Paraphrase detection

FIB | Facultat d’Informàtica de Barcelona

Introduction to human language tecnology

josep famadas alsamora / jordi riu vicente

2017

Contenido

[Table of Figures 2](#_Toc500760078)

[1. Introduction 3](#_Toc500760079)

[2. Part 1 4](#_Toc500760080)

[2.1. Metrics 4](#_Toc500760081)

[2.1.1. Phrasal overlap measure 4](#_Toc500760082)

[2.1.2. IDF overlap measure 4](#_Toc500760083)

[2.1.3. TF-IDF cosine similarity 5](#_Toc500760084)

[2.2. Results 6](#_Toc500760085)

[3. Part 2 9](#_Toc500760086)

[4. Bibliography 9](#_Toc500760087)

# Table of Figures

[Figure 1: Accuracy, Precision and Recall of our approaches 4](#_Toc498097820)

[Figure 2: Accuracy, precision and recall of our approaches using lemmas 5](#_Toc498097821)

[Figure 3: Colormap comparing the Accuracy, Precision and Recall of all our approaches 6](#_Toc498097822)

# Introduction

In this document we discuss many different approaches to detect paraphrases in pairs of sentences, i.e. if two sentences say the same but using different words.

In order to check the performance of these approaches we train a logistic regression function for each of them using a training set of 4.076 pairs of sentences properly classified as paraphrase or not.

Using each one of the calculated functions we compute the Accuracy, Precision and Recall of its corresponding approach with a test set of 1.725 pairs of sentences also classified as paraphrase or not.

The project is divided in two parts, the first has already been delivered and we add now the second one. In the second part we add new approaches using the syntactic dimension (which was not used in the first part) combined with the lexical one.

Finally, we combine all the different approaches from both parts of the whole project and compare them.

# Part 1

For each one of the approaches we have chosen a different sentence similarity metric.

## Metrics

### Phrasal overlap measure

For the first approach we have selected the phrasal overlap measure. Due to the fact that simple word overlap, which consists of counting the words that are repeated in both sentences, is not a pretty accruable metric, what phrasal overlap does is counting n-words phrases that are repeated in both sentence, being n from 1 to the shortest sentence length, and adding to the overlap value the current ‘n’ squared. The process if fully explained in [1].

Finally, to normalize so the metric returns a value from 0 to 1 we apply the normalization proposed by Ponzetto and Strube in [2].

### IDF overlap measure

For this second approach we have decided to use not the words but also their ‘importance’ in the meaning, being this given by the invers document frequency (IDF), which is inversely proportional to the appearing frequency of this word in a set of documents.

In this approached, we have used as documents all the 8.152 individual sentences of the training set. Being IDF = 1 if a word appears in every sentence and 10.006 if it appears only in 1 sentence.

The similarity for each pair of sentences has been computed as a simple word overlap but instead of adding 1 to the overlap value if the word exists in both sentence, adding the IDF of this word. The normalization is performed dividing by the sum of the weights of coinciding and non-coinciding words.

### TF-IDF cosine similarity

For this last approach we have added a modification to the IDF, the term frequency (TF), which is the appearance frequency of the word in the current pair of sentences we are comparing.

In this case, instead of using the simple word overlap, we have used the cosine similarity which consists on defining two equal vectors containing the union the sentence (words that appear in only one or in both sentences), and substituting each word vector 1 or vector 2 by its corresponding TF-IDF value in the first or the second sentences respectively. Note that now, taking into account the term frequency of the word in the own sentence, a single word might have different TF-IDF value in different sentences.

## Results

As explained in the introduction, once the three metrics defined in *Part 2* have been computed we have selected three logistic regression functions from sklearn[3] and trained them with the training set. Then, using this computed function we have classified the words of the test set and compared these predicted labels with the true ones. The performance of the three different approaches has been computes with these performance evaluation metrics:

In order to compare the approaches to something we had already done, we have also computed this evaluation metrics using the Jaccard coefficient. These are the obtained results:

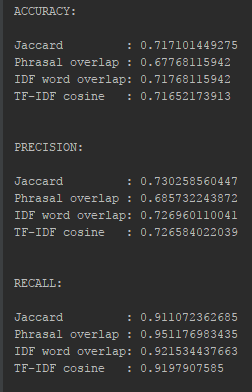


Figure 1: Accuracy, Precision and Recall of our approaches

As we can observe in *Figure 1*, the three sentence similarity metrics we have implemented perform similarly as the Jaccard coefficient. We think this might be due to the fact that in this document we are only looking at the lexical similarity. We are not taking into account the meaning of the words and sentences. But this is out of the scope of the project and will be seen in the next one.

In order to compare the performance of our metrics we have also recomputed everything but using now lemmas instead of words. These are the results:

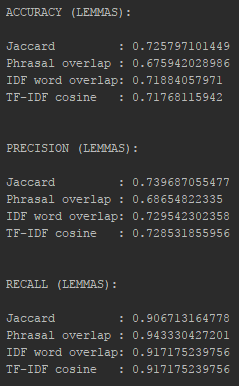


Figure 2: Accuracy, precision and recall of our approaches using lemmas

Here the results cannot be easily compared, but we have also plotted a colormap to make the comparison easier.

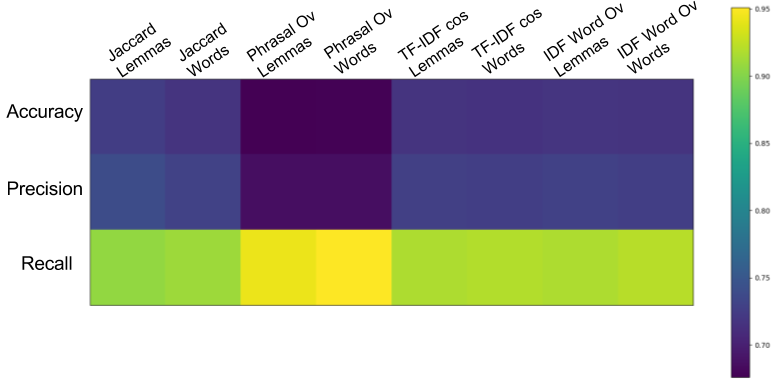


Figure 3: Colormap comparing the Accuracy, Precision and Recall of all our approaches

In this colormap can easily be seen that there are not nearly any changes in the performance of our metrics using words or lemmas and that despite the fact that the phrasal overview has a worse accuracy and precision, it has better recall.

# Part 2

In this second part of the project we have used the

## Syntactic dimension

# Bibliography

[1] <http://www.d.umn.edu/~tpederse/Pubs/ijcai03.pdf>

[2] <https://www.aaai.org/Papers/JAIR/Vol30/JAIR-3005.pdf>

[3] <http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html>