Vicomtech at eHealth-KD Challenge 2021: Deep Learning Approaches to Model Health-related Text in Spanish

Aitor García-Pablos[1 [0000−0001−9882−7521]], Naiara Perez[1 [0000−0001−8648−0428]], and Montse Cuadros[1 [0000−0002−3620−1053]]

SNLT group at Vicomtech Foundation, Basque Research and Technology Alliance (BRTA), Mikeletegi Pasealekua 57, Donostia/San-Sebastián, 20009, Spain
{agarciap,nperez,mcuadros}@vicomtech.org
https://www.vicomtech.org

Abstract. This paper describes the participation of the Vicomtech NLP team in the eHealth-KD 2021 shared task about detecting and classifying entities and relations in health-related texts written in Spanish. We participate with two systems. Joint Classifier is a simplified version of the system that won the main scenario of the previous eHealth-KD edition. It consists of a single end-to-end deep neural network with pre-trained BERT models as the core for the semantic representation of the input texts, that predicts all the output variables—entities and relations—at the same time, modelling the whole problem jointly. The main change w.r.t. the original implementation affects the representation of relations. The Joint Classifier model achieved the first position in the main scenario of the competition and ranked second in the rest of the scenarios. The second submitted system, Seq2Seq, uses an approach based on an encoder-decoder model. It transduces the input text into an output sequence by reading the input text. The target sequence is a compact representation of the information contained in the gold-labels of the datasets. This approach showed a promising performance despite not being competitive enough. However, it poses an interesting potential future work.

Keywords: Entity detection · Relation extraction · Health documents

1 Introduction

This article describes Vicomtech's participation at the *eHealth Knowledge Discovery challenge* (*eHealth-KD*) 2021 (https://ehealthkd.github.io/2021) [12]. The

IberLEF 2021, September 2021, Málaga, Spain.

challenge proposes a general-purpose semantic structure to model human language, consisting of 4 types of entities and 13 types of relations (see example in Figure 1). We refer the reader to the challenge overview article [12] for detailed information about eHealth-KD 2021, such as descriptions of the provided corpus and evaluation scenarios.

Vicomtech's participation builds partly on the system submitted to the previous edition [4], which addressed the problems of entity and relation extraction jointly with several classification heads on top of a fine-tuned BERT encoder [3] and intermediate token-pair representations. This approach ranks first in the main challenge scenario for the consecutive year.

As a novelty, Vicomtech has also experimented with a text-to-text approach, which has recently gained interest as a promising alternative to successfully solving very different NLP tasks in a unified manner [22, 16, 7, 5]. We are interested in this approach because it does not suffer from the limitations of traditional sequence labelling techniques, in particular to represent the non-contiguous and overlapping entities. Although this model ranks lower in the main challenge scenario, the results indicate that the text-to-text approach is a viable option that is worth exploring for this task as well.

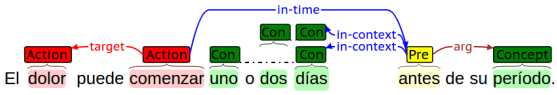

Fig. 1: Example of eHealth-KD annotations in the sentence "The pain may start a day or two before your period."

The paper is organised as follows: Section 2 describes the two proposed models, Joint Classifier and Seq2Seq, and their training setups; Section 3 presents the results obtained, including a comparison to other competing systems; finally, Sections 4 and 5 comment on several design choices and provide some concluding remarks.

2 System descriptions

This section provides a comprehensive description of the two submitted systems. For each, we first describe its architecture and then describe how the inputs and outputs have represented and handled. Finally, we present the training setup.

2.1 Joint Classifier

This model is the result of the revision of our participation in the previous eHealth-KD edition [4]. Most of the changes introduced in the current edition aim at simplifying components of the architecture that seemed redundant or

needlessly complex. In addition, we have forgone the ability to predict multi-word discontinuous and/or overlapping entities like the shown in Figure 1, which in any case required elaborate post-processing to yield acceptable results.

Architecture Joint Classifier is a deep neural network that receives the input tokens and jointly emits predictions for two output variables:

- **Entities:** the classification of each individual token into one of the task's entity types or 'O' (from 'Out', meaning that the token is not part of any entity at all, such as "puede" in Figure 1). We use the classical BIO tagging scheme [14] to encode entity boundaries, so the effective output vocabulary size for the 4 entity types is 9.
- **Relations:** whether token pairs are related by any of the relation types described in the task, or 'O' when there is no relation between the tokens, for a total of 14 output labels.

