INGEOTEC at SemEval-2024 Task 1: Bag of Words and Transformers

Daniela Moctezuma[†] and **Eric S. Tellez**[‡] and **Mario Graff**[‡]
[†] CentroGEO, Aguascalientes, México
[‡] CONACyT - INFOTEC, Aguascalientes, México
dmoctezuma@centrogeo.edu.mx
{eric.tellez,mario.graff}@infotec.mx

Abstract

Understanding the meaning of a written message is crucial in solving problems related to Natural Language Processing; the relatedness of two or more messages is a semantic problem tackled with supervised and unsupervised learning. This paper outlines our submissions to the Semantic Textual Relatedness (STR) challenge at SemEval 2024, which is devoted to evaluating the degree of semantic similarity and relatedness between two sentences across multiple languages. We use two main strategies in our submissions. The first approach is based on the Bag-of-Word scheme, while the second one uses pre-trained Transformers for text representation. We found some attractive results, especially in cases where different models adjust better to certain languages over others.

1 Introduction

Semantics refers to the meaning of language, including words, phrases, sentences, and overall text. Understanding semantics is essential for text comprehension and communication, as it allows us to interpret the intended meaning of a message accurately. Semantic relatedness measures how similar the meaning of two words or phrases is. It is based on the idea that words related in meaning tend to co-occur frequently in language or even have some causal relation connecting them. For example, *cat* and *dog* are semantically related because they refer to common household pets. Measuring semantic relatedness is essential for many natural language processing tasks, such as information retrieval, question answering, and machine translation.

Relatedness models play a crucial role in natural language processing (NLP). These models determine the degree of similarity or relatedness between two pieces of text. One of the critical benefits of relatedness models is that they can help improve the performance of NLP applications by providing more relevant and accurate results. For example, relatedness models can be used in information retrieval to rank search results based on their relevance to the user's query. Similarly, relatedness models can help identify the most relevant answer to a user's question while solving the question-answering problem.

A method based on corpus-based word similarity and string similarity, as well as their order, is proposed in (Islam and Inkpen, 2008). For string similarity, the authors used the longest common subsequence (LCS) in three ways to weight, i.e., the work is based on measuring the shared order of words. The word mover's distance, see (Kusner et al., 2015), reformulates the problem of comparing two sequences of words to an optimal transportation problem. It represents two sentences with its word embeddings and computes its optimal alignment using a dynamic programming solution; while it is pretty promising, it does not require sentences to be of some fixed size and works with a myriad of possible word embeddings. However, the technique was revisited by (Sato et al., 2022) and found diverse issues that limit its effectiveness.

Kenter and De Rijke (Kenter and de Rijke, 2015) have used word embeddings (word2vec) and external sources of semantic knowledge to represent text messages and meta-features. They aim to interpret proximity in the generated latent space as semantic similarity.

More recently, the Transformer deep neural networks have become a powerful alternative to both lexical and semantic approaches; the approach is based on a stack of encoders and decoders layers and the self-attention procedure (Vaswani et al., 2017). Transformers have a high computational cost, primarily for training. The first Transformer

Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), pages 1155–1159
June 20-21, 2024 ©2024 Association for Computational Linguistics

that can be pre-trained and fine-tuned to match different tasks is BERT (Devlin et al., 2018a); after BERT, the high cost of training is paid once since fine-tuning needs less computational power and much less data. The interested reader should review the BERT manuscript and the seminal paper about pre-training NLP models (Howard and Ruder, 2018).

Fine-tuning BERT for classification or regression tasks is straightforward, not because it is a simple architecture but due to the myriad of literature, repositories, and examples showing how to do it [1]. However, its usage for sentence similarity needs to produce a vector that works fine for the task, and it is not trivial to produce one with its standard matrix output. The sentence transformers (Reimers and Gurevych, 2019) use siamese networks to create effective sentence vector embeddings for tasks working with pairs of sentences, for instance, similarity search and clustering.

In their research, Chandrasekaran and Mago (Chandrasekaran and Mago, 2021) have surveyed the evolution of semantic similarity methods, reviewing various NLP approaches, including traditional techniques and those found in machine learning and deep learning. They have provided a detailed study describing the strengths and weaknesses of each approach.

A binary version of the relatedness tasks is as follows: given a pair of sentences u and v, predicting true if u and v are related and false otherwise. A more elaborated task is to predict a relatedness score $\mathsf{rel}(u, v) \in [0..1]$, where values near zero mean for no relation and values near 1 mean for total relatedness. The latter definition is used in the Semantic Text Relatedness Task 1 (Ousidhoum et al., 2024b) at SemEval-2024, which asked for predicting relatedness scores for nine multilingual datasets; in particular, we tackled the problem as a supervised learning problem, i.e., we focused only on subtask 1 using the data for the nine languages(for more details about dataset see (Ousidhoum et al., 2024a)).

This document outlines the strategies we employed for the Semantic Textual Relatedness (STR) challenge in SemEval 2024, specifically the track A for the nine languages considered. To tackle this task, we utilized two distinct approaches: a transformer method for the English and Spanish languages, and an EvoMSA (Graff et al., 2020) solution for the remaining languages, which include Algerian Arabic, Amharic, Hausa, Kinyarwanda, Marathi, Moroccan Arabic, and Telugu.

