

Γιαννακοπούλου Σταυρούλα – 3220027

Σταμαδιάνου Μαρία – 3220194

## Αναφορά για το Μέρος Β της 1<sup>ης</sup> εργασίας στην Τεχνητή Νοημοσύνη

Ο κώδικας ForwardChaining υλοποιεί εξαγωγή συμπερασμάτων προς τα εμπρός για οριστικές προτάσεις Horn προτασιακής λογικής και συγκεκριμένα για τύπους χωρίς άρνηση. Λαμβάνει ως είσοδο από τον χρήστη ένα αρχείο txt που περιλαμβάνει τη βάση γνώση, και ένα premise προς απόδειξη.

Για την υλοποίηση αυτής της διαδικασίας δημιουργήθηκαν τα εξής:

### Clause:

Η κλάση αυτή αναπαριστά έναν κανόνα (ή πρόταση) της βάσης γνώσης. Κάθε κανόνας αποτελείται από: α)Μια **υπόθεση** (premise): Λίστα γεγονότων που πρέπει να ισχύουν, β)Μια **συμπερασματική πρόταση** (conclusion): Το γεγονός που συμπεραίνεται αν ισχύουν όλες οι υποθέσεις.

### isEntailed():

Αυτή η μέθοδος ελέγχει αν ένα συγκεκριμένο ερώτημα μπορεί να εξαχθεί από τη βάση γνώσης χρησιμοποιώντας την εξαγωγή συμπερασμάτων προς τα εμπρός.

Χρησιμοποιεί τα εξής εργαλεία:

- **count**: Χάρτης που αντιστοιχεί κάθε κανόνα στον αριθμό των υποθέσεων που παραμένουν να επιβεβαιωθούν.
- **agenda**: Ουρά που περιέχει τα γεγονότα προς επεξεργασία.
- **induced**: Χάρτης που παρακολουθεί αν ένα γεγονός έχει ήδη εξαχθεί.
- **p\_premise**: Αντιστοιχίζει κάθε γεγονός στους κανόνες που το περιλαμβάνουν στις υποθέσεις τους.

### Διαδικασία:

1. Για κάθε κανόνα, υπολογίζεται το πλήθος των υποθέσεων και ενημερώνονται τα σχετικά Maps ενώ οι κανόνες με κενή λίστα υποθέσεων προστίθενται στην agenda.
2. Κατα την εκτέλεση του αλγορίθμου γίνεται επεξεργασία των γεγονότων στην agenda. Αν το γεγονός ταιριάζει με την υπόθεση προς εξέταση, επιστρέφεται true. Για κάθε νέο γεγονός που επιβεβαιώνεται, ενημερώνεται η κατάσταση των σχετικών κανόνων. Αν όλες οι υποθέσεις

ενός κανόνα ικανοποιηθούν, το συμπέρασμα του κανόνα προστίθεται στην agenda.

3. Ο κώδικας επεξεργάζεται αρνητικά γεγονότα, όπως “-A”, μεταφράζοντάς τα σε false για το γεγονός “A” και προσπαθεί να κάνει ανίχνευση αντιφάσεων η οποία γίνεται όταν ένα γεγονός και το αρνητικό του είναι ταυτόχρονα αληθή.
4. Αν εξαντληθούν όλα τα γεγονότα χωρίς να εξαχθεί το ερώτημα, επιστρέφεται false.

**readkb():**

Αυτή η μέθοδος διαβάζει τη βάση γνώσης από το txt αρχείο το οποίο περιλαμβάνει τους κανόνες σε μια πολύ συγκεκριμένη μορφή που ορίζεται ως : premise1, premise2, ... -> conclusion. Έπειτα, διαχωρίζει τις υποθέσεις από το συμπέρασμα και δημιουργεί αντικείμενα Clause και τα προσθέτει στη λίστα της βάσης γνώσης.

**Main():**

Αυτή η μέθοδος διαβάζει τη βάση γνώσης από αρχείο που δίνεται ως όρισμα στη γραμμή εντολών, ελέγχει αν το ερώτημα συνεπάγεται από τη βάση γνώσης, χρησιμοποιώντας τη μέθοδο isEntailed και τελικά εμφανίζει το αποτέλεσμα.