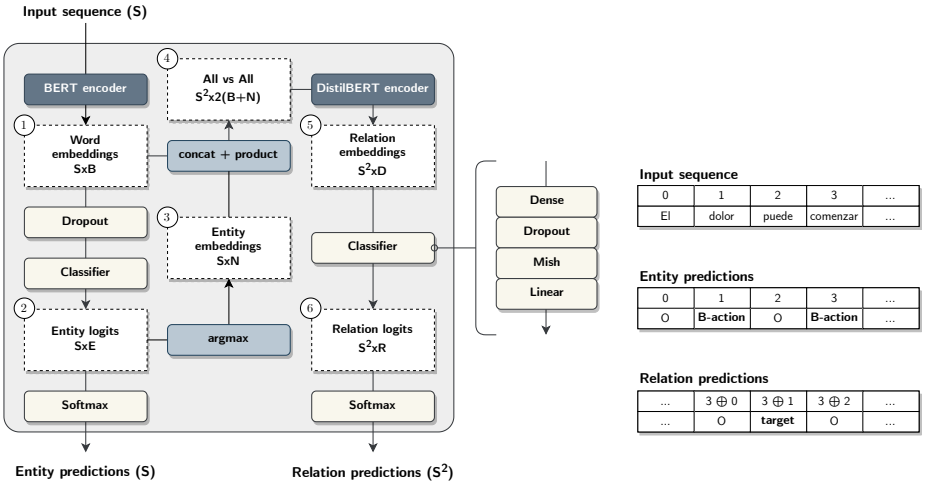

Fig. 2: High-level diagram of the proposed Joint Classifier model, including the shapes of each layer: B=768 (BERT contextual embedding size); N=64 (entity embedding size); E=9 (output vocabulary size for the entities head), D=768 (DistilBERT contextual embedding size), R=14 (output vocabulary size for the relations head).

An overview of the inner workings of the network is given in Figure 2. The computation of the model starts by feeding the input tokens into a BERT model to obtain their contextual embeddings (1). These embeddings are passed to a classification layer that emits logits with the predictions about each token being

or not an entity of a certain type (2). The output of this classification layer is one of the two outputs of the network.

The entity logits pass through an argmax function to select the entity index that, in its current state, the model would predict for each token. These entity indexes are used to select entity embeddings from a custom embeddings layer initialised at the beginning of the training (3).

Each entity embedding is concatenated to the token contextual embedding it was predicted from. The resulting vectors are operated to obtain an all-vs-all combination of all tokens, resulting in $S \times S$ combined embeddings that represent all the possible token pairs, S being the length of the input sequence (4). These embeddings are further passed to a small randomly initialised DistilBERT model [15] with only two layers of two attention heads each. The objective of this model is to further capture interactions between the token pairs via self-attention.

Finally, the resulting relation representations (5) are passed to a another classification layer to predict the type of the relation between each pair of tokens (if there is a relation at all) (6). Relations are modelled as outgoing arcs. That is, if the pair $token_i \oplus token_j$ is linked by a relation of type R, $token_i$ and $token_j$ are the source and destination of the relation R respectively. This is the second and last output of the network.

The network has a total of two classifiers built with the same stack of layers: a fully connected linear transformation layer, followed by a dropout layer and a non-linear activation function—Mish [9]—, and a final linear transformation that outputs the logits for the given output variable. The output probabilities are obtained by applying the softmax function.

Input and output handling The challenge corpus has been provided in Brat [17] standoff format (see Figures 1 and 3a). This format is character span-based, while Joint Classifier works at token level. Consequently, this system relies on a set of pre-processing and post-processing transformation steps, explained below.

Input representation Figure 3b shows an example of the information representation designed with all the network's input and output variables.

Notably, the proposed system does not address continuous and/or overlapping entities, as it relies on the BIO tagging scheme. As shown in Figure 3b, the text span "uno o dos días" is represented as "uno" on the one hand and "dos días" on the other, while the gold annotations define the entities "uno días" and "dos días" for the same text span (see Figure 3a).

