Πανεπιστήμιο Μακεδονίας

Τμήμα Εφαρμοσμένης Πληροφορικής

Μάθημα: Ανάκτηση Πληροφορίας και Μηχανές Αναζήτησης

Προγραμματιστική Εργασία 2

Θέμα: Κατηγοριοποίηση Κειμένου

Παρασκευή Ξανθοπούλου (it1490)

Καθηγητές: Γεωργία Κολωνιάρη, Γεώργιος Ευαγγελίδης

Ακαδημαϊκό Έτος: 2023-2024

Θεσσαλονίκη, Ιανουάριος 2024

Περιεχόμενα

1. Εισαγωγή

Η παρούσα μελέτη επικεντρώνεται στην ανάλυση συναισθημάτων μέσω της κατηγοριοποίησης κριτικών ταινιών ως θετικές ή αρνητικές, χρησιμοποιώντας κατηγοριοποιητές Naive Bayes. Το project περιλαμβάνει την εφαρμογή αυτών των κατηγοριοποιητών μέσω του λογισμικού RapidMiner, καθώς και την υλοποίησή τους στη γλώσσα προγραμματισμού Python.

Η εφαρμογή των κατηγοριοποιητών τόσο στο περιβάλλον RapidMiner όσο και με τη χρήση της Python, στοχεύει στην αξιολόγηση της απόδοσής τους με βάση το σύνολο δεδομένων "polarity_dataset_v2.0", το οποίο περιλαμβάνει 1000 θετικές και 1000 αρνητικές επεξεργασμένες κριτικές ταινιών.

2. Προεπεξεργασία

Η μεθοδολογία που ακολουθήθηκε περιλαμβάνει εκτεταμένες δοκιμές και βελτιστοποίηση, ειδικά για τα μοντέλα α1 (Naive Bayes με δυαδική εμφάνιση όρων) και α2 (Naive Bayes με εμφάνιση όρων) στο RapidMiner. Αφού προσδιορίστηκαν τα αποτελεσματικότερα βήματα προεπεξεργασίας για κάθε μοντέλο, μεταφέρθηκαν στο περιβάλλον Python για τα μοντέλα β1 (Multinomial Naive Bayes με εμφάνιση όρων) και β2 (Bernoulli Naive Bayes με δυαδική εμφάνιση όρων). Με αυτόν τον τρόπο, διασφαλίστηκε ότι τα ζευγάρια μοντέλων α1-β1 και α2-β2 εφαρμόζουν ακριβώς την ίδια προεπεξεργασία. Ακολουθεί ανάλυση των βημάτων προεπεξεργασίας που υιοθετήθηκαν για τη βελτιστοποίηση των μοντέλων.

2.1. Tokenization

Κατά την επιλογή της μεθόδου tokenization, δόθηκε έμφαση στη βελτιστοποίηση της επίδοσης. Και για τα δύο μοντέλα του RapidMiner (a1, a2), επιλέχθηκε η λειτουργία "Regular Expression" με το μοτίβο "\s", οδηγώντας την πλατφόρμα να χωρίσει το κείμενο με βάση τα κενά, τα tabs ή τους χαρακτήρες νέας γραμμής. Η απόφαση αυτή βασίστηκε στο ανώτερο performance σε σύγκριση με εναλλακτικές λύσεις, όπως non-letter ή linguistic tokenization, οι οποίες δοκιμάστηκαν. Στην Python, χρησιμοποιήθηκε η συνάρτηση word_tokenize από το Natural Language Toolkit (nltk), η οποία υλοποιεί μια μέθοδο tokenization παραπλήσια με την προσέγγιση που χρησιμοποιήθηκε στα μοντέλα του RapidMiner. Οι επιλογές των tokenizers για τα μοντέλα a1, a2, b1 και b2 παρουσίασαν καλύτερες επιδόσεις, επιβεβαιώνοντας την ανωτερότητα της επιλεγμένης προσέγγισης έναντι των εναλλακτικών λύσεων.

