότε τα GMMs αποτελούνται από 64 components. Τέλος, τα ορίζουμε να έχουν κοινούς πίνακες συνδιακύμανσης, οπότε μιλάμε για T-GMM.

Κάνοντας εφαρμογή όλων των παραπάνω το perplexity που μας δόθηκε για όλο το πλήθος των δεδομένων προς εκτίμηση (δηλαδή 6000 προτάσεις) είναι ***388.359***.

Συγκρίνοντας τα αποτελέσματα μας με τα αποτελέσματα που έχουμε για baseline παρατηρούμε ότι δεν είναι καλυτέρα και διαφέρουν αρκετά. Αρχικά περιμέναμε ότι θα περνάμε καλύτερα αποτελέσματα, αλλά όχι με τόσο μεγάλη διάφορα.

Για τη βελτιστοποίηση της NGD χρησιμοποιήσαμε ένα "τεχνητό πέναλτι" ώστε να απαλείψουμε τη βαρύτητα που δίνει σε λέξεις που έχουν μικρή συχνότητα εμφάνισης. Για παράδειγμα, δύο λέξεις μπορεί να εμφανίζονται μέσα στα κείμενα μας 2 φορές και κατά τύχη να εμφανίζονται και στην ίδια πρόταση. Οπότε, η NGD θα μας πει ότι οι λέξεις αυτές σημασιολογικά είναι ίδιες, κάτι το οποίο όμως είναι λάθος και συμβαίνει διότι δεν έχουμε μεγάλο όγκο δεδομένων προς εκπαίδευση. Επομένως, χρησιμοποιούμε το "τεχνητό πέναλτι" ώστε να απαλείψουμε αυτές τις ιδιαιτερότητες και παρατηρούμε ότι τώρα παίρνουμε αρκετά καλύτερα αποτελέσματα. Το καλύτερο αποτέλεσμα που πήραμε είναι perplexity ***324.938***.

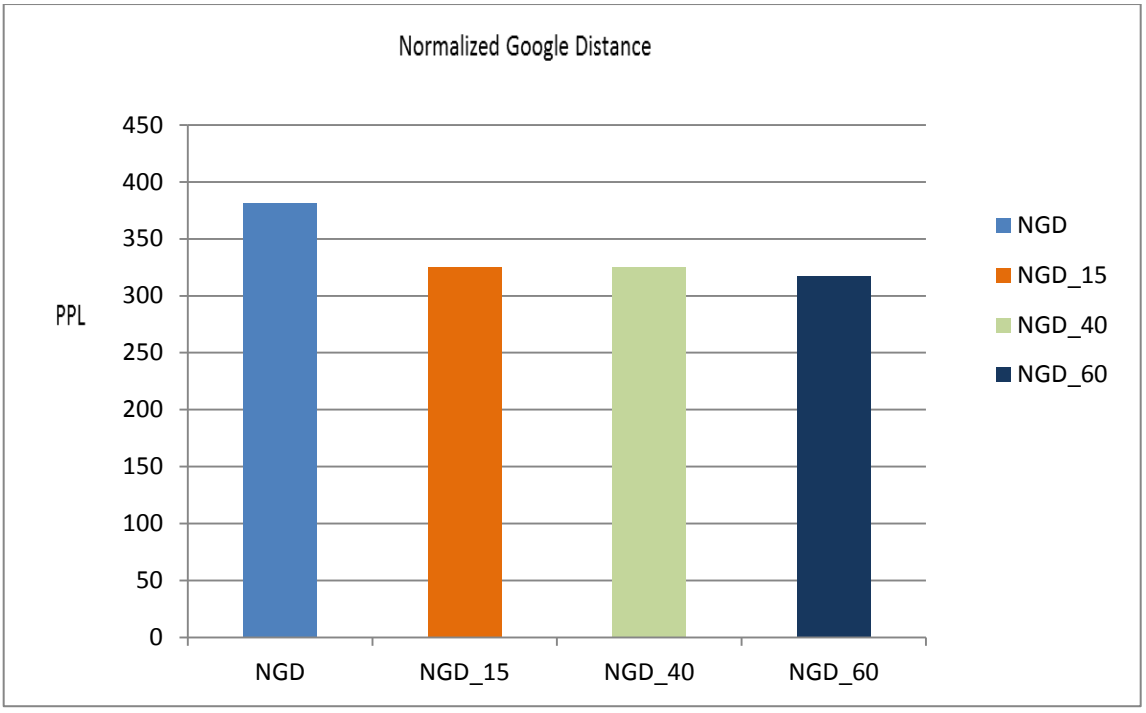
Ο παρακάτω πινάκας περιέχει τα αποτελέσματα που πήραμε κατά την εκτέλεση του κορμού μας με βάση τη NGD μετρική.

Μετρική	Τεχνητό πέναλτι	Αριθμός Προτάσεων	Διάσταση SVD	Components GMM	Perplexity PPL
Co-occurrence	-	6000	100	64T	252.336
NGD	-	6000	100	64T	388.359
NGD	15	6000	100	64T	325.601
NGD	-	1000	100	64T	381.254
NGD	15	1000	100	64T	325.430
NGD	40	1000	100	64T	324.938
NGD	60	1000	100	64T	317.562

Πίνακας 4.3: Πίνακας πειραμάτων με την χρήση της NGD

Στον παραπάνω πίνακα παρατηρούμε ότι, κάνοντας χρήση μονό την μετρική σημασιολογίας NGD δεν μπορούμε να πάρουμε καλά αποτελέσματα. Γεγονός το οποίο φαίνεται λογικό, διότι όλα τα γλωσσικά μοντέλα στηρίζονται στην συχνότητα εμφάνισης των λέξεων, κάτι το οποίο δεν το λαμβάνουμε συχνά υπόψη μας σε μεγάλο βαθμό. Στη συνέχεια, στον πίνακα που ακολουθεί, παρατηρούμε αναλυτικά τα αποτελέσματα που πήραμε κατά τη μίξη των διανυσμάτων.