As explained in Section 2.1, Joint Classifier performs all the tasks end-to-end, using its own entity predictions as input to detect relations. In *Task B*, gold entity annotations are provided by the task organisers; systems need to focus on the relations only. In this case, our model accepts gold entity labels along the input tokens, and replaces the predicted entities with a one-hot encoding of the gold ones as the input for detecting relations.

```
T705 Action 10111 10116 dolor
T706 Action 10123 10131 comenzar
T707 Concept 10132 10135;10142 10146 uno días
T708 Concept 10138 10141;10142 10146 dos días
T709 Predicate 10147 10152 antes
T710 Concept 10159 10166 período
R628 in-time Arg1:T706 Arg2:T709
R629 target Arg1:T706 Arg2:T705
R630 arg Arg1:T709 Arg2:T710
R631 in-context Arg1:T709 Arg2:T708
R632 in-context Arg1:T709 Arg2:T707
```

(a) Representation in Brat's standoff format, i.e. the original format of the challenge corpus and the format expected for submission.

```
El          O             O                       -
dolor       B-action      O                       -
puede       O             O                       -
comenzar    B-action      target,in-time          1,8
uno         B-Concept     O                       -
o           O             O                       -
dos         B-Concept     O                       -
días        I-Concept     O                       -
antes       B-Predicate   in-context,in-context   4,6
```

(b) Representation for Joint Classifier. The first column provides the input variables: the tokens. The second and third columns provide the output variables, namely, the entity tags and relation types. The last column contains pointers to the destination tokens of the relations (note it is not an output variable; this information is encoded in the $S \times S$ matrices described in Section 2.1).

```
comenzar:[Action];[in-time];antes:[Predicate]
comenzar:[Action];[target];dolor:[Action]
antes:[Predicate];[arg];período:[Concept]
antes:[Predicate];[in-context];dos días:[Concept]
antes:[Predicate];[in-context];uno días:[Concept]
```

(c) Representation for Seq2Seq (note that the actual representation consists of a single line, while here we show one pentad per line for better readability).

Fig. 3: Textual representations of the annotations depicted in Figure 1.

Output interpretation The output of the entity classifier is straightforwardly interpreted as a regular sequence-labelling task, selecting the most probable prediction for each individual token. Next, we construct the span-based annotations from the emitted BIO tags. Ill-formed tag sequences are always solved by transforming the offending tag into the begging of a new entity (e.g. the tag sequence `[B-Concept, I-Concept, I-Action]` would be fixed as `[B-Concept, I-Concept, B-Action]`).

As for relations, the network's outcome for each modelled relation variable forms a $S \times S$ matrix, S being the length of the token sequence. Each position i and j; $i, j \in [0, S]$ contains the prediction for the relation between $token_i$—the source—and $token_j$—the destination. Again, as the expected output is span-based instead of token-based, the predicted relations are mapped from source/destination tokens to source/destination entities, having the entities been interpreted just as explained above. Relations from/to tokens that are not part of an entity are simply ignored, as are repeated and reflexive relations.

Throughout the whole process, token positions must be correctly handled to account for deviations and extra offsets introduced by BERT's tokenization—BERT uses WordPiece tokenization [20], which breaks original tokens into subtokens; in addition, it requires that extra special tokens be added which distort the token positions w.r.t. the original input.

2.2 Seq2Seq

The are several phenomena in the challenge data, such as the non-contiguous multi-word entities mentioned earlier, that are difficult to model and predict with a traditional sequence labelling model. For this reason, we have explored a radically different strategy: the text-to-text paradigm. This paradigm is way more flexible because it can ingest a sequence of any length, and output another arbitrary sequence. That is, the output sequence is not tied to the structure of the input; a text-to-text model can potentially encode any sort of information [13].

Architecture The objective of the SEQ2SEQ model is to transform an input sequence (i.e. a text) into another sequence of semi-structured elements that represent the relevant information of the task. To implement this approach, we rely on a Transformers-based encoder-decoder model [18]. In particular, we use the T5 architecture [13] as is. The approach revolves around the representation format of the information encoded by the sequences, and how to reinterpret this information back to the original, task-specific, representation format.

Input and output handling How the output is encoded is critical to help the model learn deriving the desired information from the input. For this reason, we have attempted to produce a data format as compact and summarised as possible. Next, we describe this format and how the model output is mapped back to Brat's standoff format.

