PAR-MEX Shared Task Submission Description: Identifying Spanish Paraphrases Using Pretrained Models and Translations

Leander Girrbach[1]

[1]*University of Tübingen, Germany*

Abstract

This paper describes our participation in the Paraphrase Identification in Mexican Spanish (PAR-MEX) shared task [1]. We show that publicly available, pretrained models already achieve very strong performance on this task (ranking 2nd among all submissions and achieving 0.9373 f1 score on test data) and can be used in a data efficient way. The performance is good despite domain specificity of the data and linguistic differences, e.g. Mexican Spanish vs. standard Spanish. Furthermore, we show that translation to English allows for using further pretrained models with stronger performance. We conclude that combining different pretrained and multilingual models is a currently strong and efficient approach to paraphrase identification, and paraphrase identification will likely benefit more from stronger pretrained models than from specialised methods.

Keywords

Paraphrase Identification, Pretrained models

1. Introduction

This paper describes our participation in the Paraphrase Identification in Mexican Spanish (PAR-MEX) shared task [1] located with IberLEF 2022. The task is a binary classification task: Given two sentences, predict whether they are paraphrases of each other (label 1) or not (label 0). Potential challenges of this dataset are that it represents a specific domain (cuisine) and a specific dialect of Spanish, namely Mexican Spanish.

Paraphrase identification has several practical and theoretical applications. Theoretical applications most prominently include paraphrase detection for evaluation purposes. For example, paraphrase identification models can help to evaluate machine translation or text simplification models, where being semantically equivalent is an important relation that predicted sentences and ground truth sentences have to share. Practical applications include document retrieval, most prominently duplicate detection. For example, paraphrase identification can be used to detect plagiarism, deduplicate questions in online fora, or discover duplicate patents.

In this paper, we show that using only pretrained models without fine-tuning already yields very strong performance on this paraphrase identification task. In addition, we show that making use of pretrained models for other languages than the target language (Spanish in this case) is very useful. This is not surprising, since it has been shown that language models

IberLEF 2022, September 2022, A Coruña, Spain.

pretrained on large amounts of data learn to represent a wide range of linguistic properties, including semantics on the word, sentence, and document level. Therefore, a natural starting point for identifying Spanish paraphrases is using pretrained models. In fact, the baseline model proposed by the task organisers is a fine-tuned Spanish BERT model.

However, pretraining new models is infeasible for the majority of practitioners, and even fine-tuning a deep model requires access to sufficiently powerful computing resources. Another limiting factor is that most data resources and pretrained models target the English language.

Therefore, our participation in the shared tasks aims for an approach that makes as much use of freely available pretrained models as possible. Indeed, all proposed methods work on most modern computers (without access to GPU). Furthermore, we explore how well approaches work that do not use the original Spanish data, but use automatic translations into English.

Our main idea is to encode Spanish sentences (or their English translations) by a pretrained sentence encoder, and predict the paraphrase label from either the concatenation of the sentence representations or a similarity score of the sentence representation, for example cosine similarity.

In summary, the contributions of this paper are:

1. Propose and evaluate an approach to paraphrase identification that requires little training and makes use of publicly available, pretrained models.
2. Evaluate the data efficiency of the proposed approach and show that good performance can already be achieved using less than 1000 sentence pairs as training data

The rest of the paper is organised as follows: In Sec. 2, we describe our approach in detail. In Sec. 3 we describe the performance of our proposed method in a cross-validation experiment on the training data provided by the task organisers, because test and development labels remain hidden. Finally, in Sec. 4 we discuss related work, and in Sec. 5, we discuss consequences of our findings and possible future work.

2. Method

For our approach, we use the following pretrained models:

- PARAPHRASE-MULTILINGUAL-MPNET-BASE-V2 multlingual sentence embedding model from [2] for embedding Spanish sentences
- ALL-MPNET-BASE-V2 English sentence embedding model from [3] for embedding English sentences
- CROSS-ENCODER/STSB-ROBERTA-LARGE cross-encoder model from [3] for calculating similarity scores of English sentence pairs
- OPUS-MT-ES-EN Spanish-to-English translation model from [4] for translating Spanish sentences into English

The sentence embedding models are available from sentence transformers (https://www.sbert.net/docs/pretrained_models.html), and so is the cross-sentence encoder (https://www.sbert.net/docs/pretrained_cross-encoders.html). The translation model is available from the transformer model hub (https://huggingface.co/Helsinki-NLP/opus-mt-es-en).

