Yassir technical test

Implementation of the paper : Bi-LSTM and Ensemble based Bilingual Sentiment Analysis for a Code-mixed Hindi-English Social Media Text

Article : https://ieeexplore.ieee.org/document/9342241

Hello !

In this presentation we're going to talk about:

- The article problem statement
- The AI models proposed in the article
- How we reproduced the results
- Conclusion and possible improvement

Article problem statement

The article tackles the problem of **mixed-code language sentiment analysis** using artificial intelligence. A mixed-code language is when multilingual speakers switch between languages while communicating informally, which is very common on social medial.

The fact that the languages are mixed makes it a bit more difficult to extract sentiment and the conventional techniques designed for a single language **don’t provide satisfactory results** for such texts.

In the article the authors worked on **sentiment classification** for one of the most common code-mixed language pairs in India : **Hindi-English**

The AI models proposed in the article

Ensemble based model

After trying different types of classifiers the author kept the 4 best performing ones. Those classifiers are then ensembled to make one better performing classifier that uses soft voting to generate the final decision. The final model is passed through a Grid search (to tune the parameters)with Stratified 3 fold cross-validation.

Bidirectional LSTM based model

BiLSTMs is a Neural Network model that consists of two LSTMs : one taking the input in a forward direction, and the other in a backwards direction. Which effectively increase the amount of information available to the network, improving the context available to the algorithm (e.g. knowing what words immediately follow and precede a word in a sentence).