Target sequence representation As shown in Figure 3a, Brat's standoff format assigns a unique identifier to each entity. These identifiers are used to declare the relations between the entities. Further, the entities are defined not only by their type and actual text, but also by the spans in which they occur (that is, the positions in the text in terms of character offsets).

In order to relieve the encoder-decoder model from the burden of dealing with deictic information, the data has been represented as five-fold elements built around each relation: $r = (e1_{text}, e1_{type}, r_{type}, e2_{text}, e2_{type})$, where $e1$ and $e2$ are the source and destination entities of the relation r. Further, we serialise each of these elements with predefined punctuation marks as separators between the elements of the pentad. An example is shown in Figure 3c.

The proposed format gets rid of the boilerplate used by Brat's standoff format (e.g. "Arg1:", "Arg2:", and so on), reducing the sparsity and redundancy while keeping the output sequences as short as possible. Further, the entity types, the relation types and the separators are added to the model's tokenizer vocabulary, so they receive their own word-embedding.

Output interpretation Simplifying the target sequence representation as explained above means that all the omitted information, such as the locations of the entities in the input text, needs to be automatically extrapolated from the system's output. This task is made more difficult by the fact that the same term or expression may occur more than once in the input text; thus, the system must choose which of the occurrences the output refers to. On top of that, the model may produce elements that do not strictly occur in the input text, because we do not impose any constraint on the generation process (this topic is further discussed in Section 4).

The procedure of constructing Brat annotations from the model's output is as follows. First, we parse the output sequence into an array of pentads, guided by the predefined separators. Ill-formed pentads are directly ignored. As explained above, each parsed pentad represents a relation between two entities. Then, these entities must be located in the input text based on their textual form (i.e. $e1_{text}$ and $e2_{text}$).

The initial matching attempt is based on regular expressions. The pattern, built from the entity's text, allows for content of any length between the tokens of the entity. For instance, the pattern for "terapia biológica" would be `(\bterapia\b)(?:.*?)(\bbiológica\b)`. Case sensitive matches are preferred over insensitive ones, but the latter are allowed as well.

Only if the regular expressions do not yield any match whatsoever do we proceed to apply a more flexible search: we retrieve as matches all the words of the input text that have a Levenshtein distance [6] to the entity smaller than 3 edit operations. This is restricted to entities composed of only one word that is longer or equal to 5 characters.

If more than one match is obtained for a given entity, we choose the occurrence that is closest in the input text to the other related entity of the pentad. Finally, we merge entities that occur exactly in the same spans, and discard reflexive and repeated relations.

2.3 Training setup

Joint Classifier and Seq2Seq have been implemented in Python 3.7 with HuggingFace's transformers library (https://github.com/huggingface/transformers) [19] and trained each on 1 Nvidia GeForce RTX 2080 GPU with ∼11GB of memory.

For Joint Classifier, we have experimented with two pre-trained BERT models as the core for the semantic representation of the input tokens: IXAmBERT base cased [10] and BETO base cased [2]. The former has been pre-trained on Basque, English and Spanish Wikipedia content, while the latter is a monolingual model for Spanish.

The Seq2Seq system is built on a small mT5 model [21], a multilingual version of the T5 encoder-decoder [13]. The choice is merely based on the fact that mT5 is multilingual and that there are checkpoints available for several architectural sizes. Due to computational limitations, we use the mt5-small model.

Other training hyperparameters of both systems can be consulted in Table 1. It should be noted that no in-domain language post-training of the base models has been performed. In this sense, the approaches are general and domain agnostic. The only resource used for fine-tuning the whole systems is the training data provided for the task. We used the development data solely for the purpose of choosing the best models for submission.

Table 1: Training hyperparameters

	Joint Classifier	Seq2Seq
Max. input length	80	50
Max. target length	n/a	250
Batch size	2	2
Optimiser	AdamW [8]	AdamW [8]
Learning rate	2E-5	2E-5
Learning rate warm-up	linear, 50 epochs	linear, 5 epochs
Gradient acc. steps	2	4
Classifier dropout rate	0.5	n/a
Other dropout rates	0.1	0.1
Early stopping patience	500 epochs	50 epochs
Monitored metric	relations F1-score	tokens F1-score

3 Results

Vicomtech participated in the challenge with the following runs:

- Run 1: Seq2Seq with mT5 small
- Run 2: Joint Classifier with BETO base cased

– Run 3: Joint Classifier with IXAmBERT base cased

We did not submit Run 1 to Scenario 3—relation extraction—because presently the Seq2Seq system does not have a mechanism to exploit gold entity annotations.

