Wide And Deep Transformers Applied to Semantic Relatedness and Textual Entailment

Evandro Fonseca, João Paulo Reis Alvarenga

STILINGUE
{evandro, joaopaulo}@stilingue.com.br

Abstract. In this paper we present our approach to deal with semantic relatedness and textual entailment, two tasks proposed in ASSIN-2 (Second evaluation of semantic relatedness and textual entailment). We develop 18 features that explore lexical, syntactic and semantic information. To train the models we applied both supervised machine learning and an architecture based in Wide and Deep learning. Our proposal demonstrated to be competitive with the current state-of art models and with other participant models for Portuguese, mainly when the mean square error is considered.

Keywords: Semantic Relatedness,Textual Entailment, Natural Language Processing

1 Introduction

In this paper we present an approach to deal with Semantic Relatedness assertion and Textual Entailment Recognition. The Semantic Relatedness task (SR) is a process that measures the degree of semantic relatedness of a sentence pair by assigning a relatedness score ranging from 1 (completely unrelated) to 5 (very related). For example, in the pair (1) [Um homem está tocando o trompete – Alguém está brincando com um sapo] *[A man is playing the trumpet – Someone is playing with a frog]* the defined value is 1. It is because the sentence pair relates very distinct topics. In (2) [Um lêmure está lambendo o dedo de uma pessoa – Um lêmure está mordendo o dedo de uma pessoa] *[A lemur is licking a person's finger – A lemur is biting a person's finger]* we can see that the meaning is not the same, however the sentences share some aspects. So, we can infer that the relatedness value is close to 3[1]. And, in cases like: (3) [Um garoto está fazendo um discurso – Um garoto está falando] *[A boy is giving a speech – A boy is talking]* the relatedness value is close to 5. Textual Entailment (TE) consists of recognizing when a sentence "A" entails or not a sentence "B". In other words, this task consists in defining when we may conclude B from A. Thus, when we consider the Textual Entailment and the previous examples ((1), (2) and (3)) we can assert, respectively, "none","none" and "entailment". Both Semantic Relatedness and the Textual Entailment are very important tasks and also a

[1] Samples collected from ASSIN-2 test corpus

great challenges, it is because depends of many processing levels, such as: Part-of-speech tagging, Sentiment Analysis, Coreference Resolution, among others. Plus, when we deal with less resourceful languages like Portuguese, these challenges are even greater, due to lack of dense semantic bases, such as YAGO[19] and FrameNet[2]. The paper is organized as follows: Section 2 presents related work; in Section 3 we describe our proposed models; in Section 4 we show a summary of the official ASSIN-2[16] results; Section 5 presents an error analysis; in Section 6 the conclusions and future work are presented.

2 Related Work

Transfer learning technique is widely used by many NLP tasks, such as Sentiment Analysis [14], Text Classification [22], Question Answering [20] among others. The reason of that is clear. Transfer learning models may improve significantly NLP models [10]. For SR and TE tasks it is not different. In 2018 Devlin et al. [6] has proposed an approach based on transfer learning (BERT) to solve SR and TE tasks for English. They achieved 0.865 of Pearson for SR and 70.1% of F1 for TE in GLUE Benchmark [21]. In 2019 some works based on BERT architecture were emerged (also for English), such as: RoBERTa [13], whose results were 0.922 of Person for SR and 88.2% of F1 for TE; and ALBERT [11], with 0.925 of Pearson for SR and 89.2% of F1 for TE. The current state of art for Portuguese is Fonseca's work [7]. Fonseca has proposed the use of neural networks and syntactic trees distance to solve SR and TE tasks and as a result his model has achieving 0.577 of Pearson for SR and 74.2% of F1 for TE using ASSIN-2 corpus. In this competition we called his model of baseline. In our approach we propose the use of a machine learning architecture named Wide And Deep. Wide and Deep architecture consists of unifying handcrafted features with dense features. Cheng et al. [5] has proposed the use of Wide and Deep to deal with recommender systems and their results were encouraging. Plus, we believe that handcrafted features, based in linguistic knowledge, NLP techniques and in a corpora study may outperform pure deep learning features. However, when we apply handcrafted features only the results may be not so good. To show that, we train and test models using both Wide and Deep architecture and the traditional supervised machine learning architecture. The result shows that Wide and Deep architecture may outperforms significantly the traditional machine learning architecture with handcrafted features.