Στο παρακάτω ιστόγραμμα παρατηρούμε ότι, αρχικά η μείωση είναι μεγάλη αλλά στη συνέχεια σταθεροποιείται.



Σχήμα 4.3 : Normalized Google Distance

4.4.2 Συνένωση Διανυσμάτων

Το επόμενο πείραμα που κάναμε είναι ότι προσπαθήσαμε να περιγράψουμε τις λέξεις χρησιμοποιώντας και τις δύο μετρικές. Για να το πετύχουμε αυτό, δημιουργούμε ένα διάνυσμα 100 διαστάσεων, το οποίο θα περιέχει μέσα του ένα διάνυσμα μήκους 50, το οποίο θα περιγράφει τη λέξη με βάση το Co-occurrence και διαδοχικά ένα διάνυσμα μήκους 50, το οποίο θα περιγράφει τη λέξη με βάση τη μετρική NGD. Αρχικά, υπολογίσαμε δύο πίνακες διαστάσεων 2700x2700 και με τη χρήση μας μεθόδου SVD πήραμε πίνακες 2700x50. Οπότε, για κάθε λέξη έχουμε ένα διάνυσμα 100 στοιχείων.

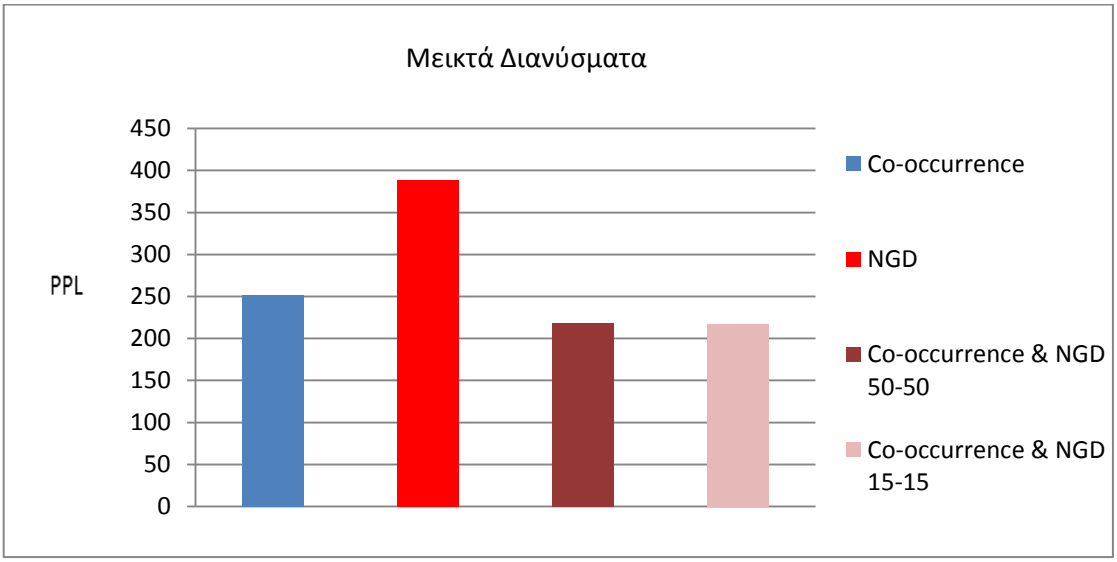
Στη συνέχεια, υπολογίζουμε τα ιστορικά με βάση τα N-grams που αναφέραμε σε προηγούμενο κεφάλαιο. Και με τη χρήση μας μεθόδου LDA προβάλουμε στο R^L με L=50. Κάνοντας χρήση τη Gaussian κατανομή προσπαθούμε να εκπαιδεύσουμε το γλωσσικό μοντέλο μας. Γενικά σε όλα τα πειράματα κρατάμε σταθερά τα μείγματα των Gaussian κατανομών, ώστε τα πειράματα να είναι άμεσα συγκρίσιμα. Οπότε τα GMMs αποτελούνται από 64 components. Τέλος, τα ορίζουμε να έχουν κοινούς πίνακες συνδιακύμανσης, οπότε μιλάμε για T-GMM.

Το αποτελέσματα που πήραμε είναι εμφανώς καλύτερα ακόμη και από αυτά που έχουμε για baseline. Το perplexity στην περίπτωση αυτή ισούται με ***218.780*** .

Μετρική	Τεχνητό πέναλτι	Αριθμός Προτάσεων	Διάσταση SVD	Components GMM	Perplexity PPL
Co-occurrence	-	6000	100	64T	252.336
NGD	-	6000	100	64T	388.359
Co-occurrence &NGD	-	6000	50-50	64T	218.780
Co-occurrence &NGD	-	6000	15-15	64T	217.513
Co-occurrence &NGD	-	1000	50-50	64T	215.861
Co-occurrence &NGD	-	1000	15-15	64T	213.762

Πίνακας 4.4: Πίνακας αποτελεσμάτων με μεικτά διανύσματα

Παρατηρούμε ότι, τα αποτελέσματα κάνοντας χρήση μεικτών διανυσμάτων βελτιώνονται, κάτι το οποίο είναι λογικό, διότι δίνουμε έμφαση στις λέξεις που εμφανίζονται αρκετά συχνά αλλά και στις λέξεις που έχουν την ίδια σημασιολογία, κάτι το οποίο όπως θα δούμε και παρακάτω, μας δίνει την καλύτερη δυνατή απόδοση.