The results for each scenario and run are shown in Table 2. We provide overall results as well as separate results for the languages present in the testing data, namely, Spanish and English. It must be noted that the training data consisted of content in Spanish exclusively. In addition, the best results obtained among all the participants in the challenge are also included per scenario for benchmarking purposes. In addition, we trained and evaluated Joint Classifier with BETO in the previous edition's data, in order to measure the impact of the changes introduced in its architecture. The results are shown in Table 3.

Table 2: Official results of the submitted runs and the best system in each scenario

	ES			EN			Total		
	P	R	F1	P	R	F1	P	R	F1
Scenario 1 - Main									
Run 1: mT5	67.01	51.64	58.33	33.69	32.12	32.88	51.27	43.44	47.03
Run 2: BETO	66.36	60.89	63.51	**37.12**	34.09	35.54	54.07	49.63	51.76
Run 3: IXAmBERT *(best)*	**68.55**	**62.68**	**65.49**	35.41	**40.73**	**37.88**	52.75	**53.46**	**53.11**
PUCRJ-PUCPR-UFMG [11] *(2nd)*	-	-	-	-	-	-	**56.85**	50.28	52.84
Scenario 2 - Task A: entity recognition and classification									
Run 1: mT5	**83.46**	71.06	76.77	**56.33**	44.11	49.48	69.99	57.11	62.90
Run 2: BETO	79.44	81.37	80.39	41.61	53.82	46.94	57.67	67.11	62.04
Run 3: IXAmBERT *(2nd)*	79.60	**83.48**	**81.49**	50.79	**66.53**	**57.60**	63.10	**74.71**	68.41
PUCRJ-PUCPR-UFMG [11] *(best)*	-	-	-	-	-	-	**71.49**	69.73	**70.60**
Scenario 3 - Task B: relation extraction									
Run 2: BETO	56.11	37.31	44.82	15.38	1.40	2.56	50.83	18.59	27.22
Run 3: IXAmBERT *(2nd)*	**57.88**	**40.10**	**47.38**	**47.77**	**17.48**	**25.60**	**54.19**	28.31	37.19
IXA [1] *(best)*	-	-	-	-	-	-	45.36	**40.95**	**43.04**

Joint Classifier with IXAmBERT has achieved the best scores of the challenge in the *Main* scenario (53.11 F1-score) by a narrow margin. It is surpassed by other participants in the individual tasks (i.e. scenarios 2 and 3), notably more so in the relation extraction task, where the best system achieves ∼6 F1-score points more (37.19 vs 43.04) due to its remarkably higher recall. Still, Joint Classifier with IXAmBERT is the second-best system in scenarios 2 and 3.

Joint Classifier with BETO performs consistently ∼2 points worse than IXAmBERT on the Spanish evaluation dataset. As is expected, this gap widens considerably on the English dataset, because IXAmBERT is a multilingual model,

unlike BETO. SEQ2SEQ, trained also on a multilingual model (i.e. mT5-small), actually performs better than JOINT CLASSIFIER with BETO on the NERC scenario; however, it obtains the worst results overall among Vicomtech's submitted runs. It must be noted, however, it shows competitive results when compared to other systems presented in the task [12].

Regarding the comparison between the original [4] and the submitted JOINT CLASSIFIER implementations, the results in Table 3 indicate that the simplifications introduced do affect the performance negatively, but the system remains competitive. The current version would still have won first place in the eHealth-KD 2020 edition despite achieving lower scores than the original version.

Table 3: Results on the eHealth-KD 2020 edition testing dataset (Scenario 1)

	P	R	F1
2020 JOINT CLASSIFIER with BETO	67.94	**65.23**	**66.56**
2021 JOINT CLASSIFIER with BETO	**68.67**	61.48	64.87

4 Discussion

The JOINT CLASSIFIER model is an attempt to reduce the complexity of the model by removing several seemingly redundant layers from the winner system of the last eHealth-KD edition. The reworked model gives up the two-way representation of the relations: instead of representing the incoming and outgoing arcs for every relation, the new model represents only the outgoing ones. Ideally, this would not cause any information loss, since the relations are in fact unidirectional. In addition, the reworked JOINT CLASSIFIER model does not model `same-as` and `multi-word` relations separately, dropping another set of layers. As a consequence, however, the model is unable to deal with non-contiguous or overlapping entities.