2.2. Generate N-Grams (Terms)

Τα n-grams (terms) καταγράφουν διαδοχικά μοτίβα λέξεων και βελτιώνουν την ικανότητα του μοντέλου να ανιχνεύει πληροφορίες σχετικά με το πλαίσιο (context) σε δεδομένα κειμένου. Η απόφαση να συμπεριληφθεί το βήμα "Generate n-grams (Terms)" εξαρτάται από τα χαρακτηριστικά του κάθε μοντέλου. Για το μοντέλο α1, η εισαγωγή των n-grams οδήγησε σε μείωση της επίδοσης, με αποτέλεσμα τον αποκλεισμό τους από τα βήματα προεπεξεργασίας. Αντίθετα, στο μοντέλο a2, η χρήση των n-grams συνέβαλε σε αύξηση της επίδοσης, δικαιολογώντας την ενσωμάτωσή τους. Αντίστοιχα εφαρμόστηκαν και στο μοντέλο β2 στην Python, εξασφαλίζοντας τη συνοχή στη μέθοδο προεπεξεργασίας του ζεύγους α2-β2. Ειδικότερα, η βέλτιστη επίδοση επιτεύχθηκε με n=3, επισημαίνοντας τη σημασία της επιλογής του κατάλληλου μεγέθους n-gram.

2.3. Negations

Τα negations, μια κρίσιμη πτυχή της ανάλυσης συναισθήματος, δημιούργησαν προκλήσεις κατά την εφαρμογή στο περιβάλλον του RapidMiner, επηρεάζοντας τις αποφάσεις οι οποίες λήφθηκαν. Ενώ το αρχικό σχέδιο προέβλεπε τη σήμανση κάθε λέξης μεταξύ του δείκτη άρνησης "n't" και της επακόλουθης στίξης, με "NOT_", αυτός ο περίπλοκος κανόνας αποδείχθηκε δύσκολο να εκτελεστεί αποτελεσματικά. Οι operators επεξεργασίας κειμένου του RapidMiner παρουσίασαν περιορισμούς στον χειρισμό του δυναμικού μετασχηματισμού της προσθήκης του "NOT_" σε κάθε ενδιάμεση λέξη. Δεδομένων αυτών των περιορισμών, υιοθετήθηκε μια ρεαλιστική προσέγγιση για την αντικατάσταση της κατάληξης "n't" με τη λέξη "not" σε όλο το κείμενο. Αυτή η εναλλακτική λύση αποδείχθηκε απλούστερη και πιο εφικτή στο πλαίσιο της ροής εργασίας του RapidMiner. Στα μοντέλα a1 και b1, αυτή η στρατηγική αντικατάστασης αύξησε οριακά την επίδοση, ενώ για τα μοντέλα a2 και b2, όπου δεν επηρέασε την επίδοση, κρίθηκε περιττή και παραλείφθηκε από την προεπεξεργασία.

2.4. Punctuation Removal

Ενώ τα σημεία στίξης έχουν σημαντικό ρόλο στη γλωσσική έκφραση, η παρουσία τους μπορεί να εισάγει θόρυβο και πλεονασμό στην επεξεργασία κειμένων. Η επιλογή της εξάλειψης των σημείων στίξης από το κείμενο παρέχει μια πιο καθαρή είσοδο για τη μετέπειτα ανάλυση. Αυτή η απόφαση είναι συνεπής σε όλα τα σενάρια, συμπεριλαμβανομένων των μοντέλων a1, a2, b1 και b2, όπου οδηγεί σε οριακή αύξηση της επίδοσης. Ο απώτερος στόχος είναι η βελτιστοποίηση της αποδοτικότητας των

μεταγενέστερων διαδικασιών, συμβάλλοντας στη συνολική βελτίωση της αποτελεσματικότητας της επεξεργασίας κειμένου.

