



Article

Recognizing Textual Entailment: Challenges in the Portuguese Language [†]

Gil Rocha *  and Henrique Lopes Cardoso 

LIACC/DEI, Faculdade de Engenharia, Universidade do Porto, Rua Dr. Roberto Frias, 4200-465 Porto, Portugal; hlc@fe.up.pt

* Correspondence: gil.rocha@fe.up.pt

† This manuscript is an extended version of “Recognizing Textual Entailment and Paraphrases in Portuguese”, presented at the Text Mining and Applications (TeMA) track of the 18th EPIA Conference on Artificial Intelligence (EPIA 2017) and published in *Progress in Artificial Intelligence*, Springer LNAI 10423, pp. 868–879.

Received: 28 January 2018; Accepted: 26 March 2018; Published: 29 March 2018



Abstract: Recognizing textual entailment comprises the task of determining semantic entailment relations between text fragments. A text fragment entails another text fragment if, from the meaning of the former, one can infer the meaning of the latter. If such relation is bidirectional, then we are in the presence of a paraphrase. Automatically recognizing textual entailment relations captures major semantic inference needs in several natural language processing (NLP) applications. As in many NLP tasks, textual entailment corpora for English abound, while the same is not true for more resource-scarce languages such as Portuguese. Exploiting what seems to be the only Portuguese corpus for textual entailment and paraphrases (the ASSIN corpus), in this paper, we address the task of automatically recognizing textual entailment (RTE) and paraphrases from text written in the Portuguese language, by employing supervised machine learning techniques. We employ lexical, syntactic and semantic features, and analyze the impact of using semantic-based approaches in the performance of the system. We then try to take advantage of the bi-dialect nature of ASSIN to compensate its limited size. With the same aim, we explore modeling the task of recognizing textual entailment and paraphrases as a binary classification problem by considering the bidirectional nature of paraphrases as entailment relationships. Addressing the task as a multi-class classification problem, we achieve results in line with the winner of the ASSIN Challenge. In addition, we conclude that semantic-based approaches are promising in this task, and that combining data from European and Brazilian Portuguese is less straightforward than it may initially seem. The binary classification modeling of the problem does not seem to bring advantages to the original multi-class model, despite the outstanding results obtained by the binary classifier for recognizing textual entailments.

Keywords: artificial intelligence; machine learning; natural language processing; recognizing textual entailment; paraphrase detection

1. Introduction

Human ability to express reasoning through natural language has given rise to an overwhelming amount of data. To tackle such huge amounts of information of heterogeneous quality, new tools are needed that assist humans in processing and interpreting written discourse. Being able to grasp the reasoning behind a given text is a path towards understanding its content.

Writing persuasive texts implies the use of appropriate argumentation skills. Backing up conclusions with appropriate premises may lead to convincing arguments. Cogent arguments typically denote rational reasoning [1]; however, valid arguments are better assessed through their consensual interpretation or objectivity [2]. The less assumptions are needed to interpret the argument, the more

likely we are in the presence of an entailment relation, i.e., an inference employing logical and objective reasoning.

In natural language processing (NLP), *recognizing textual entailment* (RTE) [3] is precisely devoted to identifying entailment relations between text fragments. Approaches to RTE have been applied before to address the problem of mining arguments from text [4]. Given two text fragments, typically denoted as “Text” (T) and “Hypothesis” (H), RTE is the task of determining whether the meaning of the hypothesis (e.g., “Joe Smith contributes to academia”) is entailed (can be inferred) from the text (e.g., “Joe Smith offers a generous gift to the university”) [5]. In other words, a sentence T entails another sentence H if after reading and knowing that T is true, a human would infer that H must also be true.

We may think of textual entailment and paraphrasing in terms of logical entailment (\models) [6]. If the logical meaning representations of T and H are Φ_T and Φ_H respectively, then $\langle T, H \rangle$ corresponds to a textual entailment pair if and only if $(\Phi_T \wedge B) \models \Phi_H$, where B is a knowledge base containing postulates that correspond to knowledge that is typically assumed to be shared by humans (i.e., common sense reasoning and world knowledge). Similarly, if the logical meaning representations of text fragments T_1 and T_2 are Φ_1 and Φ_2 , respectively, then T_1 is a paraphrase of T_2 if and only if $(\Phi_1 \wedge B) \models \Phi_2$ and $(\Phi_2 \wedge B) \models \Phi_1$.

Building a computational approach to detect textual entailment is challenged by the richness and ambiguity of natural language. Writers often make use of a rich vocabulary and different referring expressions to obtain a more fluent reading experience. In addition, writers tend to appeal to common-sense knowledge and inferring capabilities they assume the target reading audience to have. These assumptions turn out to pose very difficult challenges to computational systems aiming to automatically interpret natural language text. It turns out that the NLP community typically adopts a relaxed definition of textual entailment [6], so that T entails H if a human knowing that T is true would be expected to infer that H must also be true in a given context. A similar relaxed definition can be formulated for paraphrases.

RTE has been recently proposed as a general task that captures major semantic inference needs in several NLP applications [6,7], including: question answering [8], information extraction [9], document summarization [10], machine translation [11] and argumentation mining [4,12,13]. Since 2005, several challenges have been organized with the aim of providing concrete datasets that could be used by the research community to evaluate and compare different approaches. However, RTE from Portuguese text remains little explored. Recently, at the “International Conference on the Computational Processing of Portuguese” 2016 (*PROPOR 2016*), the “Evaluation of Semantic Similarity and Textual Inference” challenge (ASSIN, “Avaliação de Similaridade Semântica e Inferência Textual”) has been proposed [14]. This challenge introduces a Portuguese annotated corpus, useful for semantic similarity and textual inference tasks. This resource allows for the development of NLP systems using machine learning (ML) techniques to address this challenging RTE task.

In this paper, we aim to explore different approaches to address the task of recognizing textual entailment and paraphrases from text written in the Portuguese language, using supervised ML algorithms.

This paper is structured as follows: Section 2 presents related work on recognizing textual entailment and paraphrases, focusing on approaches based on text written in the Portuguese language. Section 3 introduces the existing corpora developed to provide annotated resource to train ML techniques for the task of RTE and paraphrases from text. We also describe the ASSIN corpus, the first corpus annotated with relations of entailment from Portuguese text, that was used in our experiments to validate the approach presented in this work. Section 4 describes the methods that were used to address the task of recognizing textual entailment and paraphrases using supervised machine learning algorithms. Section 5 presents the results obtained by the system described in this paper. Finally, Section 6 concludes and points to directions of future work.

2. Related Work

Computational methods for textual entailment and paraphrasing differ mainly on the initial assumptions and specific goals they were designed to address. In [6], the authors divided these systems in two main dimensions: (a) whether they focus on *paraphrasing* or *textual entailment* between text fragment pairs; and (b) whether they perform *recognition*, *generation* or *extraction* of paraphrases or textual entailment pairs. Since, in this paper, we focus on the recognition of paraphrase and textual entailment relations between pairs of sentences, the remainder of this section will focus on related work for this specific task. The main input given to a paraphrase or textual entailment recognizer is a pair of sentences, possibly in a particular context. The desired output is a (probabilistic) judgment, indicating whether or not the text fragments are paraphrases or a textual entailment pair.

State-of-the-art systems for RTE and paraphrase in natural language text typically follow a supervised machine learning approach. These systems rely on NLP pipelines (including tokenization, named entity recognition, syntactic tree parsing and dependency parsing, among other preprocessing tasks), extensive manual creation of features, several external resources (e.g., WordNet [15]) and specialized sub-components to address specific auxiliary sub-tasks [6,7,16], such as negation detection [6,17], semantic similarity [18], logical inference techniques [19,20], and coreference resolution [3,20,21]. More recently, state-of-the-art approaches for RTE rely on complex deep learning architectures employing sophisticated sentence encoding techniques and more straightforward NLP techniques (i.e., commonly requiring only tokenization and projection of the words into a distributional representation space). This paradigm shift follows a recent trend in the scientific community—focusing on how to employ ML techniques to directly extract structured and relevant knowledge from natural language resources. Deep learning techniques do not require extensive NLP pipelines in the preprocessing step.