3 Proposed model

To address the semantic relatedness and textual entailment problem we propose eighteen features, which consists of exploring the lexical, syntactic and semantic information. Besides, we use Wide and Deep Transformer architecture, which mix our proposed features and deep learning features. In subsection 3.1 we show our set of propose features. Our set of features is based on some related works[7]

and also is empirically designed, through a case study based in ASSIN-2 training set[2].

3.1 Features

1. **Sentiment Agreement:** returns true when both sentences agree in the sentiment polarity[1] and false otherwise (as in below example).
   - O animal está comendo – *The animal is eating* (+)
   - O animal está mordendo uma pessoa – *The animal is biting a person* (-)
2. **Negation Agreement:** returns true when the both sentences agree in the co-occurrence of negative terms[3] or expressions, such as: "jamais", "nada", "nenhum", "ninguém", "nunca", "não", among others. This feature is very relevant for textual entailment. It helps in cases such as:
   - O menino está pulando – *The boy is jumping*
   - **Ninguém** está pulando – *Nobody is jumping*
3. **Synonym:** returns the quantity of synonyms between the two sentences. To identify it we use Onto.PT [8]. This feature helps to improve the semantic relatedness process. It is because synonyms are used to refer to a same entity, as in below example:
   - Um **garoto** está fazendo um discurso – *A young man is giving a speech*
   - Um **menino** está falando – *A boy is talking*
4. **Hyponym:** returns the quantity of hyponyms between the two sentences. As in the Synonymy feature, we use Onto.PT to identify the semantic relations.
5. **Verb Similarity:** returns the number of similar verbs between two sentences. To recognize it, Onto.PT and VerbNet.Br[17] were used. It helps to identify pairs such as:
   - Uma menina está **caminhando** – *A girl is stepping*
   - Uma menina está **andando** – *A girl is walking*
6. **Nouns Similarity:** returns the quantity of similar nouns between two sentences. Here we use synonymy relation provided by Onto.PT and the lexical similarity(when two words is exactly equals).
   - O **garoto** está em **casa** – *The young man is in home*
   - O **menino** está em **casa** – *The boy is in home*
7. **Adjectives Similarity:** returns the quantity of similar adjectives between two sentences. As in Nouns Similarity, we use synonymy relation and the lexical similarity, but considers just adjectives.
8. **Gender:** returns the number of tokens that agree in gender (male/female).

[2] available in: https://sites.google.com/view/assin2/
[3] never, nothing, no, nobody, no one, never,...

9. **Number:** returns the number of tokens that agree in number (singular/plural). To identify number and gender features we use SNLP[4]
10. **Jaccard Similarity:** returns a real number, containing the Jaccard[12] similarity between two sentences. Here we perform a preprocessing: firstly we remove determinants[5]; second we sort the tokens alphabetically[6]; and, finally, we calculate the Jaccard similarity. Basically, each sentence is modified as in the follow example:
    - A mulher está cortando cebola → cebola cortando está mulher
    - *The woman is cutting onion → cutting is onion woman*
11. **Verb+Participle:** returns true when both the sentences have a verb+participle construction,, which do not necessarily have to be equal, as in:
    - O urso **está sentado** – *The bear is sitting*
    - O urso **está deitado** – *The bear is lying down*
12. **Verb+Participle+Equals:** returns true when both the sentences have the same verb+participle construction, as in:
    - O urso **está sentado** – *The bear is sitting*
    - O urso **está sentado** – *The bear is sitting*
13. **Conjunction_E_A:** returns true when the sentence "A" has the "e"*(and)* conjunction, which helps in cases such as:
    - Um menino **e** uma menina estão caminhando – *A boy and a girl are walking*
    - Duas pessoas estão andando – *Two people are walking*
14. **Conjunction_E_B:** the same as Conjunction_E_A, but for sentence B.
15. **TokensDif:** calculates the difference in the amount of tokens between the sentences "A" and "B". It does not consider determinants. In the below example, TokensDif returns 2, because sentence A has six tokens and sentence B has four tokens[7];
    - Uma **mulher não está fritando algum alimento** – *A woman is not frying any food*
    - Uma **mulher está fritando comida** – *A woman is frying food*
16. **Same Word:** returns an integer value, containing the number of exactly equal words in the sentences(common words). Here, we consider just verbs, nouns and adjectives and apply just lexical match.
17. **Same Subject:** returns true when the sentences has the same subject.