Σχήμα 4.4 : Ιστόγραμμα με μεικτά διανύσματα

4.4.3 Γραμμικός Συνδυασμός Των Μετρικών

Στο πείραμα αυτό επιλέξαμε να "παντρέψουμε" τις μετρικές έτσι ώστε να δώσουμε καλύτερη περιγραφή στα διανύσματα που περιγραφούν τις λέξεις μας. Γενικά η ιδέα αυτή προέκυψε με σκοπό να δώσουμε πιο ελεύθερη περιγραφή στις λέξεις και να μην είναι τόσο καθοδηγούμενη, όσο κάνοντας χρήση μόνο του Co-occurrence. Κρατάμε ως βασική περιγραφή των διανυσμάτων τον πίνακα Co-occurrence και στη συνέχεια κάνουμε γραμμικούς συνδυασμούς με όλες τις μετρικές που αναφέραμε παραπάνω. Η εξίσωση που περιγράφει το γραμμικό συνδυασμό είναι η παρακάτω:

$$M1M2 = \Lambda * M1 + (1 - \Lambda)M2 \qquad 4.4.1$$

$$M1M2M3 = \Lambda * M1 + \left(1 - \frac{\Lambda}{2}\right)M2 + \left(1 - \frac{\Lambda}{2}\right)M3 \qquad 4.4.2$$

όπου Λ είναι το ποσοστό βαρύτητας που δίνουμε στην κάθε μετρική και M1 θα είναι πάντα ο Co-occurrence, ενώ M2 θα είναι μία μετρική σημασιολογίας από αυτές που αναφέραμε στο δεύτερο κεφάλαιο.

Κάνοντας λοιπόν το γραμμικό συνδυασμό, θα έχουμε ένα πίνακα 2700x2700 που τα στοιχεία του θα λαμβάνουν πληροφορία από δύο μετρικές, ανάλογα βέβαια όμως από τη βαρύτητα που θα δώσουμε στην κάθε μετρική. Μετά με τη χρήση της μεθόδου SVD θα πάρουμε ένα πίνακα 2700x100.

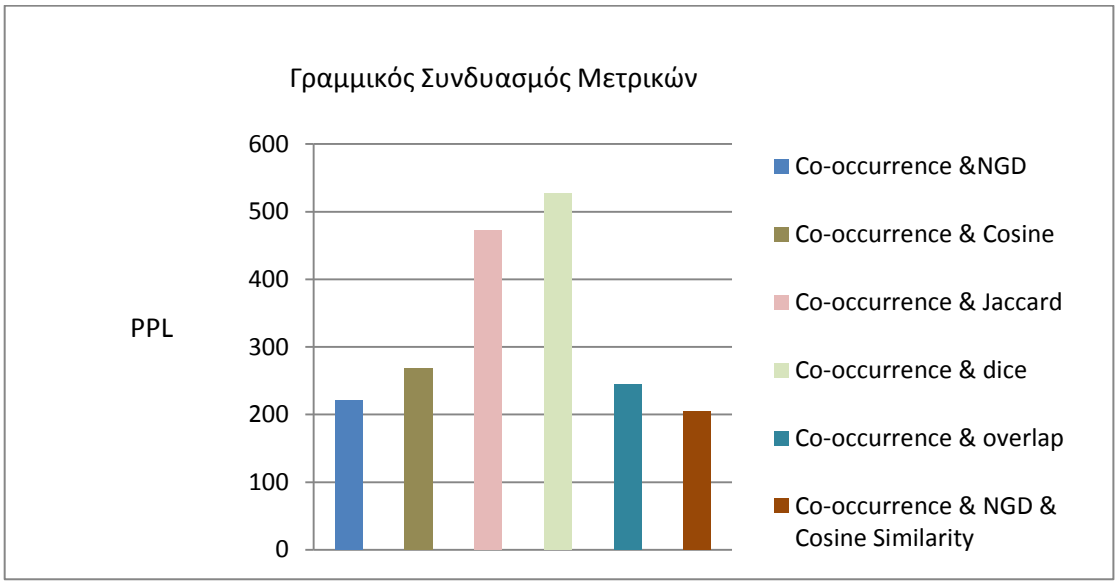
Στη συνέχεια, υπολογίζουμε τα ιστορικά με βάση τα N-grams που αναφέραμε σε προηγούμενο κεφάλαιο. Και με τη χρήση της μεθόδου LDA προβάλουμε στο R^L με L=50. Κάνοντας χρήση την Gaussian κατανομή προσπαθούμε να εκπαιδεύσουμε το γλωσσικό μοντέλο μας. Γενικά, σε όλα τα πειράματα κρατάμε σταθερά τα μείγματα των Gaussian κατανομών, ώστε τα πειράματα να είναι άμεσα συγκρίσιμα. Οπότε, τα GMMs αποτελούνται από 64 components. Τέλος, τα ορίζουμε να έχουν κοινούς πίνακες συνδιακύμανσης, οπότε μιλάμε για T-GMM. Τα αποτελέσματα αναγράφονται πλήρως στον πινάκα που ακλουθεί παρακάτω.

Τέλος, χρησιμοποιούμε την εξίσωση (4.4.2) και εισάγουμε για M1 το Co-occurrence, M2 τη Normalized Google Distance, M3 τη Cosine Similarity και το Λ ισούται με 0.8. Το αποτέλεσμα που πήραμε κάνοντας χρήση ακριβώς τις ίδιες παραμέτρους που χρησιμοποιήθηκαν και παραπάνω είναι ***205.412***.