Our approach employs the classical machine learning pipeline consisting of feature extraction, supervised training of a classifier, and evaluation. We give details on how we use the pretrained models listed above for feature extraction in the next subsection.

2.1. Feature Extraction

Given pairs of sentences (s_1, s_2), we evaluate two types of features that are based on sentence embeddings of s_1 and s_2:

1. Concatenate the sentence embeddings $\mathbf{s}_1$ and $\mathbf{s}_2$. In our case, this yields 1536 dimensional feature vectors.
2. Represent (s_1, s_2) by a similarity score s. s could be the cosine similarity $\cos(\mathbf{s}_1, \mathbf{s}_2)$ of the sentence embeddings. This yields one scalar feature, and in consequence we only need to set a threshold to classify sentence pairs as paraphrases.

Using these kinds of features, we get the following 5 feature representations of Spanish sentence pairs:

- Concatenated sentence embeddings of original Spanish sentences.
- Concatenated sentence embeddings of translated English sentences.
- Cosine similarities of Spanish sentence embeddings.
- Cosine similarities of translated English sentence embeddings.
- The output scores of the CROSS-ENCODER/STSB-ROBERTA-LARGE model given translated English sentence pairs as input.

In the next section, we give details on our methods to classify Spanish sentence pairs into paraphrase or non-paraphrase using the features listed above.

2.2. Classification Methods

Since the two types of features, concatenated sentence embeddings and cosine similarities, are different in nature, we use different classifiers for each feature type. For similarity score features, we train logistic regression models. This is equivalent to learning an optimal threshold from the data. Furthermore, we train a logistic regression model and a random forest model on the combined 3 similarity score features. If we think of similarity scores as outputs of classifiers, this would be an instance of stacked generalisation ensembling [5]. It turns out that random forests yield slightly better performance, therefore our final submission to the shared task uses a random forest classifier. For inspection, however, logistic regression models are easier to analyse. Since predictions are the same in the vast majority of cases, we can analyse the logistic regression model's weights given to each of the 3 similarity scores to determine which feature is most useful for identifying paraphrases.

For concatenated sentence embedding features, we train multi-layer perceptrons (MLPs). This is necessary, because sentence embeddings do not represent directly interpretable information. Therefore, we need to use non-linear models to extract the relevant information.

We make use of the `scikit-learn` implementation [6] of logistic regression, random forest, and MLP. Models are trained with default hyperparameters. Exceptions are the batch size

for MLP, which we set to 32, and we use two hidden layers with 128 units each. Our final best-performing submission is a random forest model trained on the 3 similarity features.

2.3. Evaluation Methods

2.3.1. Cross-validation

Since the ground-truth test labels are not available, we evaluate our approaches by 10-fold cross-validation on the training dataset provided by the task organisers. We generate stratified splits of the data for cross-validation, i.e. the ratio of positive and negative labels is equal in every split. Finally, we report median f1 score, precision, recall, and accuracy from the 10 evaluations.

2.3.2. Data efficiency

Since we use pretrained models, we hypothesise that the performance of our models is largely independent of the actual data. Especially for the similarity score features, we assume that we only need a relatively small development set to tune the threshold. To evaluate our hypothesis, we split the labelled training data into training and development set. The training data contains 5905 sentence pairs and the development contains 1477 sentence pairs. Then, we plot the f1 score of our classifiers on the development set as a function of the training set size. This means, we train the classifiers on increasingly large subsets of the training set and evaluate the performance on the held-out validation set.