For English text, several challenges have been proposed by the community, namely: eight RTE Challenges [22] organized between 2005 and 2013; SemEval 2014 Task 1 entitled “Evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment” [23], where the SICK dataset [24] has been presented; and, more recently, the Second Workshop on Evaluating Vector Space Representations for NLP (RepEval 2017) [25], featuring a shared task competition meant to evaluate natural language understanding models based on sentence encoders on the task of RTE, where the MultiGenre SNLI Corpus [26] has been presented. These challenges had a central role on the creation of the vast set of resources that are currently available to employ machine learning techniques for the task of RTE, and were responsible for stimulating the research community to work on these research lines. To the best of our knowledge, the first available resources and proposed computational systems to address the task of RTE and paraphrases for Portuguese were proposed in the ASSIN challenge [14] at *PROPOR 2016*. The ASSIN challenge follows similar guidelines and introduces the first corpus containing entailment and semantic similarity annotations between pairs of sentences in two Portuguese variants, European and Brazilian, suitable for the exploration of supervised machine learning techniques to address these tasks. To the best of our knowledge, the best ML approaches for RTE and paraphrases in Portuguese texts are presented in the ASSIN challenge.

In the following sections, we describe the current state-of-the-art on computational approaches that employ machine learning techniques to address the task of RTE and paraphrases from text. In Section 2.1, we present ML approaches that rely on heavily engineered NLP pipelines and extensive manual creation of features. These were the first approaches presented to address this task using ML techniques and remained state-of-the-art in terms of performance until 2015. In Section 2.2, we present ML approaches that rely on neural networks algorithms and sentence encoding techniques. These approaches constitute the current state-of-the-art in terms of accuracy.

2.1. Feature-Engineered Machine Learning Models

Conventional RTE systems employ (semi-)supervised ML techniques that rely on the manual creation of features that are given as input to the ML algorithms. These features map the input (i.e., natural language sentences T and H) to a numerical space that is used to represent the input in a structured format that can be used by ML algorithms. For the task of RTE, these features must capture the content of T and H and, more importantly, interconnections between T and H. Throughout the years, different features have been proposed by exploring several external resources. Typically, systems employ features at different levels of abstraction, namely lexical (e.g., words overlap, bag-of-words models, and substring matching), syntactic (e.g., part-of-speech tags, adverbs, punctuation marks, and syntactic trees), structural (e.g., sentence length), and semantic (e.g., exploring relations of synonyms and hypernyms from a WordNet, and similarity metrics from models of distributional representation of words). For instance, Bowman et al. [27] propose a classifier trained and evaluated on the SNLI [27] and SICK [24] corpora that implements six features at the lexical, syntactic and structural level, namely: BLEU [28] score of the hypothesis with respect to the premise, the length difference between the hypothesis and the premise, overlap between words in the premise and hypothesis (over all words and over just nouns, verbs, adjectives, and adverbs), an indicator for every unigram and bigram in the hypothesis, cross-unigrams, and cross-bigrams. The authors report an accuracy of 0.997 on SNLI and 0.904 on SICK training set, and an accuracy of 0.782 on SNLI and 0.778 on SICK test set.

One of the most widely used platforms to explore feature-based approaches is the Excitement Open Platform [29]. This platform follows a generic and modular architecture that allows developers to combine linguistic pipelines, entailment algorithms and linguistic resources within and across languages. The platform includes state-of-the-art algorithms, many knowledge resources, and facilities experimenting and testing different approaches. Moreover, the platform has various multilingual components for languages such as English, German and Italian.

In the remainder of this section, we will introduce some of the approaches that have been proposed by the community to address the task of RTE in Portuguese texts, using feature-based ML models. All of the proposed systems use the ASSIN Corpus to train and test ML algorithms.

In [30], Hartmann followed the supervised ML paradigm with an approach based on the cosine similarity of the vectorial representation of each sentence. These sentence representations were obtained from the sum of the vectors representing each word in a sentence using two word weighting schemes, namely: *TF-IDF* [31] and *word2vec* [32]. Then, for each sentence pair, the cosine similarity between the vectorial representation (i.e., sum of the word vectors using the weighting schemes previously described) of the text sentence T and the hypothesis sentence H is used as features (one feature using TF-IDF and another using word2vec to represent each sentence) and given as input to train a linear classifier.

Fialho et al. [33] extracted several metrics for each pair of sentences, namely edit distance, words overlap, BLEU [28] and ROUGE [34], among others. They reported several experiments considering different preprocessing steps in the NLP pipeline, namely: original sentences (baseline), removing stop-words, lower-case words and clusters of words. A feature set containing more than 90 features to represent each pair of sentences was used as input for a SVM classifier. Fialho et al. also reported experiments merging the original ASSIN corpus with annotated data from the SICK corpus translated from English to Portuguese, using a Python wrapper over the Microsoft Bing translation service. They added 9191 examples from the SICK corpus to the 6000 examples from the ASSIN training set in one of their experiments. The results reported on the augmented version of the training data were worse than the results reported on the original training data. The authors associated these results to translation errors that were probably made during the process. In addition, they trained their model in one of the Portuguese variants of the ASSIN corpus and evaluated the performance of the model in the other Portuguese variant. Reported results following this experimental setup were worse when compared with the model trained and tested in the same variant, but were better than the results

obtained in the augmented version of the original dataset (with the *SICK* data). They obtained the best results for RTE in the ASSIN challenge: 0.843 of accuracy and 0.66 of macro F1-score.

In [35], Alves et al. explored two different approaches for RTE and paraphrases: a heuristic-based approach (“Reciclagem” system) and a supervised ML approach (“ASAPP” system). Both approaches made use of the same component for analyzing lexical semantic relations, which is based on the analysis of semantic networks (e.g., wordnet for Portuguese). The “Reciclagem” system is based on lexical and semantic knowledge that calculates the similarity and relations of two sentences without any kind of supervised ML methods. This system was used as a baseline for the “ASAPP” system and to evaluate the quality of different lexical and semantic resources for Portuguese. The “ASAPP” system follows the supervised ML approach and adds to “Reciclagem” features based on the syntactic and structural information extracted from the pair of sentences, such as number of tokens, overlapping words, synonyms, hyperonyms, meronyms, antonyms and number of words with negative connotation, type of named entities, among others. In their experiments, the authors explored different strategies to divide the training data, combining results from different classifiers and several feature selection techniques. They reported accuracy of 0.731 and macro F1-score of 0.43 in the European Portuguese test data.

2.2. Neural Network Models Based on Sentence Encoding

Current state-of-the-art results on RTE were obtained by exploring neural networks with several layers of neurons (known as *deep learning* architectures) and with complex encoding of the sentences T and H . In general, these approaches follow the architecture depicted in Figure 1. In a RTE setting, the system receives as input a pair of sentences. The bottom layers are responsible for encoding the sentences written in natural language into a representation capturing information from both sentences (sentence encoding). First, we have to map natural language sentences into a representation that is suitable for being processed by a computational system. The conventional approach is to split each natural language sentence in tokens (tokenization), mapping the original sentence to a set of tokens. Then, each token is mapped to a word embedding space, a representation that is used henceforth. Next, the set of tokens must be mapped to a fixed-length vector that captures all the relevant information presented in the sentence and that is suitable for being used by the following layers in the neural network (sentence encoding step). Different ways of performing sentence encoding have been proposed by the community, but typically this step is performed in one of two ways: (a) sentence encoding-based models that encode T and H separately and then merge the encodings in a fixed-length vector; or (b) joint methods that share the encoding for each sentence in a single representation (e.g., cross-features, attention mechanisms, sequence representations).

From the sentence encoding step, a fixed-length vector is obtained that captures the information for T and H and, possibly, the relation between the sentences. The resulting vector is fed into a multilayer neural network that culminates in a softmax layer (output layer) to output the final predictions made by the neural network. The softmax layer outputs a vector of non-negative real numbers that sum to one, making the output layer a discrete probability distribution over the possible output classes (e.g., *None*, *Entailment* and *Paraphrase* in the ASSIN Corpus). Different architectures have been proposed for the multilayer neural network that maps the sentence encoding step to the softmax layer.

Following the sentence-encoding setting identified as (a) above, Bowman et al. [27] proposed three different architectures, namely Sum of Words, RNN and LSTM [36], each of them mapping the sentences to a 100d vector and concatenating each vector to obtain a 200d vector in the end. For a sentence S containing a sequence of n words (w_1, \dots, w_n) , this approach computes a representation of S as $\vec{S} = \sum_{k=1}^n \vec{e}(w_k)$, where $\vec{e}(w_k)$ represents the embedding vector of word w_k . In the Sum of Words architecture (baseline approach), the 100d vector representation of a sentence is obtained by summing the embeddings of the tokens in the sentence. In the sequence embeddings models (RNN and LSTM), the authors fed each 100d vector representation sequentially in a recurrent neural net (RNN and LSTM) and use the final 100d representation of the hidden state as the final sentence representation.