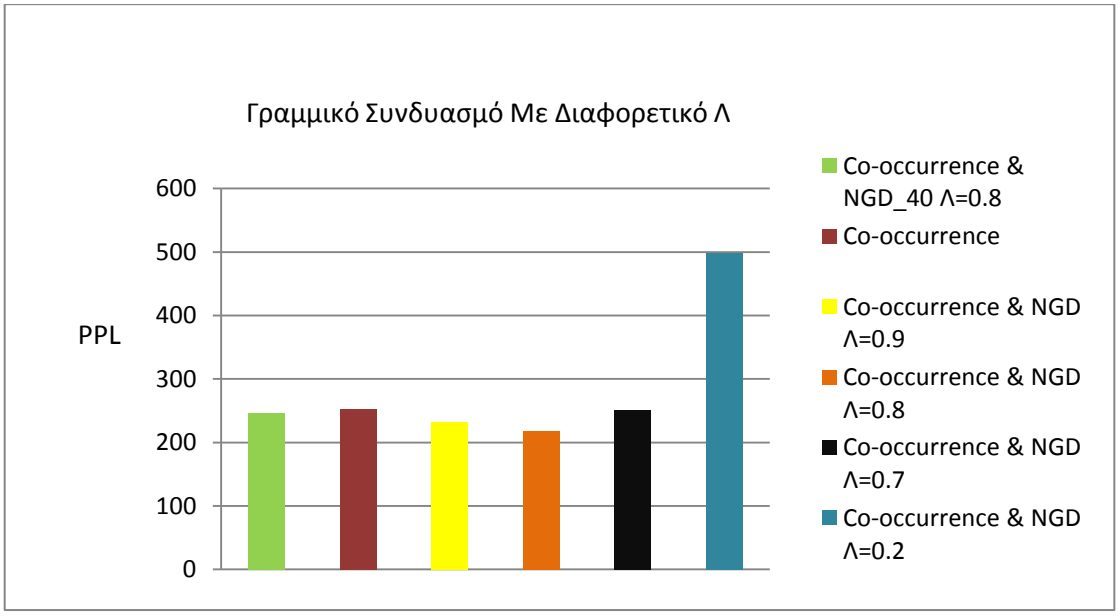
Μετρική	Τεχνητό πέναλτι	Λ	Αριθμός Προτάσεων	Διάσταση SVD	Components GMM	Perplexity PPL
Co-occurrence &NGD	-	0.8	6000	100	64T	221.227
Co-occurrence &NGD	-	0.8	5000	100	64T	221.348
Co-occurrence &Cosine	-	0.8	5000	100	64T	267.678
Co-occurrence &Jaccard	-	0.8	5000	100	64T	472.479
Co-occurrence &dice	-	0.8	5000	100	64T	527.703
Co-occurrence &overlap	-	0.8	5000	100	64T	244.611
Co-occurrence &NGD	-	0.8	1000	100	64T	218.348
Co-occurrence &NGD	40	0.8	1000	100	64T	246.908
Co-occurrence &NGD	-	0.9	1000	100	64T	232.084
Co-occurrence &NGD	-	0.7	1000	100	64T	250.523
Co-occurrence &NGD	-	0.2	1000	100	64T	498.311
Co-occurrence &NGD	-	0.8	1000	100	64T	498.311

Πίνακας 4.5: Αποτελέσματα με χρήση του γραμμικού συνδυασμού.

Στον παραπάνω πίνακα παρατηρούμε ότι κάνοντας χρήση τις μετρικές σημασιολογίας που δεν περιέχουν αρκετές πληροφορίες για την κάθε λέξη μας, όπως είναι η Jaccard ή η cosine similarity, το γλωσσικό μας μοντέλο δεν ανταποκρίνεται σωστά, κάτι το οποίο είναι λογικό. Αντίθετα όμως, όταν κάνουμε χρήση της NGD, παρατηρούμε ότι το γλωσσικό μας μοντέλο ανταποκρίνεται αρκετά καλά. Στην περίπτωση αυτή, πάλι δίνουμε βαρύτητα στον Co-occurrence πίνακα, αλλά λαμβάνουμε υπόψη μας και τις λέξεις που έχουν την ίδια σημασιολογία.



Σχήμα 4.5 : Ιστόγραμμα με γραμμικό συνδυασμό όλων των μετρικών.



Σχήμα 4.6 : Ιστόγραμμα με γραμμικό συνδυασμό με διαφορετικό Λ.

4.4.4 Χρήση Συντακτικής Ανάλυσης

Όπως είναι γνωστό μία λέξη μπορεί να περιγραφεί από τη λεκτική της σημασία και από το συντακτικό της τύπο. Μέχρι τώρα χρησιμοποιούμε μονό τη λεκτική ανάλυση για την περιγραφή της λέξης μας. Ήρθε η ώρα να προσθέσουμε στο διάνυσμα περιγραφής κάθε λέξης και το συντακτικό τύπο. Αυτό έγινε με τη χρήση της μεθόδου του γραμμικού συνδυασμού. Εφόσον έχουμε δημιουργήσει τα διανύσματα τα οποία προκύπτουν από τη λεκτική ανάλυση, προσθέτουμε και τον αριθμό, ο οποίος μας δηλώνει τι συντακτικού τύπου είναι η λέξη. Η πρόσθεση του αριθμού γίνεται μετά από τη χρήση της μεθόδου SVD, ώστε να έχει μεγαλύτερη βαρύτητα. Αυτό επιτυγχάνεται γιατί τα διανύσματα που προκύπτουν με τη μέθοδο SVD περιέχουν τιμές από 0.5 έως -0.5 . Και εμείς προσθέτουμε τιμές από 1-23. Η μέθοδος με την οποία αντιστοιχίσαμε τους αριθμούς αυτούς με τις λέξεις περιγράφεται στο δεύτερο κεφάλαιο.

Στη συνέχεια υπολογίζουμε τα ιστορικά με βάση τα N-grams που αναφέραμε σε προηγούμενο κεφάλαιο. Και με τη χρήση της μεθόδου LDA προβάλουμε στο R^L με L=50. Κάνοντας χρήση τη Gaussian κατανομή προσπαθούμε να εκπαιδεύσουμε το γλωσσικό μοντέλο μας. Γενικά, σε όλα τα πειράματα κρατάμε σταθερά τα μείγματα των Gaussian κατανομών, ώστε τα πειράματα να είναι άμεσα συγκρίσιμα. Οπότε, τα GMMs αποτελούνται από 64 components. Τέλος, τα ορίζουμε να έχουν κοινούς πίνακες συνδιακύμανσης, οπότε μιλάμε για T-GMM.

Παρακάτω υπάρχει ένας πίνακας στον οποίο προσθέτουμε το γραμμικό συνδυασμό και το συντακτικό τύπο, για την καλύτερη περιγραφή των αποτελεσμάτων.

Μετρική	Τεχνητό πέναλτι	Λ	Αριθμός Προτάσεων	Διάσταση SVD	Components GMM	Perplexity PPL
Co-occurrence &NGD	-	0.8	1000	30	64T	202.590
Co-occurrence &NGD &Syntax	-	0.8	1000	30-1	64T	198.56

Πίνακας 4.6: Αποτελέσματα με χρήση γραμμικού συνδυασμού & συντακτικού τύπου.