3. Results

3.1. Classification performance

In Tab. 1, we report f1, accuracy, precision, and recall scores from our cross-validation experiments which we describe in Sec. 2.3. For each metric and feature type, we report the median score from the 10 runs. Additionally, we report the used classifier (multi layer perceptron or logistic regression) and the language (English or Spanish), if applicable. On the test data (whose gold-labels remain hidden), our best model achieves 0.9373 f1 score according to the shared task's codalab page: https://codalab.lisn.upsaclay.fr/competitions/2345#results.

We can see that all features perform similarly in terms of f1 score and accuracy. The worst performing model is English cosine similarity. This means that a relevant amount of information is lost in the translation process. However, performance of the cross-encoder model, which also takes English translations as input, is the strongest individual model. This means that a stronger model, in this case fine-tuned for sentence similarity, can compensate the disadvantages of using translations. Generally, sentence embedding features yield slightly higher precision, but lower recall. Inversely, similarity score features yield lower precision, but higher recall. The similarity score ensemble model is able to improve both precision and recall over all individual similarity score features. In effect, the similarity score ensemble model is the strongest overall model.

	Model		f1	Accuracy	Precision	Recall
EN	Concat. Sent Embeddings	(MLP)	0.93	0.98	0.93	0.95
ES	Concat. Sent Embeddings	(MLP)	0.94	0.98	0.93	0.96
EN	Cos Similarity	(LogReg)	0.89	0.96	0.84	0.92
ES	Cos Similarity	(LogReg)	0.94	0.98	0.91	0.96
EN	Cross Encoder Score	(LogReg)	0.95	0.98	0.92	0.97
	Sim Ensemble	(LogReg)	0.95	0.98	0.93	0.97

Table 1
Results (median scores) from 10-fold cross-validation.

If we compare languages, we can see that both Spanish sentence embedding features perform better than English sentence embedding features and Spanish cosine similarity performs better than English cosine similarity. This suggests that working in the target language is superior to working with translations, in case strong enough models for the target language are available. Also, translation models currently do not perfectly preserve semantic relations, such as being paraphrases. However, English cross-encoder similarity score features outperform Spanish cosine similarity features. Therefore, if stronger models in an auxiliary language are available, but not for the target language, it may be worth considering working with translations. This is confirmed by looking at the learned weights of the ensemble logistic regression model: The median weight (from the 10 runs) of English cosine similarity is 2.73, the median weight of Spanish cosine similarity is 6.44, and the median weight of English cross-encoder scores is 8.97. Therefore, English cross-encoder scores are the most important feature for ensemble predictions, followed by Spanish cosine similarity.

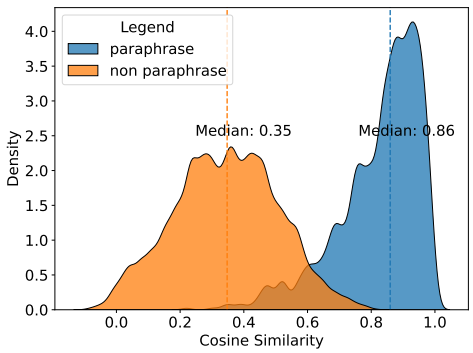
3.1.1. Similarity features density plots

In order to get a better impression of the usefulness of similarity scores induced by the pretrained sentence embedding models and the cross-encoder, in Fig. 1 we show density plots of the similarity score distributions for paraphrase sentence pairs and non-paraphrase sentence pairs.

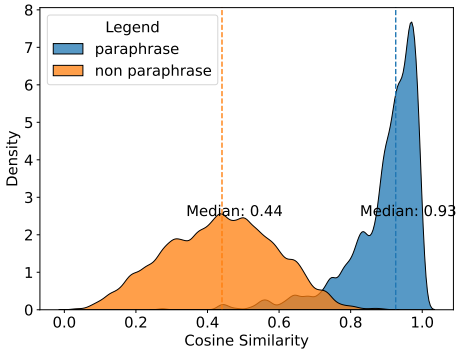
In all cases, the densities have only small overlap. Most similarity scores for paraphrase sentence pairs are close to 1. Most similarity scores for non-paraphrase sentence pairs are at least lower than 0.5. The only relevant overlap is roughly in the region between scores 0.4 to 0.6. This means that only very few paraphrases are considered strongly dissimilar by the models, and very few non-paraphrases are considered very similar by the model. Also, the densities of non-paraphrase scores have higher variance. This is probably due to the fact that sentences even when not being paraphrases can be more or less similar in many ways, while being paraphrases is a more rigid constraint on the sentences.