For a sequence of n words (w_1, \dots, w_n) the network computes a set of n hidden representations $\vec{h}_1, \dots, \vec{h}_n$ with $\vec{h}_n = \overrightarrow{RNN}(\vec{e}(w_1), \dots, \vec{e}(w_n))$. A sentence is represented by the last hidden vector \vec{h}_n . The neural network classifier is a stack of three 200d tanh layers that feeds a final softmax layer. The learning procedure for the neural network classifier and sentence encoding is performed jointly. Results reported on the SNLI corpus show that the LSTM sentence encoding setting performs better in both training set (0.848) and test set (0.776), followed by the Sum of Words setting (0.793 on the training set and 0.753 on the test set) and, finally, by the RNN setting (0.731 on the training set and 0.722 on the test set). All the reported results are presented in terms of overall accuracy for a three-class classification task. Comparing these results with a feature-based approach, they obtained similar results when training on the full corpus. However, from the increase of accuracy experienced by the LSTM as new examples were added to the corpus in comparison with the feature-based approach, the authors claim that this can be an indicator that the LSTM model may take more advantage of larger datasets and, therefore, they expect the LSTM model to achieve better performance when the number of annotations in the dataset increases.

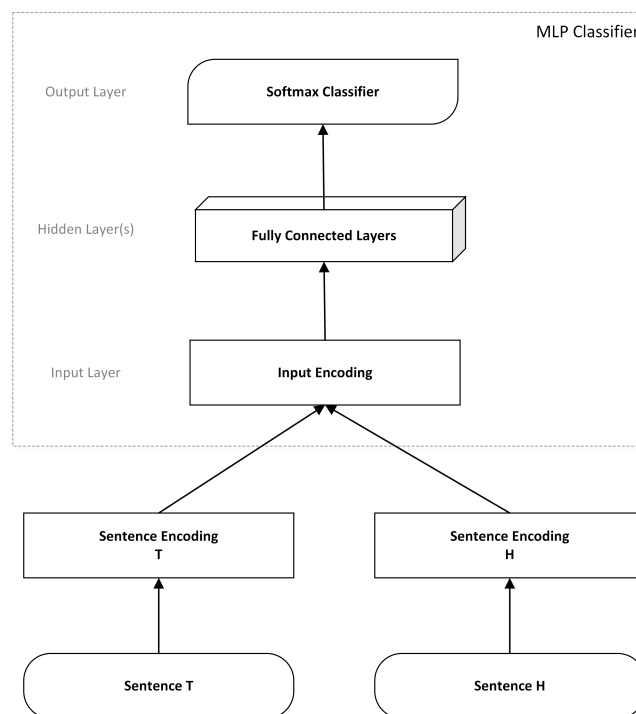


Figure 1. Generic architecture of a neural network system for RTE.

Following the sentence encoding identified as (b) above, Rocktäschel et al. [16] employed an attentive neural network that is capable of reasoning over entailments of pairs of words and spans of text by processing the hypothesis conditioned on the premise. In contrast with the approach presented by Bowman et al. [27] who encoded each sentence independently into a semantic space, in this work, the authors processed the two sentences as follows. First, a LSTM is employed to encode sentence T by processing each word in a sequential setting $\overrightarrow{LSTM}(\vec{e}(w_1), \dots, \vec{e}(w_n))$. Then, a second LSTM with different parameters reads a delimiter and the hypothesis H , but its memory state is initialized with the last cell state of the previous LSTM. In other words, the second LSTM is conditioned on the representation of the first LSTM that was obtained after processing text sentence T (conditional encoding). For classification, a softmax layer is employed over a non-linear projection of the output vector obtained from the last hidden state of the second LSTM into the target space of the three classes available in the SNLI corpus. The learning procedure of the sentence encoding and neural network classifier is performed jointly using a cross-entropy loss method. Finally, the authors propose to

augment the LSTM with attention mechanisms. The idea is to allow the model to attend over past output vectors with the aim of focusing the model to pay more attention to some parts of the output vector that are more relevant. More precisely, a LSTM with attention for RTE does not need to capture the whole semantics of the sentence T in its cell state and final hidden state. Instead, the attention mechanism informs the second LSTM about which hidden states, obtained while reading sentence T , it should attend to determine the RTE class. From the analysis of the results obtained by employing these models in the SNLI corpus, the authors concluded that: (a) conditional encoding gives an improvement of 3.3 percentage points in accuracy over Bowman et al. [27] LSTM and outperforms a simple lexicalized classifier by 2.7 percentage points; and (b) incorporating attention mechanisms improved the accuracy of the LSTM with conditional encoding by 1.4 percentage points.

In sum, the current best performing state-of-the-art systems for the task of recognizing textual entailment and contradictions from text written in English (<https://nlp.stanford.edu/projects/snli/>) employ neural network algorithms based on deep learning architectures, following a setting similar to the one depicted in Figure 1.

3. Corpora

To successfully apply ML algorithms, a good training corpus is crucial. A training corpus contains instances (or examples) from which ML algorithms generate models to automatically address some specific task. Therefore, a collection of annotated pairs of sentences (corpus) labeled with the type of entailment relation is an important requirement in order to address the task of recognizing textual entailment and paraphrases using supervised ML techniques.

3.1. English Corpora

Table 1 summarizes the available corpora containing sentence pairs annotated with entailment relations (textual entailment, contradiction and/or paraphrase) from text written in English.

Table 1. Available corpora for recognizing entailment relations in English text (based on Bowman [37]). “# Examples” indicates the number of labeled sentence pairs in the corpus. “Corpus Creation” indicates whether the corpus contains pairs of natural language sentences that were manually added by human annotators (“Manual”) or automatically generated (“Automatic”). “Labels” indicates the different labels that were assigned to each example.

Corpus	# Examples	Corpus Creation	Labels
FraCas [38]	346	Manual	Yes, No, Unk
RTE 1-5 [7]	7 K	Manual	None, Entailment, Contradiction
SICK [24]	10 K	Manual	Neutral, Entailment, Contradiction
MultiNLI [26]	433 K	Manual	Neutral, Entailment, Contradiction
SNLI [27]	570 K	Manual	Neutral, Entailment, Contradiction
Denotation Graph [39]	728 K	Automatic	Entailment, Non-Entailment
Entailment Graphs [40]	1.5 M	Automatic	Entailment, Non-Entailment
PPDB 2.0 [41]	100 M	Automatic	Entailment (FW or BW), Non-Entailment (Topic Rel. or Unrel.), Contradiction

From the eight existing corpora presented in Table 1, the first five corpora contain natural language sentences pairs, while the remaining three corpora are not structured as sentence pairs and were generated automatically.

The FraCas corpus [38] was the first available corpus for RTE. It was manually created by expert annotators and was built to highlight a diverse range of inference challenges that must be addressed to construct a powerful RTE system, such as quantification, coreference resolution and logical connectives. It was annotated using three labels, namely: Yes (forward entailment), No (contradiction) and Unk (independence). The small size of the corpus makes it useless for training ML algorithms, but the

quality of the annotations and diverse semantic phenomena captured in the annotations make it an interesting benchmark test set.

The Recognizing Textual Entailment dataset (RTE) [7] has evolved over time in the context of a series of competitions. The latest version is annotated with three classes, namely “none”, “entailment” and “contradiction”. The different versions of this dataset were widely used by the community [6] and have served as useful benchmarks for RTE ML models. However, their limited size hindered the performance of the proposed systems, in particular neural networks models that typically require a substantial amount of annotations.

The Sentences Involving Compositional Knowledge dataset (SICK) [24] was proposed in the SemEval 2014 Task 1 challenge [23]. SICK contains human-annotated examples derived from image and video captions and labeled in three classes, namely “neutral”, “entailment” and “contradiction”. The corpus creation process drew on only a few hundred source text sentences and involved the manual creation of hypothesis sentences by human annotators, following a set of guidelines (additional details can be found in [7]).

The Stanford Natural Language Inference (SNLI) corpus [27] contains a collection of sentence pairs labeled for “entailment”, “contradiction” and semantic independence (“neutral”). In contrast to other resources, all of its sentences and labels were written by humans in a grounded, naturalistic context. The annotation was based in a crowd-sourcing process but including some steps of validation to ensure the quality of the produced corpus. Text sentences (T) were obtained from Flickr30k [39], a corpus containing literal descriptions of scenes depicted in images. The task of annotators was to provide a hypothesis sentence (H) for each of the labels based on the same text sentence T .