Παρατηρούμε ότι, τα αποτελέσματα βελτιώθηκαν κάνοντας χρήση του συντακτικού τύπου. Κάτι το οποίο είναι λογικό διότι λίγες είναι οι ακολουθίες οι όποιες γίνονται με βάση τη σύνταξη. Αρά όταν χρησιμοποιούμε τη μέθοδο LDA, οι κλάσεις που δημιουργούνται είναι καλύτερες, διότι δίνουμε αρκετά μεγάλη βαρύτητα στη σύνταξη.

4.4.5 Ομαδοποίηση Δεδομένων

Εάν ανατρέξουμε στο δεύτερο κεφάλαιο, θα παρατηρήσουμε ότι όλοι οι πινάκες που προκύπτουν είναι αρκετά αραιοί, δηλαδή περιέχουν παρά πολλά μηδενικά. Για το λόγο αυτό αποφασίσαμε να κάνουμε ένα είδος ομαδοποίησης. Η ομαδοποίηση των λέξεων θα γίνεται με βάση τη σημασιολογική ομοιότητα NGD. Στην ουσία θα δημιουργήσουμε 2700 κλάσεις και με βάση την εντροπία θα επιλέξουμε 100 κλάσεις από αυτές. Για κάθε λέξη θα κάνουμε χρήση τον τύπο της εντροπίας που περιγράφουμε στην ενότητα (3.7.1). Τη χρήση της εντροπίας θα την κάνουμε βάσει του term-document πινάκα. Ο Co-occurrence είναι ένας term-document πινάκας. Στην ουσία, θα επιλέξουμε τις λέξεις που έχουν τη μεγαλύτερη συχνότητα εμφάνισης με τις υπόλοιπες λέξεις, σε σχέση με τον αριθμό εμφάνισης της λέξης μας. Εφόσον βρούμε τις 100 λέξεις με τη μεγαλύτερη εντροπία, στη συνέχεια, θα κάνουμε χρήση των κλάσεων που έγιναν με βάση τις λέξεις αυτές.

Η δυνατότητα ομαδοποίησης γίνεται ώστε να μειώσουμε τις διαστάσεις του πινάκα. Στην ουσία κάνουμε feature selection βάσει της εντροπίας. Όμως η διαφορά είναι ότι τώρα δε θα λάβουμε υπόψη μας τη συνύπαρξη μόνο με μία λέξη, αλλά τη συνύπαρξη της λέξης με τις άλλες λέξεις που βρίσκονται στην ίδια κλάση. Στην ουσία θα αθροίζουμε όλες τις τιμές που θα μας δείχνουνε πόσες φορές έχει εμφανιστεί η λέξη μας με τις λέξεις της ίδιας κλάσης. Επειδή υπάρχουν αρκετά μεγάλες τιμές θα κάνουμε smoothing. Το smoothing θα γίνει με βάση το λογάριθμο, δηλαδή θα αντικαταστήσουμε τις τιμές του πινάκα με τις τιμές που θα μας δίνει η συνάρτηση $w_{ij} = \log(w_{ij} + 1)$.

Στη συνέχεια, υπολογίζουμε τα ιστορικά με βάση τα N-grams που αναφέραμε σε προηγούμενο κεφάλαιο. Και με τη χρήση της μεθόδου LDA προβάλουμε στο R^L με L=50. Κάνοντας χρήση τη Gaussian κατανομή προσπαθούμε να εκπαιδεύσουμε το γλωσσικό μοντέλο μας. Γενικά σε όλα τα πειράματα κρατάμε σταθερά τα μείγματα των Gaussian κατανομών, ώστε τα πειράματα να είναι άμεσα συγκρίσιμα. Οπότε, τα GMMs αποτελούνται από 64 components. Τέλος, τα ορίζουμε να έχουν κοινούς πίνακες συνδιακύμανσης, οπότε μιλάμε για T-GMM.

Τα αποτελέσματα που πήραμε δεν ήταν αρκετά καλά. Το perplexity για την παραπάνω μέθοδο ισούται με ***511.059*** .

Ο παρακάτω πινάκας περιέχει τα αποτελέσματα χωρίς τη χρήση SVD, αλλά με έναν τρόπο ομαδοποίησης που αναφέραμε προηγουμένως.

Μετρική	Τεχνητό πέναλτι	Αριθμός Προτάσεων	Διάσταση SVD	Components GMM	Perplexity PPL
Co-occurrence	-	1000	100	64T	252.336
NGD	-	1000	100	64T	388.359
Clustering	-	1000	100(όχι SVD)	64T	511.059

Πίνακας 4.7: Αποτελέσματα με χρήση ομαδοποιημένων λέξεων .

Είναι εμφανές ότι χωρίς τη χρήση του SVD, τα αποτελέσματα είναι χειρότερα, για το λόγο αυτό δεν δώσαμε περισσότερη σημασία σε άλλους τρόπους για την μείωση των διαστάσεων. Στη συνέχεια θα πειραματιστούμε με τις διαστάσεις των διανυσμάτων που έχουμε προς εκπαίδευση. Θα πάρουμε πολλές διαφορετικές διαστάσεις και θα τις εφαρμόσουμε σε όλες τις μεθόδους που κάναμε παραπάνω, ώστε να δούμε πως ανταποκρίνονται. Τέλος, θα δικαιολογήσουμε τα αποτελέσματα που πήραμε.

4.4.6 Μείωση Διαστάσεων

Παρατηρούμε ότι, με τη χρήση της μεθόδου SVD τα αποτελέσματα είναι αρκετά καλυτέρα, από το να κάνουμε ένα άλλο είδος feature selection όπως αναφέραμε στην προηγούμενη ενότητα. Κάτι το οποίο είναι λογικό, διότι τα διανύσματα τα οποία προκύπτουν μετά τη μείωση των διαστάσεων από τον SVD περιέχουν πληροφορίες για όλα τα στοιχειά του αρχικού διανύσματος. Αυτό δεν επιτυγχάνεται με καμιά άλλη μέθοδο. Γενικά, στις άλλες μεθόδους κοιτάμε ποιες λέξεις έχουν τη μεγαλύτερη βαρύτητα και δημιουργούμε μικρότερα διανύσματα με βάση αυτές.