Steps to reproduce the results

STEP 1 | **STEP 2** | **STEP 3** | **STEP 4**

Data Analysis

Data Cleaning

Models building

Testing and Evaluation

Steps to reproduce the results

STEP 1

Data Analysis

STEP 2

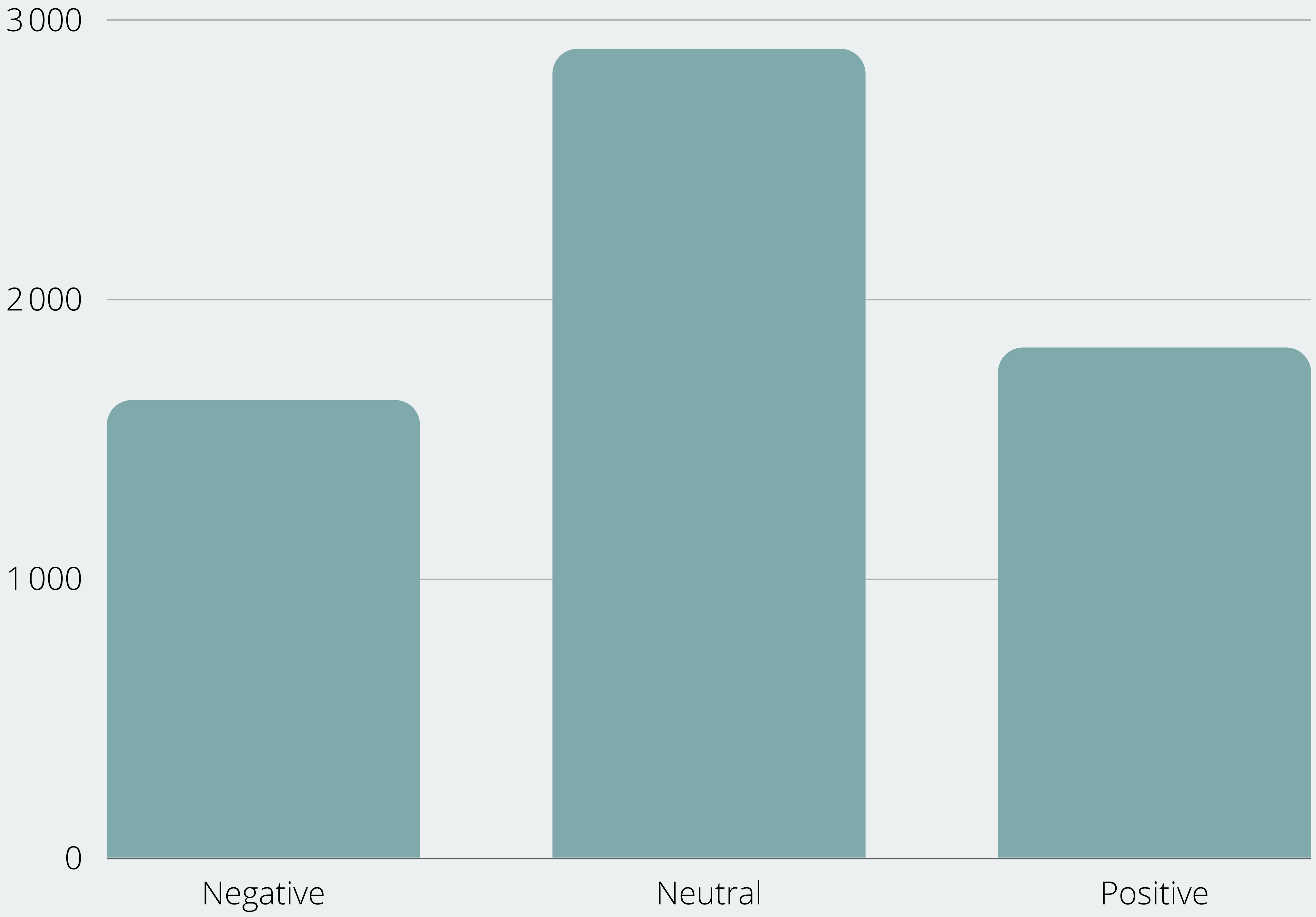
Data Cleaning

STEP 3

Models building

STEP 4

Testing and Evaluation



Labels distribution

There are more data in the **Neutral** class which means we can expect the models to be bias towards it. The total number of data is about 6000, maybe with more data we could achieve better results.

STEP 2

Data Cleaning

03 **Lemmatisation (match the words in the text with pre defined vocabulary)**

02 **Remove stop words (common words that don't add much meaning to the text)**

01 **Remove special characters and lower text**

NB : Lists of stop words and vocabulary were extracted from the github mentioned in the article