3.1.2. Data efficiency

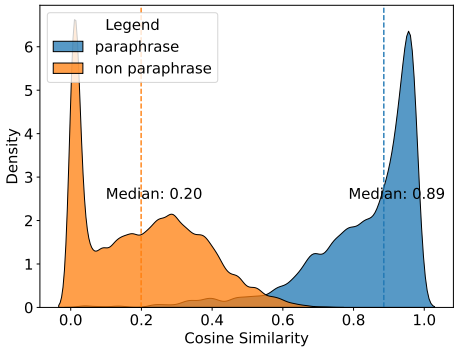
As explained in Sec. 2.3, we evaluate the data efficiency of our methods by plotting the f1 score on a held-out validation set (1477 sentence pairs) as a function of the training set size. We evaluate training set sizes in steps of 100 sentence pairs. The resulting curves are in Fig. 2.



(a) English cosine similarity distribution



(b) Spanish cosine similarity distribution



(c) English cross-encoder similarity score distribution

Figure 1: Similarity score distributions for paraphrase sentence pairs and non-paraphrase sentence pairs.

As hypothesised, the performance of similarity features begins to plateau early, already after the training set contains ≈ 500 sentence pairs. This shows that using similarity features is both strong (because of the pretrained models) and data efficient, because we only have to tune a single threshold. MLP based models seem to always benefit from more data. However, they only reach competitive performance after the training set contains ≈ 4000 sentence pairs.

4. Related Work

Paraphrase identification is a well-established task in NLP. Therefore, many approaches to paraphrase identification have been proposed. Earlier neural approaches include convolutional-neural-network-based [7] and BiLSTM-based [8] siamese architectures [9]. These approaches have been largely superseded by the pretrain-finetune approach of transformer-based language models [10].

The baseline of this task proposed by the task organisers also is a fine-tuned Spanish BERT