[4] Stilingue proprietary software.
[5] Although determinants may change a referent, in ASSIN-2 shared-task there is an agreement that consists of considering, for example "the girl" and "a girl" the same entity.
[6] it is because, to calc Jaccard we want to consider just the tokens, not its sequence in the sentence.
[7] determinants are not considered

18. **Cosine Similarity:** returns the cosine similarity[8] of two sentences. Here we use FastText Skip-Gram 300d built by NILC[9] [9].

3.2 Model Set Up and Runs

In the shared task, each participant was encouraged to submit three output files. Each output file could have results of one or the two proposed tasks. We performed experiments using three distinct configurations to produce the models. For the first model we use the traditional supervised machine learning. Basically we train a model using Random Forest[3] and our set of proposed features. For the second and third models the Wide And Deep architecture was used. For that, we use BERT-Base multilingual [6], Universal Sentence Encoder-Large multilingual[4] and our set of proposed features. In table 1 we detail the set up of each model.

Table 1. Trained models

Model	Wide And Deep	Random Forest	Bert-Base	Universal Sentence Encoder
1		x		
2	x			x
3	x		x	

Using the proposed models we perform three runs, considering the two tasks, as in table 2:

Table 2. Runnings and models

Run	Textual Entailment	Semantic Relatedness
	Model	
1	1	1
2	3	2
3	3	3

Basically, in the first run we use just Random Forest and our set of features. We tested some other traditional supervised machine learning algorithms, such as Multilayer Perceptron, Linear Regression, Naive Bayes, Decision Table, J48, Random Tree, among others. However, Random Forest easily outperforms all of then. In the second and third runs we use Wide and Deep architecture. We can see that the model three was used in second and third run. It is because in our tests we have not found a model that outperforms Bert-Base for textual entailment task.

[8] We calc Cosine Similarity considering averaged word vectors of each sentence
[9] http://nilc.icmc.usp.br/embeddings

4 Results

In table 3 we show results[10] of ASSIN-2 shared-task for the two proposed tasks. There is a great distance between Wide and Deep architecture and the traditional supervised machine learning. Regarding our model and the best models(winners), our models presents very close results. Basically, our model achieved 1.7 points less in F1 and 1.68 less accuracy for TE task. Regarding SR task, our model presented 0.009 less Pearson coefficient (for running 3) and 0.026 for running 2. However, we can see that out model presents better mean squared error (MSE). It is known that the MSE penalizes outliers. Thus, we can say that our model is more linear than others. An error analysis is presented in Section 5.

Table 3. ASSIN-2 results

Team	Run	Textual Entailment		Semantic Relatedness	
		F1	Accuracy	Pearson	MSE
Stilingue (**Our**)	1	0.788	78.84	0.748	0.53
	2	0.866	86.64	0.800	**0.39**
	3	0.866	86.64	0.817	0.47
IPR	1	0.876	87.58	**0.826**	0.52
Deep Learning Brasil	1	**0.883**	**88.32**	0.785	0.59
Baseline [7]	1	0.742	74.18	0.577	0.75

5 Error Analysis

In table 4 it is possible to see that there are over 1550 pairs with a range very near of the gold samples (ranges between 0 and 0.4); for 0.5 to 0.9 there are 618 pairs. It is important to say that this difference is acceptable. It is because, even in the annotation process, many human annotators disagree on this range. We also can see that for all 2448 pairs of test corpus, there is just one sample with a range above of 3.0. In this pair there are many equal words, however they refer to distinct facts. For the example below, our model has predicted 4.5 of similarity while gold is 1.5.

- um cachorro preto e um branco estão correndo alegremente na grama – *a black and a white dog are running happily in the grass*
- uma pessoa negra vestindo branco está correndo alegremente com o cachorro na grama – *a black person wearing white is running happily with the dog on the grass*

[10] for baseline model we unify its runs and shows only its better results

Table 4. Semantic Relatedness error range

Error Range	Stances
$0.0 \sim 0.4$	1559
$0.5 \sim 0.9$	618
$1.0 \sim 1.9$	261
$2.0 \sim 2.9$	9
$3.0 \sim 5.0$	1

Regarding TE task we found two main errors: the first refers to cases which we have referential expressions, such as:

- Um chefe mexicano está **preparando uma refeição** – *A mexican chef is preparing a meal*
- Um chefe mexicano está **cozinhando** – *A mexican chef is cooking*
- Um menino está **fazendo um discurso** – *A boy is giving a speech*
- Um menino está **falando** – *A boy is speaking*

We identify that there is a limitation in our model. It is because our synonymy feature just consider single words. The second main error found is when a sentence "A" has verb + participle construction and the sentence "B" has gerund and vice-versa, such as:

- O pelo de um gato está sendo **penteado** por uma garota – *A cat's fur is being combed by a girl*
- Uma pessoa está **penteando** o pelo de um gato – *A person is combing a cat's fur*
- O cara está **comendo** uma banana – *The guy is eating a banana*
- Uma banana está sendo **comida** por um cara – *A banana is being eaten by a guy*

6 Conclusion and Future Work

In this paper we presented our models to deal with two important tasks, Semantic Relatedness and Textual Entailment. Our models were based in 18 features, that cover natural language patterns and Wide and Deep architecture. The latter explores the mix between our linguistic features and deep learning features. As results we show that our models can be competitive. Plus, although the MSE is not the official metric, we believe that our model built for semantic relatedness task provides a good solution for the proposed task, mainly when we need a more reliable model, with less outliers. As future work we want to improve our semantic features, in order to recognizes referential expressions. We also intend to explore ConceptNet [18] e BabelNet [15] to provide a more robust semantic knowledge to our models.

References

1. L. V. Avanço and M. d. G. V. Nunes. Lexicon-based sentiment analysis for reviews of products in Brazilian Portuguese. In *2014 Brazilian Conference on Intelligent Systems*, pages 277–281. IEEE, 2014.
2. C. F. Baker, C. J. Fillmore, and J. B. Lowe. The Berkeley Framenet project. In *Proceedings of the 17th International Conference on Computational Linguistics*, pages 86–90, Quebec, Canada, 1998.
3. R. R. Bouckaert, E. Frank, M. Hall, R. Kirkby, P. Reutemann, A. Seewald, and D. Scuse. Weka manual for version 3-7-8, 2013.
4. D. Cer, Y. Yang, S. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, Y. Sung, B. Strope, and R. Kurzweil. Universal sentence encoder. *CoRR*, abs/1803.11175, 2018.
5. H.-T. Cheng, L. Koc, J. Harmsen, T. Shaked, T. Chandra, H. Aradhye, G. Anderson, G. Corrado, W. Chai, M. Ispir, et al. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*, pages 7–10. ACM, 2016.
6. J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.
7. E. R. Fonseca. *Reconhecimento de implicação textual em português*. PhD thesis, Universidade de São Paulo, 2018.
8. H. Gonçalo Oliveira and P. Gomes. ECO and Onto.PT: A flexible approach for creating a Portuguese wordnet automatically. *Language Resources and Evaluation*, 48(2):373–393, 2014.
9. N. Hartmann, E. Fonseca, C. Shulby, M. Treviso, J. Rodrigues, and S. Aluisio. Portuguese word embeddings: Evaluating on word analogies and natural language tasks. *arXiv preprint arXiv:1708.06025*, 2017.
10. J. Howard and S. Ruder. Fine-tuned language models for text classification. *CoRR*, abs/1801.06146, 2018.
11. Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut. ALBERT: A lite BERT for self-supervised learning of language representations. *CoRR*, abs/1909.11942, 2019.
12. M. Levandowsky and D. Winter. Distance between sets. *Nature*, 234(5323):34–35, 1971.
13. Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019.
14. A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 142–150, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.
15. R. Navigli and S. P. Ponzetto. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250, 2012.
16. L. Real, E. Fonseca, and H. Gonçalo Oliveira. The ASSIN 2 shared task: Evaluating Semantic Textual Similarity and Textual Entailment in Portuguese. In *Proceedings of the ASSIN 2 Shared Task: Evaluating Semantic Textual Similarity and Textual Entailment in Portuguese*, CEUR Workshop Proceedings, page [In this volume]. CEUR-WS.org, 2020.

17. C. E. Scarton. Verbnet.BR: construção semiautomática de um léxico verbal online e independente de domínio para o português do brasil. 2013.
18. R. Speer and C. Havasi. Representing general relational knowledge in conceptnet 5. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation*, pages 3679–3686, 2012.
19. F. M. Suchanek, G. Kasneci, and G. Weikum. YAGO: a core of semantic knowledge. In *Proceedings of the 16th International Conference on World Wide Web*, pages 697–706, Banff, AB, Canada, 2007.
20. E. M. Voorhees. The trec question answering track. *Nat. Lang. Eng.*, 7(4):361–378, Dec. 2001.
21. A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *CoRR*, abs/1804.07461, 2018.
22. X. Zhang, J. Zhao, and Y. LeCun. Character-level convolutional networks for text classification. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, pages 649–657, Cambridge, MA, USA, 2015. MIT Press.