2.5. Pruning

Το pruning (κλάδεμα), αφορά την αφαίρεση των όρων χαμηλής και υψηλής συχνότητας και αποτελεί απόφαση που αποσκοπεί στη βελτίωση της ποιότητας του συνόλου δεδομένων τόσο στα μοντέλα του RapidMiner όσο και στην υλοποίηση με Python. Χρησιμοποιώντας ένα κατώτατο όριο με βάση το ποσοστό, ο στόχος είναι να επιτευχθεί μια ισορροπία μεταξύ της διατήρησης ουσιαστικών όρων και του μετριασμού των ακραίων όρων που μπορεί να μην συμβάλλουν σημαντικά στη συνολική κατανόηση του κειμένου. Για το ζεύγος μοντέλων α1-β1, υιοθετήθηκε η προσέγγιση pruning με βάση το ποσοστό (percentual), με ελάχιστο κατώφλι 3,9% και μέγιστο κατώφλι 30%. Για το ζεύγος μοντέλων α2-β2, εφαρμόστηκε και πάλι ποσοστιαία περικοπή, με το ελάχιστο όριο το 3,9% και μέγιστο το 32%. Αυτή η προσαρμοσμένη προσέγγιση υιοθετήθηκε για τη βελτιστοποίηση της επίδοσης κάθε ζεύγους μοντέλων.

2.6. StopWord Removal και Stemming

Στο πλαίσιο της προεπεξεργασίας κειμένου, η αφαίρεση των StopWords συνήθως εφαρμόζεται για τη μείωση του θορύβου και τη βελτίωση της αποτελεσματικότητας, ωστόσο, στην ανάλυση συναισθήματος, η διατήρησή τους αποτελεί κρίσιμη επιλογή. Τα συναισθήματα συχνά ανιχνεύονται μέσω συγκεκριμένων λέξεων και φράσεων, και η αφαίρεση των StopWords μπορεί να οδηγήσει στην απώλεια κρίσιμων γλωσσικών ενδείξεων. Η απόφαση να μην συμπεριληφθεί το βήμα αφαίρεσης των StopWords βασίστηκε στις δοκιμές που πραγματοποιήθηκαν στα μοντέλα a1, a2, όπου διαπιστώθηκε ότι η διατήρηση των StopWords οδηγεί σε καλύτερες επιδόσεις.

Σε παράλληλη διερεύνηση, αποφασίστηκε να μην χρησιμοποιηθεί η τεχνική Stemming για τη μετατροπή των όρων στη βασική τους μορφή. Σε εργασίες όπως η ανάλυση συναισθήματος, η διατήρηση των αρχικών λέξεων είναι προτιμότερη. Αυτή η επιλογή επιτρέπει στο μοντέλο να διατηρήσει μια πιο σφαιρική κατανόηση της γλώσσας, συμβάλλοντας έτσι στη βελτίωση της ακρίβειας των αποτελεσμάτων. Σημειώνεται ότι η χρήση της τεχνικής Stemming οδηγεί σε μείωση της επίδοσης και στα δύο μοντέλα, a1 και a2, αναδεικνύοντας την αρνητική επίδρασή της στην ανάλυση συναισθήματος.

3. Αποτελέσματα

Οι τιμές των Precision και Recall κάθε μοντέλου υπολογίστηκαν με χρήση των τύπων:

- $Precision = \frac{TP}{TP + FP}$
- $Recall = \frac{TP}{TP + FN}$

3.1. Naive Bayes (RapidMiner) με Binary Term Occurrences (α1)

Βήματα προεπεξεργασίας:

- Replace Negations
- Tokenization
- Remove Punctuation
- Pruning mode: percentual (min 3.9%, max 30%)

Performance

Confusion Matrix		Actual Values	
		Positive	Negative
Predicted Values	Positive	773	159
	Negative	227	841

Accuracy: 80.70%

Precision: 82.94%

Recall: 77.30%

3.2. Naive Bayes (RapidMiner) με Term Occurrences (α2)

Βήματα προεπεξεργασίας:

- Tokenization
- Remove Punctuation
- Generate n-Grams (Terms)
- Pruning mode: percentual (min 3.9%, max 32%)

Performance

Confusion Matrix		Actual Values	
		Positive	Negative
Predicted Values	Positive	689	165
	Negative	311	835

Accuracy: 76.20%

Precision: 80.68%

Recall: 68.90%

3.3. Multinomial Naive Bayes (Python) με Term Occurrences (β1)

Βήματα προεπεξεργασίας:

- Replace Negations
- Tokenization
- Remove Punctuation
- Pruning mode: percentual (min 3.9%, max 30%)

Performance

Confusion Matrix		Actual Values	
		Positive	Negative
Predicted Values	Positive	807	191
	Negative	193	809

Accuracy: 80.80%

Precision: 80.86%

Recall: 80.70%

3.4. Bernoulli Naive Bayes (Python) με Binary Term Occurrences (β2)

Βήματα προεπεξεργασίας:

- Tokenization
- Remove Punctuation
- Generate n-Grams (Terms)
- Pruning mode: percentual (min 3.9%, max 32%)

Performance

Confusion Matrix		Actual Values	
		Positive	Negative
Predicted Values	Positive	710	175
	Negative	290	825

Accuracy: 76.75%

Precision: 80.23%

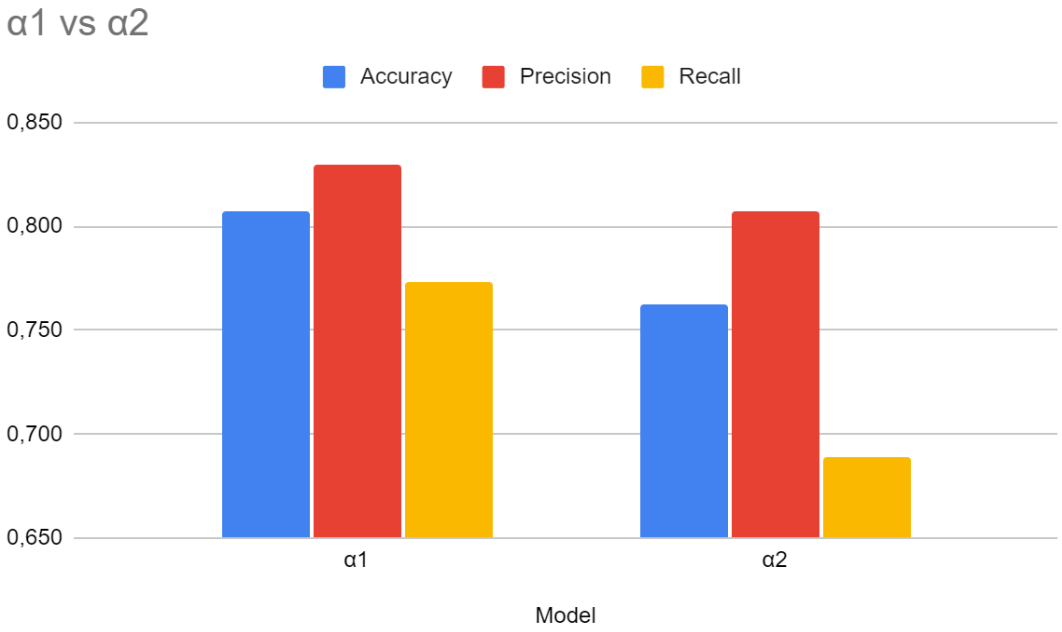
Recall: 71%

4. Σύγκριση Μοντέλων

4.1. α1 vs α2

Όπως παρατηρούμε στο γράφημα **4.1.1**, υπάρχει σημαντική διαφορά στην επίδοση των δύο μοντέλων, α1 και α2. Το μοντέλο α1, που χρησιμοποιεί διάνυσμα με δυαδικές εμφανίσεις

όρων (binary term occurrences), επιδεικνύει ανώτερο accuracy, precision, και recall σε σχέση με το μοντέλο α2, το οποίο χρησιμοποιεί διάνυσμα με το πλήθος εμφανίσεων όρων (term occurrences).