The paper is organized as follows: Section 2 describes all our solutions to task 1. Section 3 shows our experimental results. Finally, Section 4 concludes our results and findings.

2 System overview

Nowadays, one of the most common approaches to dealing with natural language processing (NLP) problems is those Transformer-based language models. However, the pre-training procedure of this kind of language model needs a vast text corpus, and therefore, it may be impossible now to train them properly in many languages. In these cases, models based on counting and computing statistics may be more robust. We used Transformers for languages we know have large language models explicitly created for that language; for other datasets, we use a back-propagation optimized EvoMSA model for each one.

2.1 Out transformer-based approach

Our model was trained as a regression using the following procedure. For each pair, we extracted the sentence embedding for each sentence and evaluated the cosine similarity between pairs of embeddings. We trained a linear Support Vector Machine regressor using the cosine similarity to learn and predict the given relatedness score.

We tested several Transformer models but chose those that gave us the best performance, all of them were used directly as Hugging Face indicated. In this case, the best ones were microsoft-mpnet-base(Song et al., 2020) and multilingual BERT (Devlin et al., 2018b).

The microsoft-mpnet-base (MPNet) is a pre-training model, it tries to deal with the dependency on the predicted tokens and takes auxiliary position info into account to see a full sentence and reduce the position difference (Song et al., 2020).

The multilingual BERT is a well-known transformers model pre-trained on a large corpus of multilingual data self-supervised. In overview, it has two main tasks, MLM (Masked Language Modeling) and NSP (Next Sentence Prediction) (Devlin et al., 2018b), nevertheless, we just used the embedding representation to deal with the competition's task.

[1] For instance, one of the main sources of pre-trained Transformer models and documentation about them is the Hugging face project `huggingface.co`

2.2 Our EvoMSA approach

We use our EvoMSA framework for languages different than English and Spanish. EvoMSA models can be tailored for the dataset or pre-trained. Our pre-trained models were constructed using a small tweet corpus per language collected from the public Twitter stream. In addition, our EvoMSA models can be lexical based on bag-of-words (BoW) or semantic based on creating embeddings using numerous pre-trained classifiers in several self-supervised problems. Our BoW model produces highly sparse vectors where each component represents a token in the vocabulary. At the same time, our semantic representation (Dense) produces dense vectors created with the decision function of several binary classifiers, each one learned in a set of self-supervised tasks. The precise construction of EvoMSA models is detailed in (Graff et al., 2023).

Our approach to tackle the relatedness problem is to state it as a regression problem combining BoW and Dense representations using the following expression:

$$V = \left(S^\top \cdot S_Q, T^\top \cdot T_Q, (D \odot D_Q) \cdot \theta_1\right) \quad (1)$$

$$\hat{V} = \sigma\left(\frac{V}{\|V\|}\theta_2 + \beta\right) \quad (2)$$

where S, S_Q, T, and T_Q are sparse BoW matrices encoding pairs of sentences with statistics from pre-trained vocabularies (S) and training set-based vocabularies (T); D and D_Q are Dense matrices corresponding to pair of sentences, again computed with models pre-trained. Matrices without sub-indices mean for the first sentence in the pair, and those matrices with sub-indices Q mean for the second pair's element. The trainable parameters θ_1 and θ_2 are vectors, and β is a trainable scalar. Also, σ is the sigmoid function. We use differential programming with the JAX framework (Bradbury et al., 2018) for the Python programming language to train our models using $1 - \mathsf{pearson\_correlation}(\cdot, \cdot)$ as a loss function. In particular, we initialize θ_2 and β as the optimized parameters of a Linear Support Vector Machine parameters (solved firstly per each model with these parameters) and then θ_1 as a vector of ones instead of the typical random initialization to help on fine-tuning parameters. We call this model as One+.

We performed multiple modifications to this scheme and also found that defining V as $(T \odot T_Q) \cdot \theta_1$ results in a very competitive option. This model is called One-B. Note that this approach works only with the training set and does not require any pre-trained models.

3 Experimental results

We considered our two approaches with several expression variants for our EvoMSA-based approach and several models for our Transformer-based approach. Our model selection finds the best models using $1 - \mathsf{pearson\_correlation}$ with k-folds cross-validation along multilingual datasets. We selected the One+ and One-B model expressions since they demonstrated to be robust among many others coming from One+, also note that One-B works only with the training set.

In particular, the Transformer approach was better for Spanish and English datasets. We tested with several BERT, SBERT, and MPNet models before selecting *microsoft/mpnet-base* model for English and the multilingual BERT model, specifically the *bert-base-multilingual-cased*.[2]

Table 1 lists our best approaches for the different languages for the relatedness tasks in the third column. We can observe how transformers work fine for English and Spanish, languages with plenty of available models and data. For the rest of the languages, our EvoMSA approach performs better, but we can also observe that the simpler model One-B performs better in several datasets; this may be because of the lack of pre-trained models for that language, in particular, for languages with low available resources.