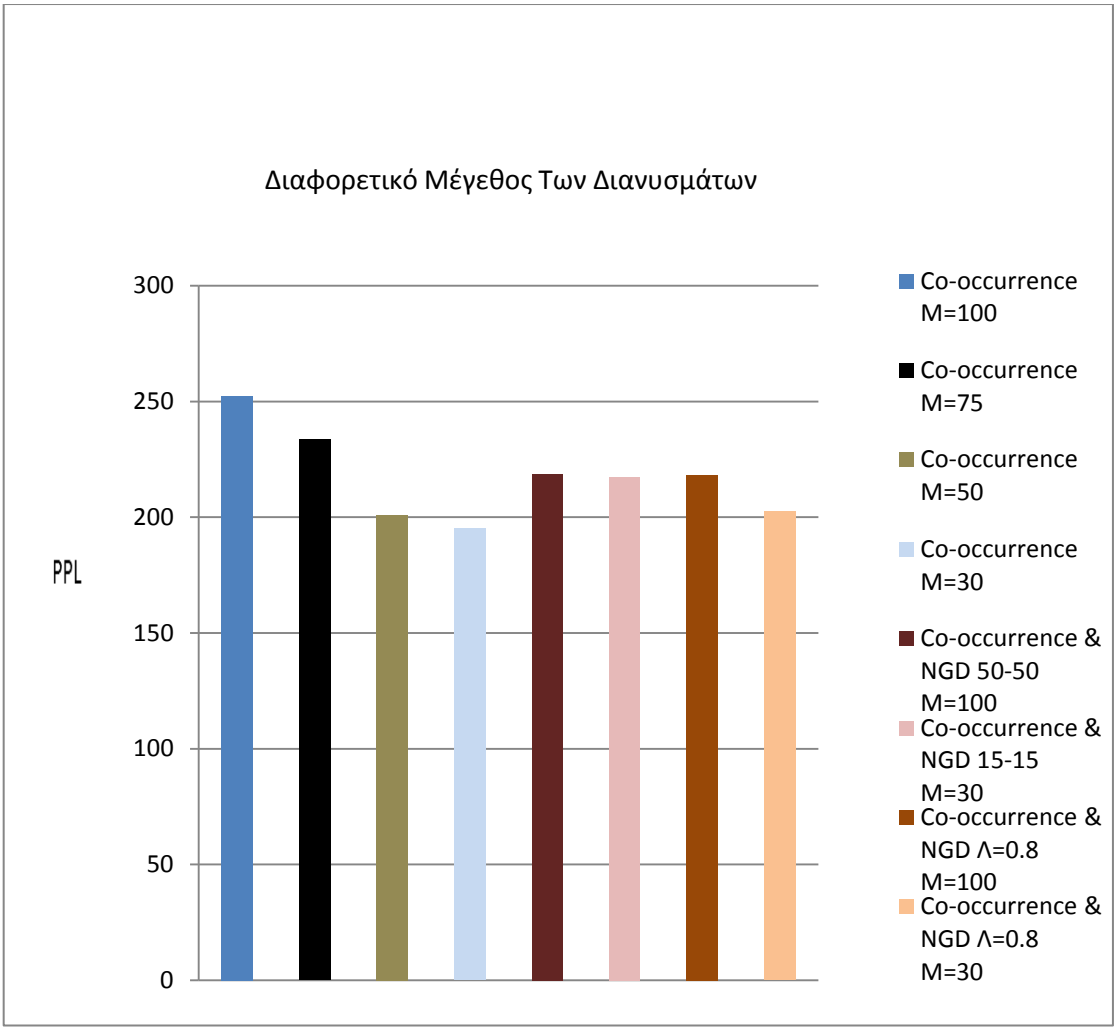
Αποφασίσαμε να αυξομειώσουμε τις διαστάσεις της μεθόδου SVD ώστε να παρατηρήσουμε τι συμβαίνει. Γενικά πήραμε τιμές για M=30,50,75,100,150 όπου M είναι η διάσταση των διανυσμάτων που μας επιστρέφονται. Επίσης, εφαρμόσαμε μερικές από τις τιμές αυτές και στα παραπάνω πειράματα και γενικά καταλήγαμε συνεχεία σε σχετικά κοντινά αποτελέσματα.

Στο πείραμα αυτό παρατηρούμε ότι, όσο μειώνουμε τις διαστάσεις των διανυσμάτων, τα αποτελέσματα που προκύπτουν είναι καλυτέρα. Γεγονός το οποίο είναι λογικό, διότι αποκόπτουμε πληροφορία η οποία δεν είναι και τόσο σημαντική. Επίσης, παρατηρούμε ότι, σε όλα τα πειράματα όσο μικραίνουμε τις διαστάσεις, τόσο καλύτερη απόδοση μας δίνει το μοντέλο μας. Όμως, παρατηρούμε ότι, το καλύτερο αποτέλεσμα το παίρνουμε για M=30, κάνοντας χρήση του Co-occurrence, κάτι το οποίο είναι λογικό, διότι το διάνυσμα που προκύπτει είναι αρκετά καθοδηγούμενο. Όπως αναφέραμε και στην αρχή της διατριβή μας, θέλουμε να δώσουμε την ελευθερία στην περιγραφή των διανυσμάτων μας. Για το λόγο αυτό κάνουμε χρήση της σημασιολογίας και της σύνταξης στην περιγραφή των διανυσμάτων.

Στον παρακάτω πίνακα παρατηρούμε τις αλλαγές που έχουμε όταν πειραματιζόμαστε με τις διαστάσεις των διανυσμάτων που θα έχουμε προς εκπαίδευση.