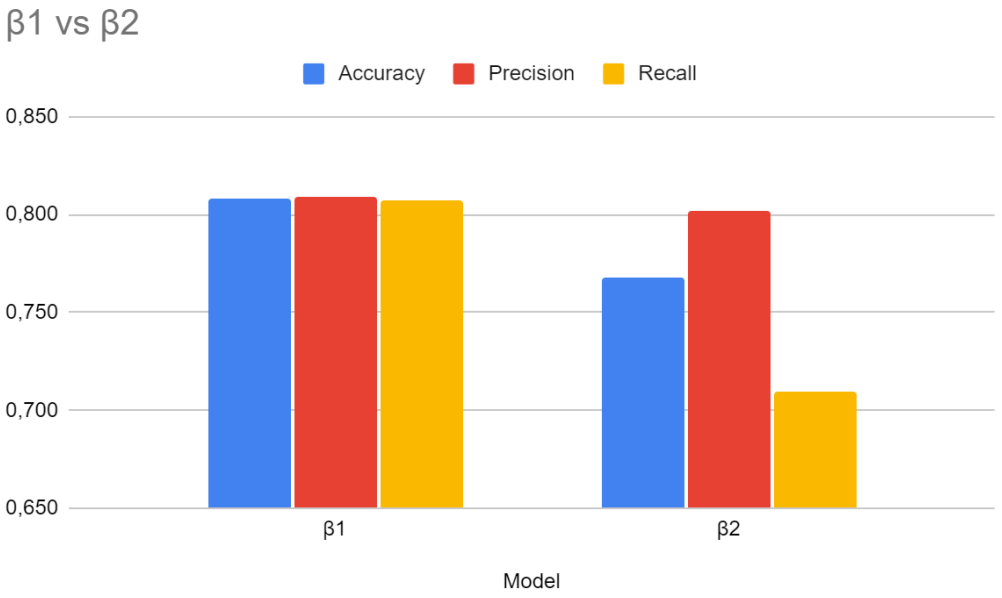


4.1.1. Σύγκριση επιδόσεων α1 vs α2

Η επιλογή των δυαδικών εμφανίσεων φαίνεται να αποφέρει υψηλότερη επίδοση στο μοντέλο α1. Η κυριότερη διαφορά μεταξύ δυαδικών εμφανίσεων και εμφανίσεων έγκειται στην αντίληψη της παρουσίας ή μη ενός όρου στο κείμενο. Στις δυαδικές εμφανίσεις, ενδιαφερόμαστε μόνο για το εάν ο όρος υπάρχει ή όχι, χωρίς να λαμβάνουμε υπόψη τη συχνότητα εμφάνισής του. Αυτό μπορεί να οδηγήσει σε καλύτερη επίδοση, καθώς η πληροφορία συχνότητας μπορεί να προκαλέσει θόρυβο και να περιπλέξει τη διαδικασία ανάλυσης συναισθημάτων.

4.2. β1 vs β2

Το μοντέλο β1 (Multinomial NB με χρήση term occurrences) φαίνεται να παρουσιάζει καλύτερη επίδοση σε σχέση με το μοντέλο β2 (Bernoulli NB με χρήση binary term occurrences), όπως φαίνεται και στο γράφημα **4.2.1**.