The Multi-Genre Natural Language Inference (MultiNLI) corpus [26] was proposed for the RepEval 2017 challenge [25]. The annotation process involved crowd-sourcing and resulted in the annotation of sentence pairs with textual entailment information. The characteristics of the MultiNLI corpus are similar to the SNLI corpus [27] (in terms of the guidelines and process followed to create the corpus), but the MultiNLI corpus differs by covering a range of genres of spoken and written texts, and by supporting a distinctive cross-genre generalization evaluation.

Denotation Graph [39] contains examples of entailments between sentences and artificially constructed short phrases; it was constructed using fully automatic methods. Entailment Graphs [40] contains semi-automatically annotated entailment examples between subject–verb–object relation triples. Finally, Paraphrase Database (PPDB) 2.0 [41] includes automatically generated entailment and paraphrase annotations over a large corpus of pairs of words and short phrases. For any given input phrase to PPDB, there are often dozens or hundreds of possible paraphrases. For each paraphrase pair, the database includes fine-grained entailment relations, word embeddings similarities, and style annotations.

3.2. Portuguese Corpora

The ASSIN corpus [14] is, to the best of our knowledge, the first corpus annotated with pairs of sentences written in Portuguese that is suitable for the exploration of textual entailment and paraphrasing classifiers. The corpus contains pairs of sentences extracted from news articles written in European Portuguese (EP) and Brazilian Portuguese (BP), obtained from *Google News Portugal* and *Google News Brazil*, respectively. To create the corpus, the authors started by collecting a set of news articles describing the same event (one news article from *Google News Portugal* and another from *Google News Brazil*) from *Google News*. Then, they employed *Latent Dirichlet Allocation* (LDA) [42] models to retrieve pairs of similar sentences between sets of news articles that were grouped together around the same topic. For that, two LDA models were trained (for EP and for BP) on external and large-scale collections of unannotated news articles from Portuguese and Brazilian news providers, respectively. Then, the authors defined a lower and upper threshold for the sentence similarity score of the retrieved pairs of sentences, taking into account that high similarity scores correspond to sentences

that contain almost the same content (paraphrase candidates), and low similarity scores correspond to sentences that are very different in content from each other (no-relation candidates).

From the collection of pairs of sentences obtained at this stage, the authors performed some manual grammatical corrections and discarded some of the pairs wrongly retrieved. Furthermore, from a preliminary analysis made to the retrieved sentence pairs the authors noticed that the number of contradictions retrieved during the previous stage was very low. Additionally, they also noticed that event though paraphrases are not very frequent, they occur with some frequency in news articles. Consequently, in contrast with the majority of the currently available corpora for other languages, which consider as labels “neutral”, “entailment” and “contradiction” for the task of RTE, the authors of the ASSIN corpus decided to use as labels “none”, “entailment” and “paraphrase”.

Finally, the manual annotation of pairs of sentences was performed by human annotators. At least four annotators were randomly selected to annotate each pair of sentences, which is done in two steps: (i) assigning a semantic similarity label (a score between 1 and 5, from unrelated to very similar); and (ii) providing an entailment label (one sentence entails the other, sentences are paraphrases, or no relation). Sentence pairs where at least three annotators do not agree on the entailment label were considered controversial and thus discarded from the gold standard annotations. The ASSIN challenge [14] included two tasks, both using the ASSIN corpus: (a) semantic similarity; and (b) textual entailment and paraphrase recognition. We will focus on the latter: the “entailment” label is the attribute that will be used as target for the proposed task.

In total, the ASSIN corpus contains 10,000 pairs, half in each of the Portuguese variants. The distribution of $\langle T, H \rangle$ pairs between each “entailment” label and between texts written in BP and EP is shown in Table 2. It is important to note that the ASSIN corpus is unbalanced in relation to the “entailment” and “paraphrase” labels. This characteristic of the corpus should be taken into account when employing ML classifiers to address the task and in the analysis of the obtained results. The split between training and test set is provided with the corpus. We kept this division in our experiments to compare the results that we obtain with state-of-the-art work developed on top of this corpus. Inter-annotator agreement metrics associated to the construction of this corpus [14] are *Fleiss’s \mathcal{K}* of 0.61 and Concordance of 0.8. The *Fleiss’s \mathcal{K}* value is relatively low, demonstrating the subjectivity associated with the annotation process. However, these values are not very different from the values reported in other corpora used for the same task: for instance, in the RTE Challenges, the values ranged from 0.6 in the first RTE Challenge to 0.75 or more in the following challenges [14,22].

Table 2. Distribution of labels in ASSIN corpus.

Label/Partition	BP		EP	
	Train	Test	Train	Test
None	2331	1553	2046	1386
Entailment	529	341	729	481
Paraphrase	140	106	225	133

Table 3 shows one example of the content and annotations available in the ASSIN corpus for each of the “entailment” labels.

One important characteristic of the ASSIN Corpus is the fact that each sentence in a pair $\langle T, H \rangle$ is unique for the whole corpus, even though some of them may be semantically close to each other (news articles were grouped together by topics when creating the corpus). In other corpora (e.g., SICK and SNLI), for each text sentence T , the corpus contains different hypothesis sentences H for each of the labels that are provided (i.e., neutral, entailment and contradiction). In terms of the learning procedure, we believe that having annotations that demonstrate how different hypothesis sentences H may yield to different $\langle T, H \rangle$ labels for a given T is important to help the ML to distinguish with more confidence the boundaries for each of the labels. Consequently, we believe that the annotations provided in the ASSIN corpus should have been generated following the same procedure that the authors of other

corpora (e.g., SICK and SNLI) containing annotations of textual entailment relations have proposed. Additionally, we believe that this characteristic of the corpus may harness the performance of systems that employ supervised ML techniques solely based on the annotations provided in the corpus.

Table 3. Annotated examples from the ASSIN corpus (extracted from [14]).

Label	Pair of Sentences
None	As apostas podem ser feitas até as 19 h (de Brasília). (T) <i>Bets can be made until 9 pm (Brasília time). (T)</i>
	As apostas podem ser feitas em qualquer lotérica do país. (H) <i>Bets can be made at any lottery house in the country. (H)</i>
Entailment	Como não houve acordo, a reunião será retomada nesta terça, a partir das 10 h. (T) <i>Since there was no deal, the meeting will resume this Tuesday, starting at 10 am. (T)</i>
	As partes voltam a se reunir nesta terça, às 10 h. (H) <i>The factions will meet again this Tuesday, at 10 am. (H)</i>
Paraphrase	Vou convocar um congresso extraordinário para me substituir enquanto presidente. (T) <i>I will convene an extraordinary congress to replace me as president. (T)</i>
	Vou organizar um congresso extraordinário para se realizar a minha substituição como presidente. (H) <i>I will organize an extraordinary congress to carry out my replacement as president. (H)</i>

4. Methods

We here describe the approach we follow to address the task of entailment and paraphrase recognition from natural language Portuguese text. We formulated the problem following two different settings: first, as a multi-class classification problem, in which we aimed to classify each $\langle T, H \rangle$ with one of the labels *Entailment* (if $T \models H$), *Paraphrase* (if $T \models H$ and $H \models T$, i.e., if T is paraphrase of H), or *None* (if T and H are not related with one of the previous labels); and, second, as a binary classification problem, aimed to distinguish each $\langle T, H \rangle$ with one of the labels *Entailment* or *None* (details regarding the experimental setup and obtained results following these formulations is described in Section 5). In both formulations, we employed supervised ML techniques to construct a computational system capable of RTE and paraphrases from text, using the ASSIN corpus to train and test the quality of the predictions made by the system (using the training and test partitions of the ASSIN corpus, respectively).

Designing a system able to automatically recognize textual entailment and paraphrases given a pair of sentences written in natural language requires the implementation of different techniques to process natural language text written in the Portuguese language. Additionally, methods to represent natural language pairs of sentences into a set of features suitable to employ ML algorithms is also a required step in this pipeline.

Firstly, to transform each sentence into the corresponding set of tokens and to obtain for each token the corresponding lemma and part-of-speech information (including syntactic function, person, number and tense, among others), we used the *CitiusTagger* [43] NLP tool. This tool includes a named entity recognizer trained in natural language text written in Portuguese.