Μετρική	Τεχνητό πέναλτι	Λ	Αριθμός Προτάσεων	Διάσταση SVD	Components GMM	Perplexity PPL
Co-occurrence	-	-	6000	100	64T	252.336
Co-occurrence	-	-	6000	150	64T	863.675
Co-occurrence	-	-	6000	75	64T	233.569
Co-occurrence	-	-	6000	50	64T	200.842
Co-occurrence	-	-	6000	30	64T	195.117
Co-occurrence &NGD	-	-	6000	50-50	64T	218.780
Co-occurrence &NGD	-	-	6000	15-15	64T	217.513
Co-occurrence &NGD	-	0.8	1000	100	64T	218.348
Co-occurrence &NGD	-	0.8	1000	30	64T	202.590

Πίνακας 4.8: Αποτελέσματα ανάλογα με την παράμετρο M του SVD



Σχήμα 4.7 : Ιστόγραμμα ανάλογα με την παράμετρο M του SVD.

4.4.7 GMM και T-GMM

Γενικά όλες οι έρευνες που έχουν γίνει μέχρι τώρα επικεντρώνονται στον τύπο και τις παραμέτρους των GMM και κάνουν χρήση GMM και T-GMM για να αναλύσουν τα συμπεράσματα τους. Εμείς θα χρησιμοποιήσουμε T-GMM. Κάνοντας χρήση την εντολή HHEd (του εργαλείου HTK), μπορούμε να μετατρέψουμε ένα GMM μοντέλο σε T-GMM. Με τις εντολές HCompV και HRest εκπαιδεύουμε το μοντέλο μας.

Στα πειράματά μας θα ασχοληθούμε με ολόκληρους τους πινάκες συνδιακύμανσης. Θα χρησιμοποιήσουμε επίσης διαγώνιους και σφαιρικούς.

Επιπλέον, θα επιλέξουμε να δούμε πως αλλάζει το perplexity κάνοντας χρήση GMM που περιγράφονται από 2, 32, 64 components.

Τέλος, στον παρακάτω πίνακα παρατηρούμε πως μεταβάλετε η απόδοση του συστήματος, ανάλογα με τον αριθμό των components που περιέχουν τα GMMs.

Μετρική	Τεχνητό πέναλτι	Λ	Αριθμός Προτάσεων	Διάσταση SVD	Components GMM	Perplexity PPL
Co-occurrence &NGD	40	0.8	1000	100	64T	246.908
Co-occurrence &NGD	40	0.8	1000	100	32T	246.424
Co-occurrence &NGD	40	0.8	1000	100	2T	232.078
NGD	60	-	1000	100	64T	317.562
NGD	60	-	1000	100	32T	317.377
NGD	60	-	1000	100	2T	297.133
Co-occurrence &NGD	-	0.8	1000	30	64T	202.590
Co-occurrence &NGD	-	0.8	1000	30	32T	202.155
Co-occurrence &NGD	-	0.8	1000	30	2T	206.833

Πίνακας 4.9: Αποτελέσματα με χρήση διαφορετικών components στα GMM

Από τον παραπάνω πινάκα είναι εύκολο να διαπιστώσουμε ότι, κάνοντας χρήση 32 ή 64 components, τα αποτελέσματα που προκύπτουν είναι αρκετά καλά, σε σχέση με τα αποτελέσματα που παίρνουμε όταν κάθε GMM περιγράφεται από 2 component. Γενικά, πειραματιστήκαμε με T-GMM, διότι όλες οι έρευνες που έχουν γίνει μέχρι τώρα στα δεδομένα που έχουμε προς εκπαίδευση στο συνεχή χώρο χρησιμοποιούσαν μόνο GMM. Παρατηρήθηκε λοιπόν ότι τα αποτελέσματα για T-GMM είναι καλυτέρα από την χρήση των GMM.

4.5 Σύνοψη Συμπερασμάτων

Βασιζόμενοι λοιπόν στα παραπάνω αποτελέσματα, είναι εύκολο να διακρίνουμε πως στα γλωσσικά μοντέλα την κύρια σημασία έχουν οι συχνότητες εμφάνισης των λέξεων. Κάναμε χρήση μετρικών σημασιολογίας και παρατηρήσαμε ότι τα γλωσσικά μοντέλα τα οποία δημιουργήθηκαν δεν είχαν την απόδοση την οποία μας έδινε το γλωσσικό μοντέλο το οποίο εκπαιδεύτηκε με βάση το Co-occurrence.

Επιπλέον, παρατηρήσαμε ότι, δίνοντας περισσότερα χαρακτηριστικά στην περιγραφή των λέξεων, τα αποτελέσματα τα οποία παίρναμε ήταν ποιοτικότερα. Με τον τρόπο αυτό δίνουμε μία ελευθερία στην περιγραφή των λέξεων, αν και περισσότερη έμφαση δίναμε στον πινάκα Co-occurrence.

Επίσης, προσπαθήσαμε με ένα διαφορετικό τρόπο να μειώσουμε τις διαστάσεις των διανυσμάτων μας , αντί να κάνουμε χρήση της μεθόδου SVD. Η ιδέα ήταν να κάνουμε μία ομαδοποίηση των λέξεων και στη συνέχεια να πάρουμε ένα διάνυσμα το οποίο θα περιέχει τις λέξεις με τη μεγαλύτερη εντροπία. Παρατηρήσαμε τότε ότι το μοντέλο μας δεν ήταν αποδοτικό και απείχε αρκετά από το αποτέλεσμα που παίρναμε με βάση τη μέθοδο SVD.

Από την άλλη, προσπαθήσαμε να αυξομειώσουμε τις διαστάσεις και να παρατηρήσουμε πως ανταποκρίνεται το μοντέλο μας σε αυτό. Παρατηρήσαμε, λοιπόν, μεγάλη βελτίωση στην απόδοση του μοντέλου μας, όσο μειώναμε τις διαστάσεις των διανυσμάτων.

Τέλος, ασχοληθήκαμε με τον αριθμό των components που περιγράφουν τα GMM, όπως και με τη χρήση των T-GMM. Παρατηρήσαμε ότι, όσο αυξάναμε τον αριθμό των Components, η απόδοση του μοντέλου μας βελτιωνόταν, παρόλο που από ένα σημείο και μετά υπήρχε καμπή, δηλαδή η απόδοση του μοντέλου μας γινόταν χειρότερη.