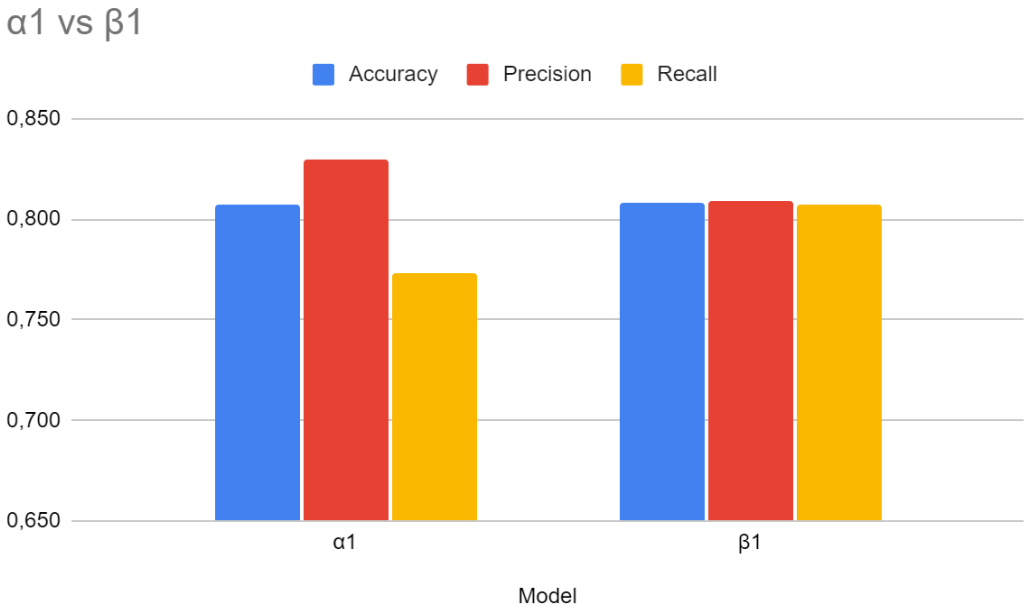


4.2.1. Σύγκριση επιδόσεων β1 vs β2

Το μοντέλο β1, συγκεκριμένα, εμφανίζει υψηλότερο Accuracy και Recall σε σύγκριση με το μοντέλο β2, ενώ το Precision είναι σχεδόν ίσο. Σε αντίθεση με την προηγούμενη σύγκριση μεταξύ α1 και α2, όπου το μοντέλο που χρησιμοποιεί δυαδικό διάνυσμα φάνηκε να έχει καλύτερη επίδοση, εδώ το διάνυσμα με εμφανίσεις όρων παρουσιάζει καλύτερη επίδοση. Αυτό υποδεικνύει πως η επίδοση των μοντέλων εξαρτάται από πολλούς παράγοντες. Είναι σημαντικό να σημειωθεί ότι το μοντέλο β1 ακολουθεί την ίδια διαδικασία προεπεξεργασίας με το μοντέλο α1, ενώ το μοντέλο β2 ακολουθεί τη διαδικασία του α2, και ότι τα βήματα προεπεξεργασίας επιλέχθηκαν μετά από ελέγχους προκειμένου να βελτιστοποιηθεί η επίδοση των μοντέλων α1 και α2, με άλλα λόγια είναι προσαρμοσμένα στα μοντέλα α1 και α2.

4.3. α1 vs β1

Η σύγκριση μεταξύ του μοντέλου α1 (Binary Term Occurrences) και του μοντέλου β1 (Term Occurrences) αποκαλύπτει μικρές διαφορές στην επίδοση. Το μοντέλο α1 παρουσιάζει ελαφρώς υψηλότερο precision σε σύγκριση με το μοντέλο β1, αλλά χαμηλότερο recall, όπως φαίνεται στο γράφημα **4.3.1**.