Table 1 also reports the Spearman correlation score and the global rank under the *dev* and *eval* datasets. Here, we can observe how our approach achieves different language ranks. In particular, we reached among the top ten results for Algerian and Moroccan Arabic. The English model is not among the top, but the score is not very different from the best ones. Note how One-B is competitive for Amharic, Hausa, Kinyarwanda, and Telugu, working without additional data.

It is important to say that, we did not achieve outstanding results, so, further analysis cannot be done, we saw lower results in those languages less studied, and more generalized models performed better in most common languages such as English and Spanish. Also, in the case of less-known languages, a simpler strategy was the best such as the Bag-of-Words-based proposed approach.

[2] Available on huggingface and its *Transformers* library.

Code	Language	Model	Spearman Correlation (dev/eval)	Rank Dev	Rank Eval
Arq	Algerian Arabic	One+	0.574 / 0.566	8	10
Amh	Amharic	One-B	0.676 / 0.702	19	15
Eng	English	Transformer	0.789 / 0.809	35	29
Hau	Hausa	One-B	0.547 / 0.576	20	15
Kin	Kinyarwanda	One-B	0.430 / 0.630	14	12
Mar	Marathi	One+	0.750 / 0.784	21	20
Ary	Moroccan Arabic	One+	0.820 / 0.811	12	9
Spa	Spanish	Transformer	0.701 / 0.678	7	13
Tel	Telugu	One-B	0.818 / 0.801	10	14

Table 1: Best model and results for each language dataset for the relatedness prediction problem.

4 Conclusion

This manuscript describes our participation in Task 1 of Semantic Textual Relatedness (STR) at SemEval 2024. We used two main approaches: a transformer-based approach and an EvoMSA-based one. The latter has lexical and semantic representations, with variants using pre-training and fully learned from the training data. Our transformer solution works better for Spanish and English, while our EvoMSA works better for the other languages. In particular, we support low-resource languages using our EvoMSA without pre-trained models. Our competitive results give evidence suggesting that languages with fewer resources can benefit from models that do not require an enormous corpus to be trained; this can be an alternative to large models. Nevertheless, this is a very complex task, and better efforts could be made in the future.

References

James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. 2018. JAX: composable transformations of Python+NumPy programs.

Dhivya Chandrasekaran and Vijay Mago. 2021. Evolution of semantic similarity—a survey. *ACM Comput. Surv.*, 54(2).

Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova Google, and A I Language. 2018a. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Technical report.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018b. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Mario Graff, Sabino Miranda-Jiménez, Eric S. Tellez, and Daniela Moctezuma. 2020. EvoMSA: A Multilingual Evolutionary Approach for Sentiment Analysis. *Computational Intelligence Magazine*, 15(1):76 – 88.

Mario Graff, Daniela Moctezuma, Eric Tellez, and Sabino Miranda. 2023. Ingeotec at DA-VINCIS: Bag-of-Words Classifiers. *CEUR Workshop Proceeding*, 3496:1–10.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. *Preprint*, arXiv:1801.06146.

Aminul Islam and Diana Inkpen. 2008. Semantic text similarity using corpus-based word similarity and string similarity. *ACM Trans. Knowl. Discov. Data*, 2(2).

Tom Kenter and Maarten de Rijke. 2015. Short text similarity with word embeddings. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, CIKM '15, page 1411–1420, New York, NY, USA. Association for Computing Machinery.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From word embeddings to document distances. In *International conference on machine learning*, pages 957–966. PMLR.

Nedjma Ousidhoum, Shamsuddeen Hassan Muhammad, Mohamed Abdalla, Idris Abdulmumin, Ibrahim Said Ahmad, Sanchit Ahuja, Alham Fikri Aji, Vladimir Araujo, Abinew Ali Ayele, Pavan Baswani, Meriem Beloucif, Chris Biemann, Sofia Bourhim, Christine De Kock, Genet Shanko Dekebo, Oumaima Hourrane, Gopichand Kanumolu, Lokesh Madasu, Samuel Rutunda, Manish Shrivastava, Thamar Solorio, Nirmal Surange, Hailegnaw Getaneh Tilaye, Krishnapriya Vishnubhotla, Genta Winata, Seid Muhie Yimam, and Saif M. Mohammad. 2024a. Semrel2024: A collection of semantic textual relatedness datasets for 14 languages. *Preprint*, arXiv:2402.08638.

Nedjma Ousidhoum, Mohamed Abdalla Shamsuddeen Hassan Muhammad, Idris Abdulmumin, Ibrahim Said Ahmad, Sanchit Ahuja, Alham Fikri Aji, Vladimir Araujo, Meriem Beloucif, Christine De Kock, Oumaima Hourrane, Manish Shrivastava, Thamar Solorio, Nirmal Surange, Krishnapriya Vishnubhotla, Seid Muhie Yimam, and Saif M. Mohammad. 2024b. SemEval-2024 task 1: Semantic textual relatedness. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Ryoma Sato, Makoto Yamada, and Hisashi Kashima. 2022. Re-evaluating word mover's distance. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 19231–19249. PMLR.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *arXiv preprint arXiv:2004.09297*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.