Steps to reproduce the results

STEP 1

STEP 2

STEP 3

STEP 4

Data Analysis

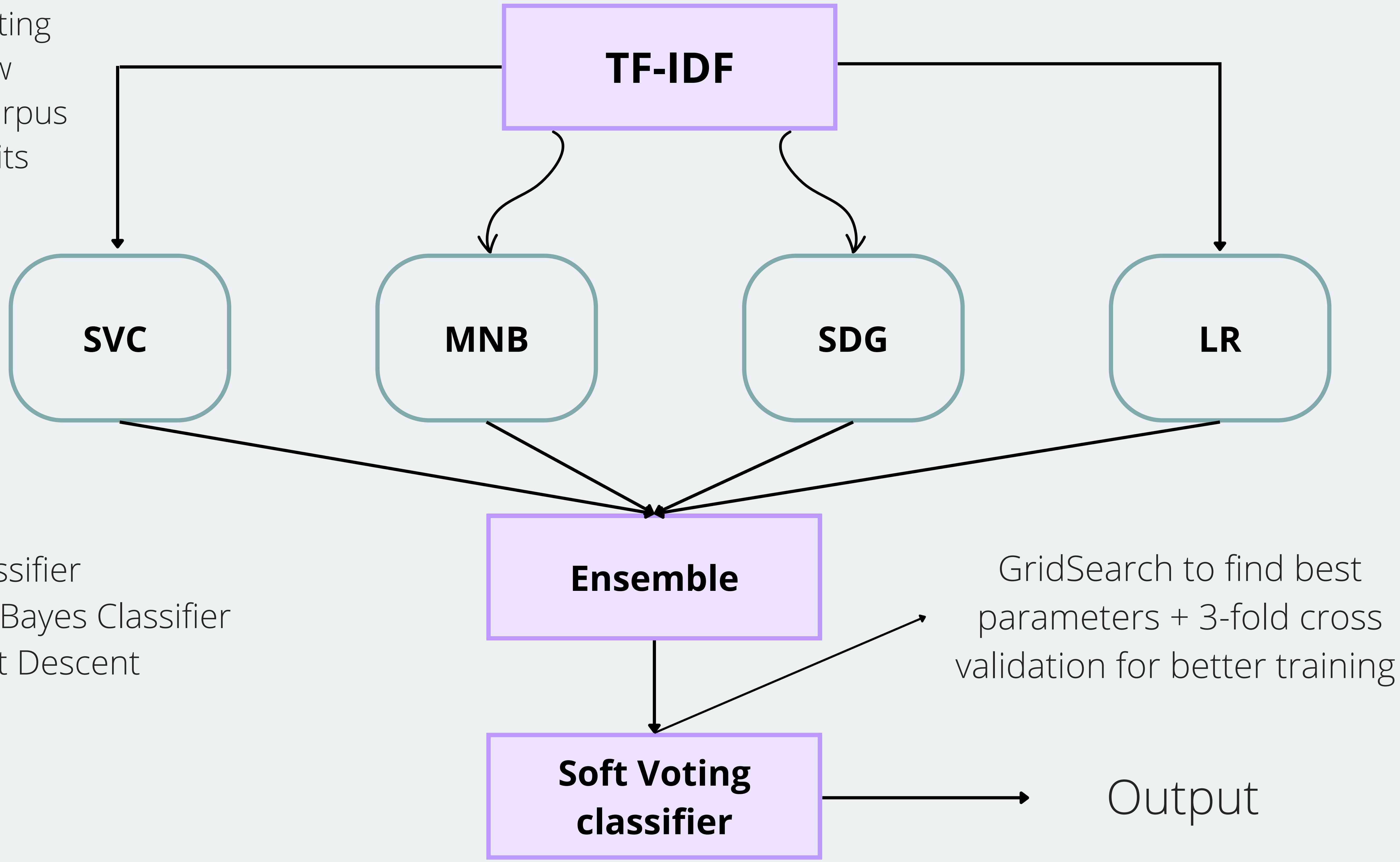
Data Cleaning

Models building

Testing and Evaluation

Ensemble based model

TF-IDF : Used as a weighting factor that reflects how important a word is to a corpus of words depending on its appearance



SVC: Support Vector Classifier
MNB: Multimodel Naive Bayes Classifier
SGD: Stochastic Gradient Descent
LR: Logistic regression

Bidirectional LSTM based model

tokenisation layer 01 — Creation of the model vocabulary and transformation of the text into a "hot-one" vector

Embedding layer 02 — Projection of the "hot-one" vectors into a smaller space

Bidirectional LSTM layers 03 — Type of neural network widely used for sequential data thanks to their ability to memorize important information from a sequence

Dense layers 04 — Performs a linear transformation that we use to find the predicted classes

Dropout layers 05 — Random deactivation of neurons to avoid over-fitting

Model training

85% of the data for training

15% of the data for testing

Ensemble Model

80% of the data for training

10% for validation

10% for testing

Bi LSTM Model

ÉTAPE 1

ÉTAPE 2

ÉTAPE 3

ÉTAPE 4

ÉTAPE 5

Extractions des données

Analyse de données

Nettoyage de données

Selection et entrainement des models

Test et evaluation

Classes/Metrics	Precision	Rappel	Score-F1	Support
Negative	0.58	0.54	0.55	256
Neutral	0.60	0.72	0.66	438
Positive	0.70	0.52	0.60	260
Average	0.62	0.62	0.61	Sum=954

Ensemble Model

Classes/Metrics	Precision	Rappel	Score-F1	Support
Negative	0.55	0.51	0.53	177
Neutral	0.61	0.62	0.61	289
Positive	0.52	0.54	0.53	170
Average	0.57	0.57	0.57	sum=636

Bi LSTM Model

Conclusion

- We were able to achieve precision, recall, and F1 score of 59 % for the Bi LSTM model and about 61% with the ensemble model and as expected both models were bias towards the "Neutral" class.

- Further improvements can be made by : Experimenting with different parameters of the models, better preprocessing, higher ressources and more data.

- Self-attention based models can also be used in this cases and could perform better.

- The results of the paper don't seem adapted to the dataset used (?)