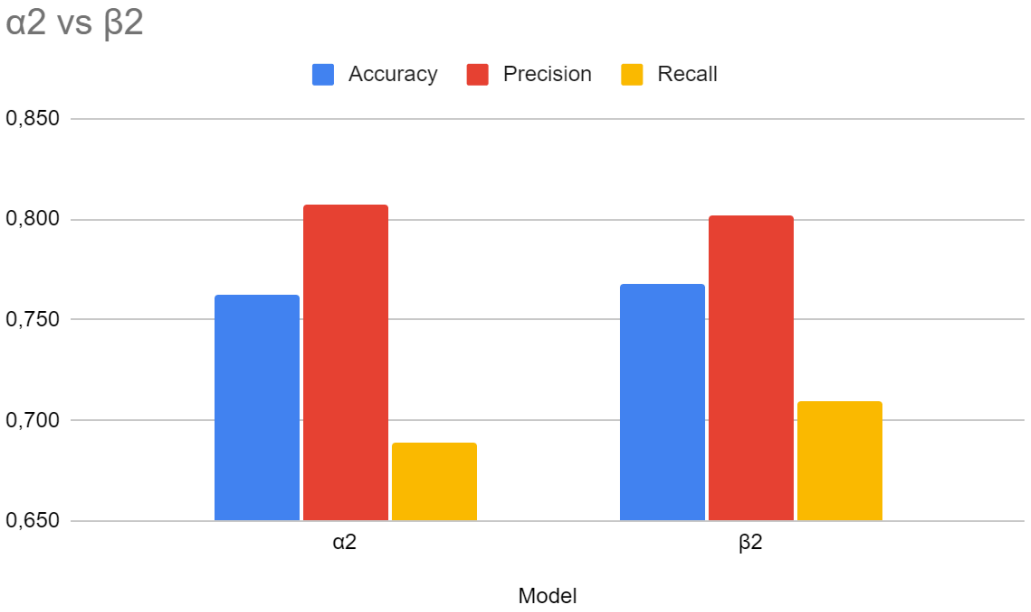


4.3.1. Σύγκριση επιδόσεων α1 vs β1

Ενδιαφέρον παρουσιάζει το γεγονός ότι, παρόλο που τα δύο μοντέλα χρησιμοποιούν διαφορετικούς τύπους εμφανίσεων όρων, η επίδοσή τους παραμένει παρόμοια. Αυτό υπογραμμίζει τη σημασία της προεπεξεργασίας δεδομένων και της επιλογής αλγορίθμου και παραμέτρων για την επίτευξη βέλτιστης επίδοσης στην ανάλυση συναισθημάτων.

4.4. α2 vs β2

Η σύγκριση μεταξύ του μοντέλου α2 (Term Occurrences) και του μοντέλου β2 (Binary Term Occurrences) αποκαλύπτει διαφορές στην επίδοση. Όπως φαίνεται στο γράφημα **4.4.1**, το μοντέλο α2 παρουσιάζει υψηλότερο precision σε σύγκριση με το μοντέλο β2, αλλά χαμηλότερο recall. Παρόλα αυτά, η διαφορά αυτή είναι ελάχιστη, και μπορεί να οφείλεται σε διάφορους παράγοντες, όπως η διαφορετική υλοποίηση των αλγορίθμων στα δύο περιβάλλοντα.