In spite of these simplifications, the resulting model has won this year's competition as well. According to our experiments, it would have obtained slightly lower scores in the 2020 data, suffering a loss of almost 4 recall points. This suggests that the redundancy in the modelling of relations did actually help the system detect more relations.

With reference to the SEQ2SEQ model, our experiments show promising results. A manual error analysis has revealed that many mismatches of entities are due to the model's output containing semantic, grammatical and/or orthographic variations of the input text (see examples in Table 4). Furthermore, we have observed that the model has difficulties with numeric expressions. It may also produce incorrect words, that is, words that do not exist in Spanish nor English. In a few occasions it even produces expressions that have no apparent relation to the input target, as shown by the last three examples. These

issues could be tackled by partly constraining the output of the model to elements present in the input text. Moreover, these models usually need more data than the available in this challenge—1,500 training sentences. A larger encoder-decoder model would also be necessary to obtain results that compete with the traditional approaches, given that the task requires a deep language understanding.

Table 4: Examples of predictions of entities generated by SEQ2SEQ ("*" means that the word is not part of the Spanish nor English vocabulary); the original terms to which the entity may be referring to are marked in boldface

Prediction	Sentence
quicker	[...] their detection can be much **faster** and simpler than RT-PCR.
principal	[It] was the **major** blood immune response for COVID-19 infection.
identificad	[...] mNGS **identified** six patients harboring transcriptionally active [...]
pandemia	[...] la **epidemia** de COVID-19 podría prolongarse por doce semanas.
Estas	**Todas** las personas tienen estreñimiento alguna vez.
Usted	[**Usted**] también puede comunicarle sus deseos a su familia.
25. 6 casos	[...] de coronavirus per cápita con **256.2 casos** por millón de personas.
60 años	Un hombre de **70 años** en el cantón de habla italiana de Ticino [...]
*probearse	La vacuna se está fabricando para que pueda **probarse** primero en animales.
*toriste	Es más que sentirse "**triste**" por algunos días.
guardan	También admiran lo externo, como a sus amigos, quienes suelen ser del mismo sexo.
compra	El cirujano cose el pulmón nuevo a los vasos sanguíneos y las vías respiratorias.
personas	Los nervios periféricos se encuentran fuera del cerebro y de la médula espinal.

5 Conclusions

In these working notes we have described Vicomtech's participation in the eHealth-KD 2021 shared task. We have presented a reworked version of the end-to-end deep-learning-based architecture that won last year's main scenario. This version of the model drops some expendable layers that were encoding somewhat redundant information. It still shows a very competitive performance, having won the main scenario again.

We have also presented an approach based on a sequence-to-sequence model. We have encoded the target information into a sequence of elements, so the model learns to derive those from the input text. This data representation is more flexible, and allows us to represent non-contiguous multi-word entities more easily. The results do not improve the more traditional approach, but the obtained

scores and the error analysis suggest that this approach may become competitive after addressing some specific issues, that we leave as future work. On the one hand, a more controlled and constrained output generation may improve the results. A relevant source of errors was related to the generation of entities that are not present in the input text. On the other hand, the use of more data or additional pre-training, combined with larger or better encoder-decoder models may also help.

All in all, the task remains challenging regardless of the model and approach. Further research will be necessary to improve the state of the art for key aspects of the task, in particular relation extraction.

Acknowledgments

This work has been partially funded by the projects DeepText (KK-2020-00088, SPRI, Basque Government) and DeepReading (RTI2018-096846-B-C21, MCIU/AEI/FEDER, UE).