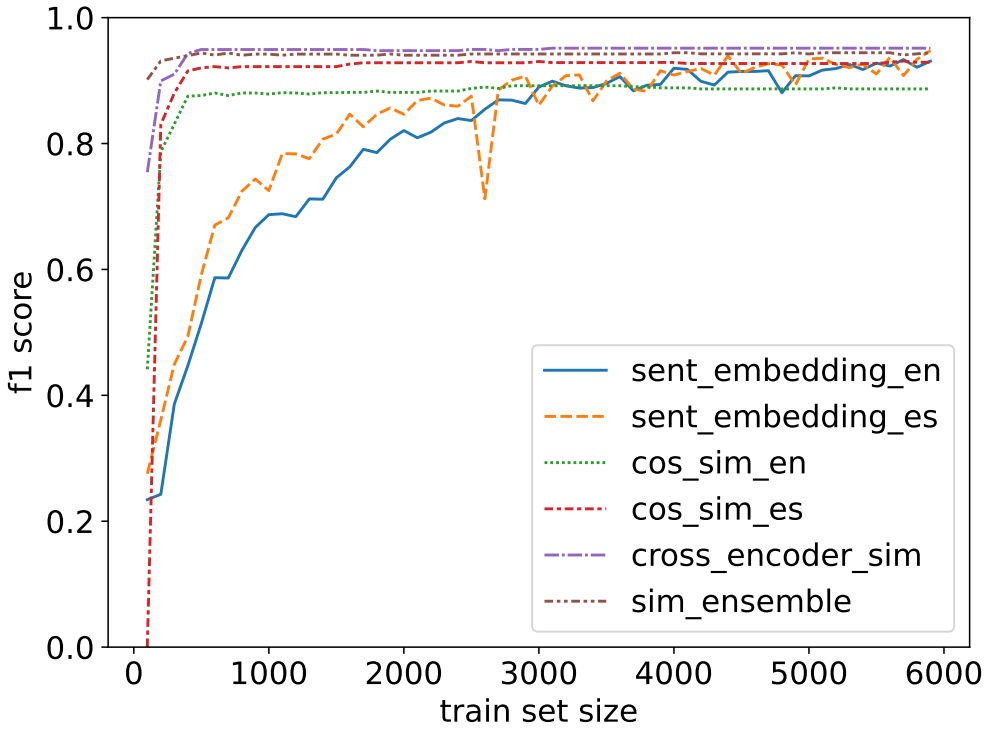


Figure 2: f1 score on held-out validation set as a function of the training set size for different features.

model. In this paper, however, we only use pretrained models without fine-tuning. However, the used sentence embedding models and the cross-encoder model have already been fine-tuned on similar tasks before release. Otherwise, we use the pretrained models as feature extractors.

There are many paraphrase identification or related datasets available, such as [11]. However, most of them only include English data. One exception is the data provided for SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Cross-lingual Focused Evaluation [12] which includes data for Arabic, Spanish, English, and Turkish.

Multilingual approaches, such as we evaluate in this paper, are common as a method for dealing with low-resource languages or constructing baselines. For example, [13] propose backtranslation as a means of data augmentation. [14] propose to parse sentences into AMR by first translating them into English and then applying an English AMR parser. This is the same idea that we evaluate in this paper. But, differently than [14], we find models optimised for Spanish to be competitive with models working on English translations. A fair comparison is difficult, however, because we use a multilingual model, that was trained on Spanish data and more languages, for obtaining Spanish sentence embeddings. For a fairer comparison, using a monolingual Spanish model would be necessary.

Finally, different evaluation schemes for paraphrase identification have been proposed. For

example, [15] annotate an already existing paraphrase identification dataset with linguistic information to allow a more fine-grained evaluation. [16] point out weaknesses of using pretrained language models for paraphrase identification. They also question whether stating paraphrase identification as a binary classification task is useful for practical purposes. According to [16], stating paraphrase identification as a binary classification task does not match retrieval tasks as the primary practical application.

These more general considerations do not affect this paper, as our main purpose is finding and analysing successful methods that work for the given data. However, future research, both on evaluation of paraphrase identification and methods for identifying paraphrases, may benefit from these ideas.

5. Discussion

In this paper we show that publicly available, pretrained models already achieve strong performance in Spanish paraphrase identification. All computations can be performed on most modern computers within reasonable time. Furthermore, we show that combining different models including translations is useful.

From this, we draw the following conclusions:

- Despite of domain and language specificity of the dataset, pretrained models perform well even without fine-tuning on the dataset. This suggests that the used pretrained models learn to represent general semantic information.
- Using stronger pretrained models is likely more successful than devising dataset or task specific methods for paraphrase identification. This confirms observations for metaphor identification in [17].
- If suitable models are not available for the target language, translating into another language (especially English), where suitable models are available, is a promising option. Given the effort invested in automatic machine translation, translation models are likely to be available, in case any pretrained models for the target language exist. However, this may be restricted to typologically similar languages.
- Ensembling different approaches, also across languages, can further improve performance.

Therefore, in some cases, merely assembling available tools to build new compound solutions to NLP tasks can be a valid approach. This perspective may especially be useful in practical or transdisciplinary contexts, because, for example, some digital humanities practitioners may not be able to fine-tune models, but they can likely use pretrained models when provided with sufficient documentation.

Acknowledgments

We thank the organisers of the shared task for their support and fast responses during evaluation and write-up phase. We also thank the reviewers for their comments.

References

[1] G. Bel-Enguix, H. Gomez-Adorno, G. Sierra, J.-M. Torres-Moreno, J.-G. Ortiz-Barajas, J. Vasquez, Overview of PAR-MEX at Iberlef 2022: Paraphrase Detection in Spanish Shared Task, Procesamiento del Lenguaje Natural 69 (2022).

