

СИСТЕМА МАШИННОГО ПЕРЕКЛАДУ З ОДНІЄЇ МОВИ ВИСОКОГО РІВНЯ НА ІНШУ

Метою нашої роботи є спрощення розробки мультиплатформлених застосунків за допомогою використання системи машинного перекладу з однієї мови високого рівня на іншу, таку систему назовемо транскompілятор. Транскompілятор приймає код програми, написаної на мові програмування в якості вхідних даних і виконує еквівалентний вихідний код на іншій мові програмування. Наведемо вимоги до системи машинного перекладу: простота та зручність використання, коректність перекладу з мінімальним втручанням людини, можливість розширювати функціонал на інші мови програмування, достатній рівень швидкодії, прийнятний рівень вимог до апаратних ресурсів.

Окрім того для процесу розробки мультиплатформених застосунків є особливі вимоги, як то:

- можливість роботи системи на різних платформах – оскільки розробка таких застосунків зазвичай ведеться на багатьох операційних системах – Linux, Windows, Mac OS X;

- можливість «безшовної» інтеграції з існуючим кодом – неприпустимо щоб система транскompіляції перезаписала або якось змінила важливі частини вже існуючого коду розробленого інженерами;

- можливість двонаправленої трансляції, тобто з мови А на мову Б, а потім з мови Б на мову А;

- можливість масштабування та ітеративної компіляції – з часом кодова база зростає та процес транскompіляції може займати значний час, щоб уникнути проблем в майбутньому потрібно врахувати можливість розподіленого запуску системи на різних комп'ютерних системах зв'язаних мережею.

Неможливо безпосередньо зробити переклад з однієї мови на іншу та отримати результуючий працюючий код, так як різні мови програмування мають різну структуру проєктів та наповнення файлів з програмним кодом. Тому цей процес має відбуватись поетапно та наведено на рисунку 1.

Спочатку відбувається обробка початкового коду на мові високого рівня та генерується його опис на метамові за допомогою нейронної мережі. Потім формується абстрактне синтаксичне дерево з опису на метамові. Воно має наступну ієрархію рівнів: простір імен; рівень класів, інтерфейсів; рівень

методів та полів; рівень аргументів та логіки. Для навчання нейронної мережі можна використати Project CodeNet від IBM.

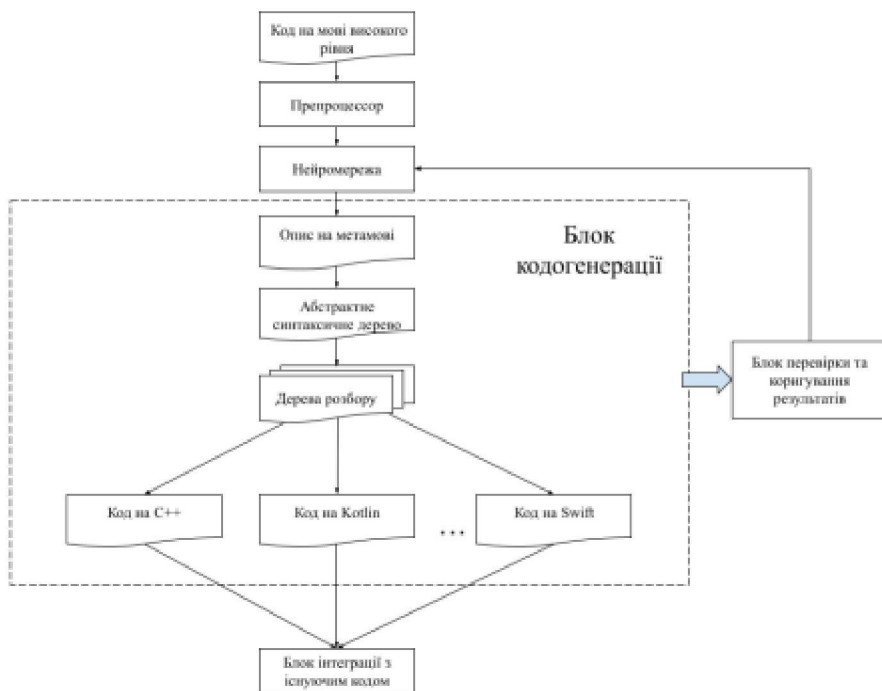


Рисунок 1 – Діаграма роботи системи машинного перекладу

Далі за правилами потрібної мови високого рівня відбувається трансляція та побудова вихідного дерева розбору, яке має наступні рівні: розбиття на файли; рівень просторів імен, класи та інтерфейси, рівень методів та полів; рівень аргументів та логіки.

Перевірка та коригування результатів допомагає з кожним разом все краще обучати нейронну мережу та поліпшувати результати машинного перекладу коду.

На останньому етапі відбувається побудова вихідного коду за вихідним синтаксичним деревом та запис його в файли. Нас цікавить можливість генерації вихідного опису на кілька мов високого рівня для автоматизації та спрощення написання застосунків для різних платформ. Блок інтеграції з існуючим кодом має автоматично або при мінімальному втручанні програмістів додавати згенерований код в існуючі частини проекту. Цей

функціонал базується на можливостях розподіленої системи керування версіями файлів та спільної роботи Git.