Several experiments were made using different NLP techniques to process the sentences received as input: removing stop-words, removing auxiliary words (i.e., words relevant for the discourse structure but not domain specific, such as prepositions, determiners, conjunctions, interjections, numbers and some adverbial groups) and lemmatization. From this possible pre-processing setup, we expect that: (a) Transforming each token into the corresponding lemma is a promising approach, particularly in Portuguese, a language where word inflection is extremely rich. Additionally, it will make explicit that some of the words are repeated in both sentences even if small variations of these words are used in each sentence (e.g., different verb tenses). (b) Removal of stop-words and auxiliary words will have a positive impact in the obtained results by focusing the attention on words that may indicate relations of entailment (e.g., hypernyms).

After this pre-processing step, each sentence contained in the pair $\langle T, H \rangle$ under analysis is represented in a structured format (set of tokens) and annotated with some additional information regarding the content of the text (e.g., part-of-speech tags).

Secondly, to apply ML algorithms, we need to represent each learning instance, each $\langle T, H \rangle$ pair, by a set of numerical features. A good set of features should represent the training instances in such a way that would make it possible for the machine learning algorithms to find patterns in the data which can be used to classify instances according to the desired target labels. Since in this problem we received a pair of sentences as input and we aimed to automatically classify the relation between them as output, the feature set should be designed taking special attention to the properties that characterize such relation.

To represent each pair $\langle T, H \rangle$, we employed a set of features (listed in Table 4) at the lexical, syntactic and semantic level. The first four lexical features aimed to capture the overlap of information expressed in T in relation to H and vice versa. Feature T_Bigger_H tries to capture the intuition that in a relation of *Entailment*, sentence H is usually smaller than sentence T . Regarding syntactic features, changes in verb tense are typically not expected to occur in *Paraphrase* relations, while rewriting the same sentence using alternation between passive and active voice is the most common case of paraphrase relations. Semantic features were employed for tokens in one of the sentences that do not occur in the other, after removing auxiliary words and named entities, to focus attention on words that are possible indicators of relations of entailment. The first three features captured semantic relations between each pair of tokens using knowledge extracted from a Portuguese wordnet. The last two features explored the word embeddings model and aimed to capture different ways of measuring semantic relations between H and T , after projecting each sentence in the embedding space.

Table 4. Feature set.

Type	Feature	Description
Lexical	Overlap_T	% of (unique) tokens in T that exist in H .
	Overlap_H	% of (unique) tokens in H that exist in T .
	NE_T	% of (unique) named entities in T that exist in H .
	NE_H	% of (unique) named entities in H that exist in T .
	T_Bigger_H	If $ T > H $ returns 1. Returns 0, otherwise.
Syntactic	Tense	If T and H are written in the same grammatical tense.
	Voice	If T and H are written in the same grammatical voice.
Semantic	Synonym	% of tokens in T synonyms of tokens in H , and vice versa.
	Hyperonym	% of tokens in T hyperonyms of tokens in H , and vice versa.
	Meronym	% of tokens in T meronyms of tokens in H , and vice versa.
	Cos_Sim	cosine similarity between $\vec{e}(T)$ and $\vec{e}(H)$.
	Entail_Versor	entailment versor (\hat{d}) in the word embeddings space

Knowledge about the words of a language and their semantic relations with other words can be exploited with large-scale lexical databases. With the aim of enabling the system to better deal with the diversity and ambiguity of natural language text, we explore external semantic resources. Similar to WordNet [15] for the English language, CONTO.PT [44] is a wordnet for Portuguese, which groups words into sets of cognitive synonyms (synsets), each expressing a distinct concept. In addition, synsets are interlinked by means of conceptual and semantic relations (e.g., “hyperonym” and “part-of”). Synsets included in CONTO.PT were automatically extracted from several linguistic resources, namely based on the redundancy of the relations existing on other Portuguese wordnets. Since CONTO.PT can be seen as an updated agglomeration of existing resources for Portuguese, we decided to use CONTO.PT in our experiments. Additionally, all relations represented in CONTO.PT (both relations between words and synsets, and relations between synsets) include degrees of membership. Two tokens (obtained after tokenization and lemmatization) are considered synonyms if they occur in the same synset. Token t_i is considered hyperonym of t_j if there exists a hyperonym relation

("hyperonym_of") between the synset of t_i and the synset of t_j . Similarly, t_i is considered meronym of t_j if there exists a meronym relation ("part_of" or "member_of") between the synset of t_i and the synset of t_j . Given that in CONTO.PT each synset contains words combined with the corresponding syntactic function (e.g., noun and adjective), we retrieve the corresponding synsets taking into account the part-of-speech tags that were associated to each word in the pre-processing stage; this allows us to perform a simplified disambiguation of CONTO.PT senses.

Finally, we exploit a distributed representation of words (word embeddings) to compute the last two features listed in Table 4. These distributions map a word in a dictionary to a feature vector in a high-dimensional space, without human intervention, by observing the usage of the word on large (non-annotated) corpora. This real-valued vector representation tries to arrange words with similar meanings close to each other based on the occurrences of these words in large-scale corpora. Then, from these representations, interesting features can be explored, such as semantic and syntactic similarities. In our experiments, we used a pre-trained model provided by the *Polyglot* (<http://polyglot.readthedocs.io/en/latest/index.html>) tool [45], in which a neural network architecture was trained with Portuguese Wikipedia articles.

To obtain a score indicating the similarity between two text fragments T and H , we compute the cosine similarity between the vectors that represent each of the text fragments in the high-dimensional space. Each text fragment is projected into the embedding space as $\vec{T} = \sum_{k=1}^n \vec{e}(w_k)n^{-1}$, where $\vec{e}(w_k)$ represents the embedding vector of the word w_k and n corresponds to the number of words contained in the text fragment (T or H). Then, we compute the final value of the cosine similarity $\delta_{\vec{T}, \vec{H}} = \cos(\vec{T}, \vec{H})$, where $\delta_{\vec{T}, \vec{H}} \in [-1, 1]$, followed by the following rescaling and normalization: $(1.0 - \delta_{\vec{T}, \vec{H}})/2.0$. The entailment versor (\hat{d}) corresponds to the normalized direction vector obtained by subtracting the projection of H ($\vec{e}(H)$) from the projection of T ($\vec{e}(T)$) in the embedding space.

For each classification task, we have run several experiments exploring some well known state-of-the-art algorithms, namely: *Support Vector Machine* (SVM) using linear and polynomial kernels, *Maximum Entropy model* (MaxEnt), *Adaptive Boosting* (AdaBoost) using *Decision Trees* as weak classifiers, *Random Forests* using *Decision Trees* as weak classifiers, and *Multilayer Perceptron* (Neural Net) with one hidden layer. All the ML algorithms previously mentioned were employed using the *scikit-learn* library [46] for the *Python* programming language. Since the best overall results for the baseline scenario (see Section 5.1) were obtained using *MaxEnt*, all results reported in Section 5 were obtained using this classifier.

5. Experiments and Results

In this section, a detailed description of the experiments that were made to validate some of the hypothesis presented in previous sections and to evaluate the quality of the proposed system designed to automatically recognize textual entailment and paraphrases from natural language text written in Portuguese is presented. The learning instances used to train the classifiers and to evaluate the predictions made by the system were obtained from the ASSIN Corpus, described in Section 3. Since our aim is to design a computational system to address the problem of recognizing textual entailment and paraphrases from text written in EP, in all the evaluation scenarios we report on the results obtained by the models on a separate test set from the ASSIN corpus containing examples annotated in EP.

We investigate three evaluation scenarios. First, in Section 5.1, we report 10-fold cross validation results over all the training examples of the European Portuguese partition of the ASSIN corpus, following a supervised ML setting and employing the methods described in Section 4. In this evaluation scenario, the system obtained by employing the lexical, syntactic and semantic-based features (complete set of features described in Table 4) corresponds to our baseline system. Furthermore, we also report on the impact that semantic-based features have in overall performance of the system. In the second evaluation scenario (Section 5.2), we report 10-fold cross validation results over all the training examples available in the ASSIN corpus, including both the European Portuguese and the Brazilian

Portuguese partitions, using the complete set of features described in Table 4. In this evaluation scenario, we aim to validate our intuition that increasing the training set with more training data, regardless of the differences between European Portuguese and Brazilian Portuguese, should increase the performance of the system for the task of recognizing textual entailment and paraphrases from text written in European Portuguese. Finally, in the third evaluation scenario (Section 5.3), we address the problem in a different perspective by exploring the characteristics of the ASSIN corpus and the theoretical formulations of entailment and paraphrases. In this scenario, we formulate the task of RTE as binary classification problem and report on the results obtained using this formulation. Then, we evaluate the quality of the predictions made by the binary classifier, trained for recognizing textual entailments, in the original multi-class classification task for RTE and paraphrases on the EP test set partition of the ASSIN corpus.