[2] N. Reimers, I. Gurevych, Making monolingual sentence embeddings multilingual using knowledge distillation, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online, 2020, pp. 4512–4525. URL: https://aclanthology.org/2020.emnlp-main.365. doi:10.18653/v1/2020.emnlp-main.365.

[3] N. Reimers, I. Gurevych, Sentence-BERT: Sentence embeddings using Siamese BERT-networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3982–3992. URL: https://aclanthology.org/D19-1410. doi:10.18653/v1/D19-1410.

[4] J. Tiedemann, The tatoeba translation challenge – realistic data sets for low resource and multilingual MT, in: Proceedings of the Fifth Conference on Machine Translation, Association for Computational Linguistics, Online, 2020, pp. 1174–1182. URL: https://aclanthology.org/2020.wmt-1.139.

[5] D. H. Wolpert, Stacked generalization, Neural networks 5 (1992) 241–259.

[6] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[7] W. Yin, H. Schütze, Convolutional neural network for paraphrase identification, in: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Denver, Colorado, 2015, pp. 901–911. URL: https://aclanthology.org/N15-1091. doi:10.3115/v1/N15-1091.

[8] A. Mahmoud, M. Zrigui, Blstm-api: Bi-lstm recurrent neural network-based approach for arabic paraphrase identification, Arabian Journal for Science and Engineering 46 (2021) 4163–4174.

[9] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, R. Shah, Signature verification using a siamese time delay neural network, in: J. D. Cowan, G. Tesauro, J. Alspector (Eds.), Advances in Neural Information Processing Systems 6, [7th NIPS Conference, Denver, Colorado, USA, 1993], Morgan Kaufmann, 1993, pp. 737–744. URL: http://papers.nips.cc/paper/769-signature-verification-using-a-siamese-time-delay-neural-network.

[10] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, in: J. Burstein, C. Doran, T. Solorio (Eds.), Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), Association for Computational Linguistics, 2019, pp. 4171–4186. URL: https://doi.org/10.18653/v1/n19-1423.

doi:10.18653/v1/n19-1423.

[11] W. B. Dolan, C. Brockett, Automatically constructing a corpus of sentential paraphrases, in: Proceedings of the Third International Workshop on Paraphrasing (IWP2005), 2005. URL: https://aclanthology.org/I05-5002.

[12] D. Cer, M. Diab, E. Agirre, I. Lopez-Gazpio, L. Specia, SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation, in: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 1–14. URL: https://aclanthology.org/S17-2001. doi:10.18653/v1/S17-2001.

[13] J.-P. Corbeil, H. A. Ghadivel, Bet: A backtranslation approach for easy data augmentation in transformer-based paraphrase identification context, arXiv preprint arXiv:2009.12452 (2020).

[14] S. Uhrig, Y. Garcia, J. Opitz, A. Frank, Translate, then parse! a strong baseline for cross-lingual AMR parsing, in: Proceedings of the 17th International Conference on Parsing Technologies and the IWPT 2021 Shared Task on Parsing into Enhanced Universal Dependencies (IWPT 2021), Association for Computational Linguistics, Online, 2021, pp. 58–64. URL: https://aclanthology.org/2021.iwpt-1.6. doi:10.18653/v1/2021.iwpt-1.6.

[15] V. Kovatchev, M. A. Marti, M. Salamo, J. Beltran, A qualitative evaluation framework for paraphrase identification, in: Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019), INCOMA Ltd., Varna, Bulgaria, 2019, pp. 568–577. URL: https://aclanthology.org/R19-1067. doi:10.26615/978-954-452-056-4_067.

[16] H. Chen, Y. Ji, D. Evans, Pointwise paraphrase appraisal is potentially problematic, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop, Association for Computational Linguistics, Online, 2020, pp. 150–155. URL: https://aclanthology.org/2020.acl-srw.20. doi:10.18653/v1/2020.acl-srw.20.

[17] A. Neidlein, P. Wiesenbach, K. Markert, An analysis of language models for metaphor recognition, in: Proceedings of the 28th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 3722–3736. URL: https://aclanthology.org/2020.coling-main.332. doi:10.18653/v1/2020.coling-main.332.