Ryerson University
Data Science & Analytics Graduate Studies

2018 M.Sc. in Data Science and Analytics

APPLICATIONS OF DEEP LEARNING AND PARALLEL PROCESSING FRAMEWORKS IN DATA MATCHING

Bernardo Najlis **Supervisor: Dr. Raahemifar**

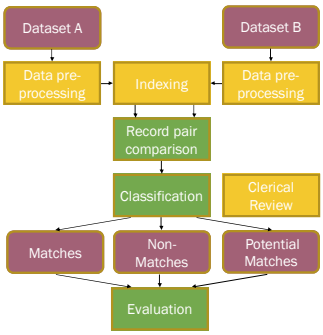
Objective

Data Matching (also known as Record Linkage, Entity Resolution or Duplicate Detection) is a prevalent problem in all disciplines that use data: if a data record (database or data frame row, line on a data file) is a data representation of an entity that exists in the real world (i.e. a person, object or event) how can we determine when and which multiple data records represent the same real-world entity?

This research work uses modern neural network techniques to create a model that given a pair of records that can be a potential data match (i.e. represent the same real-world entity) returns a highly accurate result.

Background

The main reference and consultation book for traditional Data Matching (Christen, 2012) describes the following traditional pipeline:



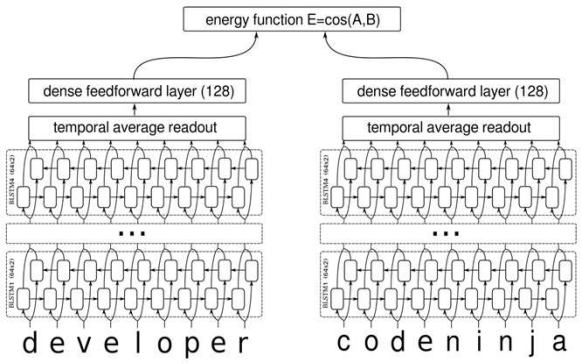
Source: Data Matching-Concepts and Techniques for Record Linkage, Entity Resolution and Duplicate Detection (Christen, 2012)

Our approach simplifies most of the *Data pre-processing* step, and replaces the *Record Pair comparison*, *Classification* and *Evaluation* from the traditional Machine Learning classifier models (i.e. n-gram score matching) with a **Siamese Bidirectional LSTM network** that calculates the similarity between two input records.

Indexing can still used for optimization and parallelization, *Clerical Review* provides a control step to review an feed manually labelled data during re-training.

Methodology

LSTMs are an improvement over Recurrent Neural Networks used to remember information for long periods of time as their default behaviour. By making their input layer bidirectional we provide an additional reversed version of the characters that optimize pattern recognition tasks.



Source: Learning Text Similarity with Siamese Recurrent Networks. Paul Neculoiu, Marteen Vesteegh and Mihai Rotaru, 2016

The Siamese arrangement creates two networks, both using the same set of weights that are connected at their outputs by a common loss function. In our case we use a cosine distance function to calculate the similarity between the vector representation provided by each sub network.

Results

The dataset used for our experimentation was extracted from the FEBRL paper, and it comprises 10,000 individual medical records with patient names, addresses. After arranging the original data to provide a balanced dataset with positive and negative match values, we experimented with different possible network parameter values.

- **Embedding Dimension Size**: This is determined by the maximum size of any input record in the dataset, and set to 103.
- **Hidden Units**: Has a direct correlation with the complexity and expressiveness the model has to generate the target function. Values higher than 300 do not improve results, set to 300.
- **Batch Size**: Number of data points processed in group and used to recalculate network weights, relative to training set number of rows. Higher values degrade model performance, set to 121.
- **Dropout Keep Probability**: Helps normalizing data distribution, selecting random nodes to be dropped out of the network. Traditionally set to 0.5, better performance found at 0.25.
- **L2 Regularization Lambda**: Penalizes coefficients and reduces potential overfitting, ranges from 1.0 to 0.0. Experimentation proves better performance with 0.0 (no regularization required).

$$Cost = Loss + \frac{\lambda}{2 * m} * \sum \|w\|$$

Iterating over multiple combination of network parameters to find the following optimal set of values:
This lead to an **Accuracy of 95.94%** for validation data set and an **Accuracy of 97.94%** for the test data set.

Conclusions

Deep Neural Networks (more specifically a Siamese BLSTM network) can effectively be used as a method for Data Matching / Record Linkage.

The deep learning approach removes the complexity of tuning multiple parameters, and heavy customization required with the traditional Data Matching / Linkage approach (i.e.: an n-gram score matching). It also provides a generic engine that can be used across any type of record input, as long as it is fed as a concatenated string representation of its characters. This also alleviates tasks like reference table based name synonym substitution (i.e. “Bob” = “Robert”), address disambiguation and cleanup (i.e. “Rd” = “Road”), as all these features are learned by the network based on the training data set.