5.1. Using EP Annotations

In the baseline scenario, we formulate the problem as a multi-class classification task employing the methods described in Section 4. We train the ML algorithms on the training set of the EP partition and test the corresponding models on the test set of the EP partition of the ASSIN corpus. The aim of this baseline scenario is to obtain experimental results of employing the methods described in Section 4 on the standard setting of the ASSIN challenge. Additionally, we also report on the importance that the semantic-based features have in the overall performance of the system.

Table 5 summarizes the results obtained in our experiments regarding the multi-class formulation. The first line corresponds to the system trained on the EP training set partition of the ASSIN corpus, containing a total of 3000 annotated sentence pairs, employing the complete set of features described in Table 4. The first three columns correspond to the averaged F1-score evaluation metric obtained after performing 10-fold cross validation on the training data for each label considered in the classification problem: *None* (N), *Entailment* (E) and *Paraphrase* (P). The following three columns correspond to the overall results obtained in the training set for each evaluation metric, namely *micro F1-score* (F1), *macro F1-score* (Macro-F1) and *accuracy* (Acc.). Finally, the last two columns correspond to the overall *macro F1-score* and *accuracy* obtained in the test set.

Table 5. Evaluation results for each evaluation scenario of the multi-class setting.

	Train						Test	
	N	E	P	Total			Total	
	F1	F1	F1	F1	Macro-F1	Acc.	Macro-F1	Acc.
EP	0.9	0.7	0.59	0.83	0.73	0.826	0.73	0.835
EP + BP	0.9	0.65	0.54	0.83	0.7	0.824	0.72	0.832

In general, we obtained better overall results in the recognition of the *None* relation (0.9), followed by *Entailment* relations (0.7) and by *Paraphrase* relations (0.59). We associate these results to the different numbers of learning instances available in the corpus (see Table 2).

To determine the impact of the semantic-based features in the overall performance of the system, we made additional experiments by evaluating the performance of a system employing only the lexical and syntactic-based features described in Table 4. Trained on the EP training set partition of the ASSIN corpus, this system obtained an accuracy of 0.819 and a macro F1-score of 0.7 on the EP test set partition. The 10-fold cross-validation results on the training set are very similar to the results reported in the first line of the Table 5 for the system employing the complete set of features. Enhancing the feature set with semantic-based features has improved overall results. In particular, we observed that the system employing semantic-based features obtained significant improvements for the results reported on the EP test set, showing a better capability of generalizing for unseen data. However, we expected these improvements to be more significant, since it seems intuitive that semantic-based features are relevant

for the task of recognizing textual entailment and paraphrases. After performing feature and error analysis, we associate these results with the following: (a) the system gave too much importance to the “percentage of overlapping tokens” feature (i.e., when the value of the feature “Overlap_T” is very high the system tends to predict *Paraphrase*, when the feature “Overlap_H” is very high the system tends to predict *Entailment*, and when these values are both very low the system tends to predict *None*); and (b) the coverage of semantic-based features is not total (i.e., some words in the corpus do not occur in the external resources that we rely on to gather the semantic knowledge), causing some words (or features) to have null values in some situations. Regarding the first observation, this seems to be one characteristic of the corpus used in these experiments: simply by measuring the overlap of lexical terms is very useful to discriminate between each of the annotated pairs. Regarding the latter observation, we aim to investigate in future work how to deal with these coverage issues and study better ways to compute semantic-based metrics using external resources.

5.2. Adding BP Annotations

The last line in Table 5 reports 10-fold cross validation results of a system employing the complete set of features described in Table 4, when trained on both EP and BP partitions of the ASSIN training corpus, and the corresponding performance metrics on the test set of the EP partition. We observe that increasing the training set with the BP partition reduced the overall performance of the system. These results suggest that some characteristics of entailment and paraphrase relations between two text fragments of the Brazilian Portuguese partition are different from the European Portuguese partition. Furthermore, syntactic and semantic differences between the two variants are responsible for the majority of the errors made by the system. Even though both partitions are written in the same native language, they correspond to different variations of Portuguese, including differences that seem to impact the results obtained by the system for the task addressed in this paper.

To validate this intuition, we have made some additional experiments. In this new experimental setup, we take the system employing the full set of features described in Table 4 and trained on the complete EP training set (results can be seen in the first line of Table 5) as the theoretical upper bound in terms of performance—training with all available EP annotations should in theory perform better on the EP test set. Then, we train the same system configuration on a smaller partition of the EP training set (maintaining approximately the same proportion of examples per class) and retrieve the corresponding performance metrics (always using the complete EP test set partition of the ASSIN corpus). We then add learning instances from the BP training set partition of the ASSIN corpus until the number of training examples is the same as the original setup (containing 3000 training examples, namely 2046 *None*, 729 *Entailment* and 225 *Paraphrase* examples, according to Table 2). Finally, we retrieve the corresponding performance metrics.

The results obtained, summarized in Table 6, lead us to conclude that adding BP annotations harnesses the performance of the system on the EP test set of the ASSIN corpus. The first line in Table 6 corresponds to the results obtained when training the system on 2000 examples randomly selected from the EP training set, but keeping the distribution of examples per class (1364 *None*, 486 *Entailment* and 150 *Paraphrase* learning instances). The second line shows the results obtained by adding learning instances from the BP training set (randomly selected from the ASSIN corpus), also keeping the same distribution of examples per class (682 *None*, 243 *Entailment* and 75 *Paraphrase* examples were added). By comparing the scores for the test set, we can observe that adding BP learning instances to the training set brings worse performance metrics. These results are in line with those reported in Table 5. By comparing the scores in the last line of Table 6 (containing 2000 learning instances from the EP training set and 1000 learning instance from the BP training set) with the results shown in the first line of Table 5 (containing 3000 learning instances from the EP training set), we can conclude that adding EP learning instances improves the performance of the system, while adding BP learning instances harnesses the performance. These results support our initial intuition for explaining the results obtained when we trained the system on the complete EP and BP partitions of the training set.

Table 6. Adding BP annotations to different partitions of the EP training set.

	Train					Test		
	N	E	P	Total		Total		
	F1	F1	F1	F1	Macro-F1	Acc.	Macro-F1	Acc.
2000 EP	0.9	0.68	0.59	0.82	0.723	0.822	0.727	0.834
2000 EP + 1000 BP	0.89	0.67	0.56	0.81	0.707	0.810	0.717	0.828

A detailed analysis of these differences is left for future work. It seems that employing language-variant specific tools (e.g., PoS taggers and tokenizers) in the preprocessing step may solve some of the problems for lexical and syntactic-based features; also, using a Brazilian wordnet (instead of a Portuguese wordnet) for learning in the Brazilian Portuguese may solve some of the problems regarding semantic-based features.

5.3. Binary Formulation

In the third evaluation scenario, we address the problem in a different perspective, motivated by the characteristics of the ASSIN corpus. As shown in Table 2, the distribution of classes in the ASSIN corpus is very unbalanced, with a much lower number of examples for the *Paraphrase* class. As introduced in Section 1, a *Paraphrase* can be formulated as a bidirectional entailment. In this experimental setup we formulate the problem of recognizing textual entailment as a binary classification problem between the classes *Entailment/Paraphrase* and *None*. The training set was built as follows: (a) each *Paraphrase* example from the ASSIN corpus was transformed into two new *Entailment* examples (i.e., *T* entails *H* and *H* entails *T*); and (b) the remaining *None* and *Entailment* examples from the ASSIN corpus were added. The test set comprises the same examples of the ASSIN corpus, where *Entailment* and *Paraphrase* classes are aggregated in the same class (*E + P*).

Results are shown in Table 7. The first two lines correspond to each of the target classes: *None* (*N*) and *Entailment/Paraphrase* (*E + P*). For each of the partitions (training and test set) of the ASSIN corpus containing annotations for European Portuguese, the first column presents the total number of samples used in the experiments and the following two columns correspond to accuracy and averaged micro F1-score, respectively (for the training set, these results were obtained after performing 10-fold cross validation). These results show that this binary classification task makes the decision boundaries easier to distinguish.

Table 7. Evaluation results for the binary classification setting.