5

Επίλογος

5.1 Σύνοψη και συμπεράσματα

Στόχος της διατριβή μας, ήταν να κάνουμε χρήση σημασιολογικών μετρικών για την εκπαίδευση ενός γλωσσικού μοντέλου και να παρατηρήσουμε την απόδοση αυτού. Είναι εμφανές, από τα αποτελέσματα που πήραμε κατά την διάρκεια των πειραμάτων μας , πως ακέραια σημασία για τα γλωσσικά μοντέλα έχει ο πινάκας Co-occurrence. Επίσης, έπρεπε να βρούμε κάποιους τρόπους για τη μείωση των διαστάσεων, όπως είναι η μέθοδος SVD ή η ομαδοποίηση των δεδομένων μας, κάνοντας χρήση της εντροπίας. Τέλος, έπρεπε να βρούμε μία κατανομή η οποία να μπορεί να περιγράφει αρκετά καλά τα δεδομένα, ώστε να μεταφερθούμε στο συνεχή χώρο.

Εν τέλει, διαπιστώσαμε ότι, κάνοντας χρήση όσο το δυνατόν περισσότερων στοιχείων για την περιγραφή των λέξεών μας, το γλωσσικό μας μοντέλο βελτίωνε αισθητά την απόδοσή του.

5.2 Μελλοντικές επεκτάσεις

Στην εργασία μας χρησιμοποιήθηκαν αρκετά βήματα προ-επεξεργασίας μέχρι να έρθουν τα κείμενα στη τελική τους μορφή . Θα μπορούσε όμως να γίνει χρήση επιπλέον βημάτων, όπως για παράδειγμα χρήση του λεξικού Wordnet (http://en.wikipedia.org/wiki/WordNet), το οποίο περιέχει συνώνυμα λέξεων, συντακτικά στοιχεία των λέξεων κ.ά.. Επίσης, για το κομμάτι του stemming χρησιμοποιείται ο αλγόριθμος του Porter, υπάρχουν όμως ακόμη αρκετοί τέτοιοι αλγόριθμοι που έχουν τη δυνατότητα να "κόβουν" λιγότερες λέξεις, ή να "αφήνουν" περισσότερες, όπως για παράδειγμα οι αλγόριθμοι Lovins , Paice/Husk , Dawnson κ.ά..

Στη διατριβή μας, χρησιμοποιήσαμε πολλούς τρόπους για την περιγραφή των διανυσμάτων κάθε λέξης, ώστε να μην υπάρχει η καθοδήγηση που υπήρχε με βάση το Co-occurrence. Χρησιμοποιήσαμε N-grams για την περιγραφή των ιστορικών. Όμως, για τη μείωση των διαστάσεων των διανυσμάτων χρησιμοποιήσαμε τη μέθοδο SVD. Υπάρχει και η εξέλιξη της μεθόδου που λέγεται truncated SVD που θα μπορούσε να χρησιμοποιήσει κάποιος για τη βελτίωση του μοντέλου.

Επίσης, υπάρχουν και άλλες μέθοδοι οι οποίες μπορούν να χρησιμοποιηθούν για feature selection αντί της μεθόδου SVD, την οποία κάναμε χρήση εμείς. Όπως για παράδειγμα είναι οι μέθοδοι NMF, MRMR κλπ. ή ένα διαφορετικό είδος ομαδοποίησης από αυτό που χρησιμοποιήσαμε εμείς για τη μείωση των διαστάσεων.

Τέλος, θα μπορούσε να χρησιμοποιηθεί ένα είδος adaptation, έτσι ώστε να συλλέγονται περισσότερες πληροφορίες για την περιγραφή κάθε λέξης.

6

Βιβλιογραφία

1 M.Afify, O. Siohan and R. Sarikaya, 2007. *Gaussian Mixture Language Models for Speech Recognition*, ICASSP, Honolulu, Hawaii.

2 Xuedong Huang, Alex Acero, Hsiao- Wuen Hon, 2001. *Spoken Language Processing*, Prentige Hall PTR

3 Wei Chen, 2007. *Building Language models on continuous space using gaussian mixture models*. Technical report, Carnegie Mellon University

4 Berlin Chein, *Introduction to SRILM Toolkit*, 2007. Department of Computer Science & Information Engineering National Taiwan Normal University

5 D. Oikonomidis, *2002. Language Models for Speech Recognition*. Technical University of Crete, MSc Thesis

6 R. Duda, P.Hart, and D. Stork, *Pattern Classification (Second Edition)*. Wiley-Interscience, October 2000.

7 A. Stolcke*," SRILM – an extensible language modeling toolkit,"* Proc. ICSLP'02, Denver, Colorado, Sept., 2002.

8 R. Sarikaya, M.Afify, Brian Kingsbury, *2007. Tied–Mixture Language Modeling in Continuous Space.* ICASSP, Honolulu, Hawaii.

9 P. Brown, P. V. DeSouza, R. L. Mercer, V. J. Della Pietra, J. C. Lai. *Class-Based n-gram Models of Natural Language*, IBM T. J. Watson research Center.

10 S. Geirhofer, 2004. *Feature Reduction with Linear Discriminant Analysis and its Performance on Phoneme Recognition*, University of Illinois at Urbana-Champaign Department of Electrical and Computer Engineering

11 I. T. Nabney, 2004. Netlab*: Algorithms for Pattern Recognition*. APR, Springer.

12 S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev P. Woodland, *2006, The HTK Book*, Cambridge University Engineering Department

13 S. Theodoridis, K. Koutroumbas, 2009. *Pattern Recognition, 4th edition*. Elsevier Inc. AP.

14 L. van den Maaten, E. Postma, J. van den Herik, 2009. Dimensionality Reduction: A Comparative Review. TiCC

15 D. Bollegala, Y. Matsuo, and M. Ishizuka, ―*Measuring semantic similarity between words using web search engines*,‖ in Proc. of International Conference on World Wide Web, 2007, pp. 757–766.

16 Rudi L. Cilibrasi and Paul M.B. Vita´ nyi, ―*The Google SimilarityDistance*‖, IEEE Transactions On Knowledge And Data Engineering, Vol.19, No. 3, March 2007