BTW, I can just suggest to make a couple of experiments:
1) try to log your queries with something like (but about your ms sql server)

sqlite3.Open(db)
#ifdef _DEBUG
    sqlite3.SetTrace(); //logging queries
#endif
& without Commit & Rollback... AFAIK, such command is being self-rollbacked automatically in the case of error or commited automatically if no errors - in ms sql server - but I have no chance to check it in ms sql server
// initialize transaction
BEGIN TRAN;
// transaction
INSERT INTO sales
VALUES ('7896', 'JR3435', 'Oct 28 1997', 25, 'Net 60', 'BU7832');
2) I'm not sure, that having duplicated PKs the query starts anyway, therefore I'm not sure that you have the right to do Rollback when this error arrised...
therefore you can try just SQL-statement to .Execute, expressing your sql-command in sql-string, not u++ method Insert... something like this - that is doing rollback ONLY IF @@TRANCOUNT > 0
BEGIN TRY
    BEGIN TRANSACTION
        exec(@sqlHeader)
    COMMIT
END TRY
BEGIN CATCH
    IF @@TRANCOUNT > 0
        ROLLBACK
END CATCH
or check this count in U++ somehow, before rollback - though it seems meaningless in the case of error... If you'd had several queries - you'd had something to rollback if previous queries already done... but in a single Insert failed - nothing to Rollback... imho