	Train			Test		
	# Samples	Acc.	F1	# Samples	Acc.	F1
N	2046	0.87	0.89	1386	0.87	0.89
E + P	1179	0.84	0.81	614	0.81	0.77
total/avg	3225	0.86	0.86	2000	0.85	0.85

To compare this approach with the results obtained in the experiments previously described, we adapt this binary classification to the original multi-classification scenario (i.e., *None*, *Entailment* and *Paraphrases*) by making bidirectional predictions for each sentence pair. If the classifier predicts entailment in both directions, we assume the output to be *Paraphrase*; if it predicts entailment only from the text (*T*) to the hypothesis (*H*), the output is *Entailment*. Otherwise, the prediction is *None*. We obtained an averaged micro-F1 score of 0.91 for *None*, 0.62 for *Entailment* and 0.40 for *Paraphrase* corresponding to a total average micro-F1 score of 0.8 and a macro-F1 score of 0.64. In contrast with our intuition, the binary formulation setting does not seem to generalize well on the test set. We believe that the propagation of errors from the binary classifier has a negative effect in the second step of the

process (i.e., when we make predictions in both directions and apply a set of rules to determine the multi-class label).

Consequently, we conclude that the binary formulation presented in this section yields promising results for recognizing textual entailment (RTE) but the proposed algorithm to make predictions for RTE and paraphrases does not yield the best results. For RTE and paraphrases, the best performing system is the multi-class approach presented in Section 5.1.

6. Conclusions

In this paper, we present several approaches to address the NLP task of recognizing textual entailment and paraphrases from text written in the Portuguese language. We started by the natural formulation of this task as a multi-class classification problem. The overall results obtained following this setting are promising (with an accuracy of 0.835 in the test set). Comparing our results with those of the participants in the *ASSIN Challenge* [14], our approach obtains an overall accuracy close to the results obtained by the best performing system (0.8385 of accuracy obtained by Fialho et al. [33]) and outperforms all the proposed systems in the corresponding macro F1-score metric (the best performing system [33] reported 0.71 of macro F1-score). Additionally, we observe that the performance of our approach improved with semantic-based features, albeit not significantly. Notwithstanding, a detailed analysis points that this is one of the most promising directions for future work.

A closer assessment of our results shows that the number of annotated sentence pairs may not be sufficient to build a system that generalizes well for unseen data since the implemented classifiers tend to prefer labels that contain more training instances simply because they are more representative of the training data in statistical terms. As discussed in this paper, we believe that training with more quality data, balanced in relation to the different classes, and covering more entailment phenomena (e.g., by reasoning with quantifiers) is essential to obtain systems to address the task of RTE and paraphrases that generalizes better to unseen data.

Some subtasks could be added to help capturing relations of entailment and paraphrase, namely: reasoning over numerical and temporal expressions, dealing with missing values in the semantic resources employed, detecting negated expressions, named entity disambiguation, and semantic role labeling. By addressing these subtasks in future work, we believe that our results could be significantly improved.

To overcome the lack of annotated data and to improve the coverage of textual entailment phenomena captured in the *ASSIN* corpus, we aim to explore natural language generation techniques to synthesize new learning instances in a semi-automatic process, where annotators would validate the sentence pairs generated automatically. For instance, based on a given text sentence T (extracted from some textual resource), we aim to study techniques to generate artificial hypothesis sentences based on the knowledge extracted from semantic resources (e.g., wordnet relations and knowledge graphs). This approach to automatically generate learning instances follows similar guidelines as the ones defined in the construction of the *SICK* and *SNLI* corpora but, instead of asking to human annotators to manually provide the hypothesis sentences, we aim to interact with human annotators only in the validation step. These approaches have the advantage of generating balanced datasets and at the same time a large quantity of data. Providing a synthesized dataset containing considerable annotations of entailment relations is, in our perspective, one of the most promising directions of future work to train systems capable of better recognizing textual entailments in the Portuguese language.

Increasing the training set with the Brazilian Portuguese partition of the *ASSIN* corpus had an unexpected impact in the overall performance of the system. We associate this result to syntactic and semantic differences between European and Brazilian Portuguese and because some of the external resources that were employed (i.e., fuzzy wordnet, part-of-speech tagger, and word embeddings model) are based on the European Portuguese language. Consequently, some lexical, syntactic and semantic Brazilian Portuguese linguistic phenomena may be missing or misleading in this approach.

Furthermore, formulating the problem as a binary classification task seems to be adequate for recognizing textual entailment, but attempts to adapt the binary classification for recognizing textual entailments and paraphrases in a multi-class classification setting led to poor generalization.

In future work, we would like to enhance the semantic-based features employed in our system, including: metrics to evaluate semantic similarity between fragments of text using the fuzzy wordnet described in this paper, sentence-level representations (e.g., using a dependency parser) and more sophisticated computations using distributed representation models. Furthermore, we aim to study transfer learning techniques using deep neural network models (current state-of-the-art models for RTE from English text). The idea is to explore large-scale corpora annotated with relations of entailment from text written in English that is currently available. First, a deep neural network model will be trained on these large-scale annotated resources for recognizing textual entailment from text written in English. Next, we adapt the trained model by providing pre-trained word embeddings for Portuguese and by performing some retraining on the ASSIN corpus to obtain a ML model that is capable of using the knowledge gathered when training in the English corpora to make predictions for unseen data in the Portuguese language. Then, we aim to compare the performance of the obtained model with the feature-engineered ML models presented in this paper.

Acknowledgments: The first author was partially supported by a doctoral grant from Doctoral Program in Informatics Engineering (ProDEI) from the Faculty of Engineering of the University of Porto (FEUP).

Author Contributions: Gil Rocha has collected and analyzed the corpora used in this paper, has designed and implemented the methods proposed, and has run the experiments. Henrique Lopes Cardoso has supervised the work, namely for defining the methods proposed, designing the experimental settings and analyzing results. Both authors have contributed in reviewing the state-of-the-art and in writing the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Vorohej, M. *A Theory of Argument*; Cambridge University Press: Cambridge, UK, 2009.
2. Gizbert-Studnicki, T. Consensus and objectivity of legal argumentation. In *Argumentation 2012: International Conference on Alternative Methods of Argumentation in Law: Conference Proceedings*; Araszkievicz, M., Myška, M., Smejkalová, T., Šavelka, J., Škop, M., Eds.; Masaryk University: Brno, Czech Republic, 2012; Volume 423, pp. 1–13.
3. Dagan, I.; Roth, D.; Sammons, M.; Zanzotto, F.M. *Recognizing Textual Entailment: Models and Applications*; Synthesis Lectures on Human Language Technologies; Morgan & Claypool Publishers: Williston, VT, USA, 2013.
4. Cabrio, E.; Villata, S. Combining Textual Entailment and Argumentation Theory for Supporting Online Debates Interactions. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers, Jeju Island, Korea, 8–14 July 2012*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2012; Volume 2, pp. 208–212.
5. Sammons, M.; Vydiswaran, V.; Roth, D. Recognizing Textual Entailment. In *Multilingual Natural Language Applications: From Theory to Practice*; Bikel, D.M., Zitouni, I., Eds.; Prentice Hall: Upper Saddle River, NJ, USA, 2012; pp. 209–258.
6. Androutsopoulos, I.; Malakasiotis, P. A Survey of Paraphrasing and Textual Entailment Methods. *J. Artif. Int. Res.* **2010**, *38*, 135–187.
7. Dagan, I.; Glickman, O.; Magnini, B. The PASCAL Recognising Textual Entailment Challenge. In *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Textual Entailment*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2006; pp. 177–190.
8. Mollá, D.; Vicedo, J.L. Question Answering in Restricted Domains: An Overview. *Comput. Linguist.* **2007**, *33*, 41–61.
9. Moens, M.F. *Information Extraction: Algorithms and Prospects in a Retrieval Context*; Springer: Dordrecht, The Netherlands, 2009.
10. Madnani, N.; Dorr, B.J. Generating Phrasal and Sentential Paraphrases: A Survey of Data-driven Methods. *Comput. Linguist.* **2010**, *36*, 341–387.