---

Ο κώδικας ForwardChainingFOL υλοποιεί εξαγωγή συμπερασμάτων προς τα εμπρός για οριστικές προτάσεις Horn πρωτοβάθμιας κατηγορηματικής λογικής και σε αυτή την περίπτωση οι τύποι προς απόδειξη δεν περιλαμβάνουν άρνηση. Το πρόγραμμα διαβάζει δεδομένα (γεγονότα και κανόνες) από ένα αρχείο txt και προσπαθεί να συμπεράνει αν μπορεί να προκύψει ένας συγκεκριμένος στόχος μέσω του forward chaining. Όπως και πριν, λαμβάνει ως είσοδο από τον χρήστη ένα αρχείο txt που περιλαμβάνει τη βάση γνώση, και ένα premise προς απόδειξη.

Στις μεθόδους unify() και forwardChaining(), γίνεται έλεγχος αν μια πρόταση ή ένα γεγονός είναι αρνητικό, αφαιρώντας το πρόθεμα “-“ για την επεξεργασία του πυρήνα της πρότασης και στη συνέχεια το επαναφέρει στην σωστή του μορφή.

**Main():**

Η μέθοδος αυτή είναι το σημείο εκκίνησης της εφαρμογής. Αναλαμβάνει να διαβάσει την είσοδο από τα ορίσματα της γραμμής εντολών, να φορτώσει τη

βάση γνώσεων από αρχείο txt, να διαχωρίσει γεγονότα και κανόνες, και να καλέσει τον αλγόριθμο προωθημένης αλυσιδωτής συλλογιστικής.

Εκτελεί τις εξής ενέργειες: ελέγχει ότι υπάρχουν δύο είσοδοι (αρχείο και ερώτημα), καλεί τη μέθοδο `readKB` για να διαβάσει τη βάση γνώσεων, διαχωρίζει τους κανόνες και τα γεγονότα και καλεί τη μέθοδο `forwardChaining` για να αποφασίσει αν το ερώτημα μπορεί να προκύψει.

Σχέσεις με άλλες μεθόδους:

- Χρησιμοποιεί τη `readKB()` για να φορτώσει δεδομένα.
- Καλεί τη `forwardChaining()` για να εκτελέσει την κύρια λογική.

#### **readKB():**

Διαβάζει τη βάση γνώσεων από το αρχείο που δίνεται ως είσοδος με χρήση `BufferedReader` και επιστρέφει λίστα γραμμών, αγνοώντας κενές γραμμές.

#### **forwardChaining():**

Υλοποιεί τον αλγόριθμο εξαγωγής συμπερασμάτων προς τα εμπρός. Ελέγχει αν το ερώτημα μπορεί να παραχθεί από τα γεγονότα και τους κανόνες.

- **Κύριες Ενέργειες:**

1. Επαναληπτικά προσπαθεί να συνάγει νέα γεγονότα από τους κανόνες.
2. Χρησιμοποιεί τη μέθοδο `findAllUnifications()` για να βρει ενοποιήσεις μεταξύ γεγονότων και υποθέσεων κανόνων.
3. Ενημερώνει τα σύνολα γεγονότων και παραγόμενων συμπερασμάτων.
4. Καλεί τη `replacements()` για αντικαταστάσεις μεταβλητών

#### **findAllUnifications():**

Επιστρέφει όλους τους δυνατούς χάρτες ενοποιήσεων για ένα σύνολο υποθέσεων σύμφωνα με τα διαθέσιμα γεγονότα.

- **Κύριες Ενέργειες:**

1. Καλεί αναδρομικά την `prem_f_unification()` για να εξετάσει όλες τις ενοποιήσεις.
2. Επιστρέφει μια λίστα χαρτών με επιτυχημένες ενοποιήσεις.

### **prem\_f\_unification():**

Αναδρομική μέθοδος που εξετάζει όλες τις πιθανές ενοποιήσεις για ένα σύνολο υποθέσεων.

- **Κύριες Ενέργειες:**

1. Ελέγχει αν όλες οι υποθέσεις έχουν ενοποιηθεί.
2. Χρησιμοποιεί τη unify() για να ενοποιήσει μεμονωμένες υποθέσεις.
3. Προσθέτει επιτυχείς ενοποιήσεις στη λίστα.

### **Unify():**

Προσπαθεί να ενοποιήσει μια πρόταση με ένα γεγονός μέσω αντικαταστάσεων.

- **Κύριες Ενέργειες:**

1. Ελέγχει το πρόθεμα και τα ορίσματα της πρότασης και του γεγονότος.
2. Διαχειρίζεται μεταβλητές και σταθερές.
3. Ενημερώνει τον χάρτη αντικαταστάσεων.

### **Replacements():**

Αντικαθιστά όλες τις εμφανίσεις μεταβλητών στη συμβολοσειρά με τις αντίστοιχες τιμές τους από τον χάρτη αντικαταστάσεων.

**is\_Var():** Ελέγχει αν ένας όρος είναι μεταβλητή