References

1. Andrés, E.: IXA at eHealth-KD Challenge 2021: Generic Sequence Labeling as Relation Extraction Approach. In: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2021) (2021)
2. Cañete, J., Chaperon, G., Fuentes, R., Pérez, J.: Spanish Pre-Trained BERT Model and Evaluation Data. In: Proceedings of the Practical ML for Developing Countries Workshop at the Eighth International Conference on Learning Representations (ICLR 2020). pp. 1–9 (2020)
3. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). pp. 4171–4186 (2019)
4. García-Pablos, A., Perez, N., Cuadros, M., Zotova, E.: Vicomtech at eHealth-KD Challenge 2020: Deep End-to-End Model for Entity and Relation Extraction in Medical Text. In: Proceedings of the Iberian Languages Evaluation Forum co-located with 36th Conference of the Spanish Society for Natural Language Processing, IberLEF@ SEPLN. pp. 102–111 (2020)
5. Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., Steinhardt, J.: Measuring massive multitask language understanding. In: Proceedings of the Ninth International Conference on Learning Representations (ICLR 2021). pp. 1–27 (2021)
6. Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions, and reversals. Soviet Physics-Doklady **10**(8), 707–710 (1966)
7. Lopez, L.E., Cruz, D.K., Cruz, J.C.B., Cheng, C.: Transformer-based end-to-end question generation. arXiv preprint arXiv:2005.01107 pp. 1–9 (2020)
8. Loshchilov, I., Hutter, F.: Decoupled Weight Decay Regularization. In: Proceedings of the Seventh International Conference on Learning Representations (ICLR 2019). pp. 1–18 (2019)

9. Misra, D.: Mish: A Self Regularized Non-Monotonic Neural Activation Function. arXiv:1908.08681 pp. 1–13 (2019)
10. Otegi, A., Agirre, A., Campos, J.A., Soroa, A., Agirre, E.: Conversational question answering in low resource scenarios: A dataset and case study for basque. In: Proceedings of The 12th Language Resources and Evaluation Conference. pp. 436–442 (2020)
11. Pavanelli, L., Rubel Schneider, E.T., Bonescki Gumiel, Y., Castro Ferreira, T., Ferro Antunes de Oliveira, L., Andrioli de Souza, J.V., Meneghel Paiva, G.P., Silva e Oliveira, Lucas Emanuel Cabral Moro, C.M., Cabrera Paraiso, E., Labera, E., Pagano, A.: PUCRJ-PUCPR-UFMG at eHealth-KD Challenge 2021: A Multilingual BERT-based system for Joint Entity Recognition and Relation Extraction. In: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2021) (2021)
12. Piad-Morffis, A., Gutiérrez, Y., Estevez-Velarde, S., Almeida-Cruz, Y., Muñoz, R., Montoyo, A.: Overview of the eHealth Knowledge Discovery Challenge at IberLEF 2021. Procesamiento del Lenguaje Natural **67**(0) (2021)
13. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. Journal of Machine Learning Research **21**, 1–67 (2020)
14. Ramshaw, L.A., Marcus, M.P.: Text chunking using transformation-based learning. In: Natural language processing using very large corpora, pp. 157–176. Springer (1999)
15. Sanh, V., Debut, L., Chaumond, J., Wolf, T.: DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. In: Proceedings of the 5th Workshop on Energy Efficient Machine Learning and Cognitive Computing (EMC2) co-located with the Thirty-third Conference on Neural Information Processing Systems (NeurIPS 2019). pp. 1–5 (2019)
16. Santra, B., Anusha, P., Goyal, P.: Hierarchical transformer for task oriented dialog systems. arXiv preprint arXiv:2011.08067 pp. 1–10 (2020)
17. Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., Tsujii, J.: BRAT: A Web-based Tool for NLP-assisted Text Annotation. In: Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL '12). pp. 102–107 (2012)
18. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention Is All You Need. In: Proceedings of the Thirty-first Conference on Advances in Neural Information Processing Systems (NeurIPS 2017). pp. 5998–6008 (2017)
19. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Brew, J.: HuggingFace's Transformers: State-of-the-art Natural Language Processing. arXiv:1910.03771 pp. 1–11 (2019)
20. Wu, Y., Schuster, M., Chen, Z., Le, Q.V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al.: Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv:1609.08144 pp. 1–23 (2016)
21. Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A., Raffel, C.: mT5: A massively multilingual pre-trained text-to-text transformer. arXiv:2010.11934 pp. 1–17 (2020)
22. Zou, Y., Zhang, X., Lu, W., Wei, F., Zhou, M.: Pre-training for Abstractive Document Summarization by Reinstating Source Text. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 3646–3660 (2020)