11. Padó, S.; Galley, M.; Jurafsky, D.; Manning, C. Robust Machine Translation Evaluation with Entailment Features. In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP, Suntec, Singapore, 2–7 August 2009; ACL: Stroudsburg, PA, USA, 2009; Volume 1, pp. 297–305.
12. Lippi, M.; Torrioni, P. Argumentation Mining: State of the Art and Emerging Trends. *ACM Trans. Internet Technol.* **2016**, *16*, 10:1–10:25.
13. Rocha, G.; Lopes Cardoso, H.; Teixeira, J. ArgMine: A Framework for Argumentation Mining. In Proceedings of the 12th International Conference Computational Processing of the Portuguese Language, Student Research Workshop, Tomar, Portugal, 13–15 July 2016.
14. Fonseca, E.; Santos, L.; Criscuolo, M.; Aluísio, S. ASSIN: Avaliação de Similaridade Semântica e Inferência Textual. In Proceedings of the 12th International Conference on Computational Processing of the Portuguese Language, Tomar, Portugal, 13–15 July 2016.
15. Fellbaum, C. (Ed.) *WordNet: An Electronic Lexical Database*; Language, Speech, and Communication; MIT Press: Cambridge, MA, USA, 1998.
16. Rocktäschel, T.; Grefenstette, E.; Hermann, K.M.; Kociský, T.; Blunsom, P. Reasoning about Entailment with Neural Attention. *arXiv* **2015**, arXiv:1509.06664.
17. De Marneffe, M.C.; Rafferty, A.N.; Manning, C.D. Finding Contradictions in Text. In Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics, Columbus, OH, USA, 15–20 June 2008.
18. Lai, A.; Hockenmaier, J. Illinois-LH: A Denotational and Distributional Approach to Semantics. In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), Dublin, Ireland, 23–24 August 2014; ACL: Dublin, Ireland, 2014; pp. 329–334.
19. Bos, J.; Markert, K. Recognising Textual Entailment with Logical Inference. In Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing, Vancouver, BC, Canada, 6–8 October 2005; ACL: Stroudsburg, PA, USA, 2005; pp. 628–635.
20. Beltagy, I.; Roller, S.; Cheng, P.; Erk, K.; Mooney, R.J. Representing Meaning with a Combination of Logical and Distributional Models. *Comput. Linguist.* **2016**, *42*, 763–808.
21. Pakray, P.; Neogi, S.; Bhaskar, P.; Poria, S.; Bandyopadhyay, S.; Gelbukh, A.F. A Textual Entailment System Using Anaphora Resolution. In Proceedings of the Text Analysis Conference (TAC), Gaithersburg, MD, USA, 2011, 14–15 November 2011; NIST: Gaithersburg, MD, USA, 2011.
22. Bentivogli, L.; Dagan, I.; Dang, H.T.; Giampiccolo, D.; Magnini, B. Fifth PASCAL Recognizing Textual Entailment Challenge. In Proceedings of the Text Analysis Conference, Gaithersburg, MD, USA, 16–17 November 2009.
23. Marelli, M.; Bentivogli, L.; Baroni, M.; Bernardi, R.; Menini, S.; Zamparelli, R. SemEval-2014 Task 1: Evaluation of Compositional Distributional Semantic Models on Full Sentences through Semantic Relatedness and Textual Entailment. In Proceedings of the 8th International Workshop on Semantic Evaluation, COLING, Dublin, Ireland, 23–24 August 2014; Nakov, P., Zesch, T., Eds.; ACL: Vancouver, BC, Canada, 2014; pp. 1–8.
24. Marelli, M.; Menini, S.; Baroni, M.; Bentivogli, L.; Bernardi, R.; Zamparelli, R. A SICK Cure for the Evaluation of Compositional Distributional Semantic Models. In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), Reykjavik, Iceland, 26–31 May 2014; Chair, N.C.C., Choukri, K., Declerck, T., Loftsson, H., Maegaard, B., Mariani, J., Moreno, A., Odijk, J., Piperidis, S., Eds.; European Language Resources Association (ELRA): Reykjavik, Iceland, 2014.
25. Nangia, N.; Williams, A.; Lazaridou, A.; Bowman, S.R. The RepEval 2017 Shared Task: Multi-Genre Natural Language Inference with Sentence Representations. *arXiv* **2017**, arXiv:1707.08172.
26. Williams, A.; Nangia, N.; Bowman, S.R. A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. *arXiv* **2017**, arXiv:1704.05426.
27. Bowman, S.R.; Angeli, G.; Potts, C.; Manning, C.D. A large annotated corpus for learning natural language inference. *arXiv* **2015**, arXiv:1508.05326.
28. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. BLEU: A Method for Automatic Evaluation of Machine Translation. In Proceedings of the 40th Annual Meeting Association Computational Linguistics, Philadelphia, PA, USA, 7–12 July 2002; ACL: Stroudsburg, PA, USA, 2002; pp. 311–318.
29. Magnini, B.; Zanolini, R.; Dagan, I.; Eichler, K.; Neumann, G.; Noh, T.; Padó, S.; Stern, A.; Levy, O. The Excitement Open Platform for Textual Inferences. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore, MD, USA, 22–27 June 2014; pp. 43–48.

30. Hartmann, N.S. Solo Queue at ASSIN: Combinando Abordagens Tradicionais e Emergentes. *Linguamática* **2016**, *8*, 59–64.
31. Sparck Jones, K. Chapter A Statistical Interpretation of Term Specificity and Its Application in Retrieval. In *Document Retrieval Systems*; Taylor Graham Publishing: London, UK, 1988; pp. 132–142.
32. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases and Their Compositionality. In Proceedings of the 26th International Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–10 December 2013; Curran Associates Inc.: Red Hook, NY, USA, 2013; pp. 3111–3119.
33. Fialho, P.; Marques, R.; Martins, B.; Coheur, L.; Quaresma, P. INESC-ID@ASSIN: Medição de Similaridade Semântica e Reconhecimento de Inferência Textual. *Linguamática* **2016**, *8*, 33–42.
34. Lin, C.Y.; Och, F.J. Automatic Evaluation of Machine Translation Quality Using Longest Common Subsequence and Skip-bigram Statistics. In Proceedings of the 42nd Annual Meeting Association for Computational Linguistics, Barcelona, Spain, 21–26 July 2004.
35. Oliveira Alves, A.; Rodrigues, R.; Gonçalo Oliveira, H. ASAPP: Alinhamento Semântico Automático de Palavras aplicado ao Português. *Linguamática* **2016**, *8*, 43–58.
36. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780.
37. Bowman, S.R. Modeling Natural Language Semantics in Learned Representations. Ph.D. Thesis, Stanford University, Stanford, CA, USA, 2016.
38. Consortium, T.F.; Cooper, R.; Crouch, D.; Eijck, J.V.; Fox, C.; Genabith, J.V.; Jaspars, J.; Kamp, H.; Milward, D.; Pinkal, M.; et al. Using the Framework. 1996. Available online: <https://files.ifi.uzh.ch/cl/hess/classes/seminare/interface/framework.pdf> (accessed on 28 March 2018).
39. Young, P.; Lai, A.; Hodosh, M.; Hockenmaier, J. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Trans. Assoc. Comput. Linguist.* **2014**, *2*, 67–78.
40. Levy, O.; Dagan, I.; Goldberger, J. Focused Entailment Graphs for Open IE Propositions. In Proceedings of the Eighteenth Conference on Computational Natural Language Learning, Baltimore, MD, USA, 26–27 June 2014; pp. 87–97.
41. Pavlick, E.; Rastogi, P.; Ganitkevitch, J.; Durme, B.V.; Callison-Burch, C. PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics, Beijing, China, 26–31 July 2015; pp. 425–430.
42. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent Dirichlet Allocation. *J. Mach. Learn. Res.* **2003**, *3*, 993–1022.
43. Garcia, M.; Gamallo, P. Yet Another Suite of Multilingual NLP Tools. In *Languages, Applications and Technologies. Communications in Computer and Information Science*; José-Luis Sierra-Rodríguez, J.P.L., Simões, A., Eds.; Springer: Cham, Switzerland, 2015; Volume 563, pp. 65–75.
44. Gonçalo Oliveira, H. CONTO.PT: Groundwork for the Automatic Creation of a Fuzzy Portuguese Wordnet. In Proceedings of the 12th International Conference on Computational Processing of the Portuguese Language, Tomar, Portugal, 13–15 July 2016; Silva, J., Ribeiro, R., Quaresma, P., Adami, A., Branco, A., Eds.; Springer: Cham, Switzerland, 2016; pp. 283–295.
45. Al-Rfou, R.; Perozzi, B.; Skiena, S. Polyglot: Distributed Word Representations for Multilingual NLP. In Proceedings of the Seventeenth Conference on Computational Natural Language Learning, Sofia, Bulgaria, 8–9 August 2013; pp. 183–192.
46. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.