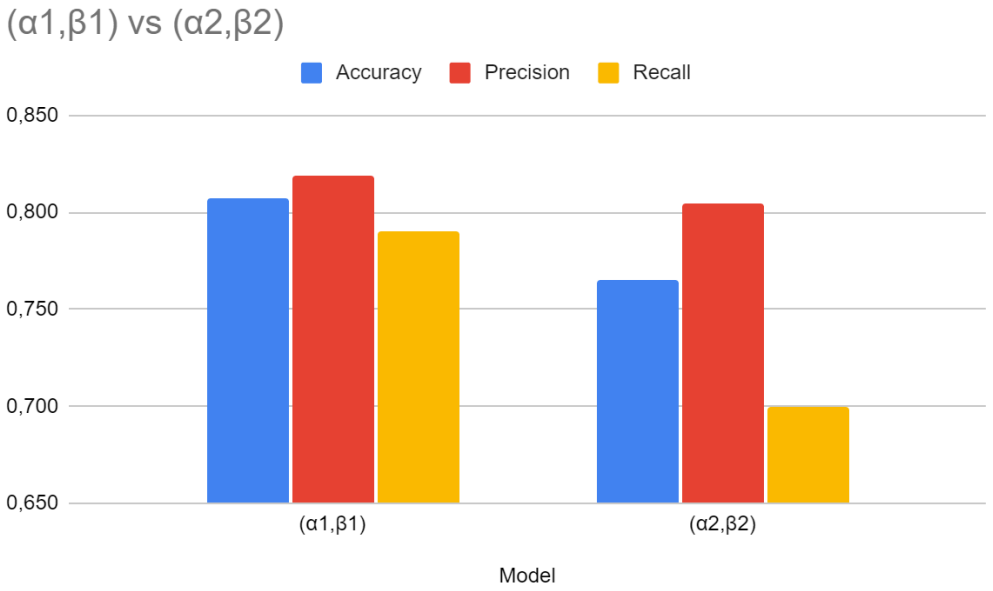


4.4.1. Σύγκριση επιδόσεων α2 vs β2

Μια ενδιαφέρουσα παρατήρηση είναι ότι, αν και το μοντέλο α2 χρησιμοποιεί διάνυσμα με το πλήθος εμφανίσεων όρων, ενώ το μοντέλο β2 χρησιμοποιεί δυαδικό διάνυσμα, η ακρίβεια του μοντέλου α2 είναι υψηλότερη. Αυτό μπορεί να οφείλεται στις διαφορετικές απαιτήσεις των δύο αλγορίθμων, καθώς η χρήση δυαδικού διανύσματος μπορεί να περιλαμβάνει λιγότερη πληροφορία από τον όγκο των εμφανίσεων, ενώ ταυτόχρονα μπορεί να βοηθά στη διακριτική αναγνώριση των όρων. Συνολικά, η σύγκριση μεταξύ των μοντέλων α2 και β2 αναδεικνύει ότι η επίδοση των μοντέλων εξαρτάται από πολλούς παράγοντες, συμπεριλαμβανομένου του τύπου του διανύσματος χαρακτηριστικών και των ιδιαιτεροτήτων του προβλήματος. Η κατάλληλη επιλογή μοντέλου και προεπεξεργασίας δεδομένων παραμένει κρίσιμη για την επιτυχή ανάλυση συναισθημάτων.

4.5. (α1,β1) vs (α2,β2)

Στο γράφημα **4.5.1** μπορούμε να δούμε τη σύγκριση του μέσου όρου των μετρικών των ζευγών (α1,β1) και (α2,β2), και παρατηρούμε ότι συνολικά το ζεύγος (α1,β1) έχει ανώτερη επίδοση.



4.5.1. Σύγκριση επιδόσεων ζευγών (α1,β1) vs (α2,β2)

Οι διαφορετικοί τύποι εμφανίσεων όρων για κάθε μοντέλο (δυαδικές για το α1 και term occurrences για το β1) φαίνεται ότι οδηγούν σε πιο αξιόπιστη και αποτελεσματική ανάλυση συναισθημάτων σε σχέση με το ζεύγος (α2, β2). Η διαφοροποίηση αυτή ενισχύει την ιδέα ότι η επιλογή του κατάλληλου τύπου εμφάνισης όρων για κάθε μοντέλο είναι κρίσιμη για την επίτευξη βέλτιστων αποτελεσμάτων στην ανάλυση συναισθημάτων.

5. Συμπεράσματα

Συνολικά, μπορούμε να συμπεράνουμε ότι:

- Η επιλογή μεταξύ δυαδικών εμφανίσεων όρων και term occurrences εξαρτάται από τη φύση του προβλήματος. Σε περιπτώσεις όπου η συχνότητα εμφάνισης των όρων είναι σημαντική, η χρήση term occurrences μπορεί να παρέχει περισσότερη πληροφορία. Αντίθετα, σε προβλήματα όπου μας ενδιαφέρει μόνο η παρουσία ή μη ενός όρου, οι δυαδικές εμφανίσεις μπορεί να οδηγήσουν σε αποτελεσματικότερη ανάλυση.
- Η επίδοση των μοντέλων επηρεάζεται από πολλούς παράγοντες, όπως η επιλογή του αλγορίθμου, ο τύπος του διανύσματος χαρακτηριστικών, αλλά και ιδιαιτερότητες του προβλήματος. Για παράδειγμα, η φύση των κειμένων μπορεί να απαιτεί προσαρμογή στην προεπεξεργασία των δεδομένων.
- Η στοχευμένη προεπεξεργασία των δεδομένων αποτελεί κρίσιμο βήμα για την επιτυχή ανάλυση συναισθημάτων. Η προσαρμοσμένη προεπεξεργασία σε συνδυασμό

με τον κατάλληλο αλγόριθμο βοηθά στη βελτιστοποίηση της επίδοσης, λαμβάνοντας υπόψη τη φύση και τις απαιτήσεις του συγκεκριμένου προβλήματος.

Κλείνοντας, η επιτυχής ανάλυση συναισθημάτων απαιτεί προσεκτική επιλογή των κατάλληλων τύπων εμφανίσεων όρων, σε συνδυασμό με τη στοχευμένη προεπεξεργασία δεδομένων και τη σωστή επιλογή αλγορίθμου. Η εξειδικευμένη προσέγγιση στη χειρισμό του συγκεκριμένου προβλήματος ενισχύει την ακριβή ανάλυση των συναισθημάτων.