Text-to-Text Paradigm for Italian Text classification

Michele Papucci and Chiara De Nigris

September 2022

Abstract

Text-to-text paradigm made it possible to achieve state of the art results in NLP tasks using a single and simple framework. The aim of this work is to test the capabilities of a generative pre-trained model on Italian language, IT5 (Sarti and Nissim 2022), for text classification, comparing them with a state of art baseline model such as BERT (Devlin et al. 2018). Even if this pattern has already been tested with success for different languages, such as English (see Raffel et al. 2020), it has not been tested for Italian yet. To assess the results on classification task, there have been carried out different experiments on the effect of label representation and on the possibility to train the model simultaneously on more tasks.

1 Data understanding and preparation: TAG IT

Data for the experiment have been taken from TAG-IT (Cimino et al. 2020), a dataset defined for an EVALITA 2020 task, which contains posts written in Italian collected from different blogs, labeled with the age and gender of the writer and topic of the sentences.
The original tasks for EVALITA was to use a collection of post of the same author and try to predict the gender and age of the writer and the topic of the sentences.
The dataset is composed by four variables: *Sentence*, the text contained in the post, and three different target variables, *Topic*, *Age* and *Gender*. Data semantic is explained in some details in Table 1.
For what the label distribution is concerned, classes are allocated as shown in graphs in Figures 1, 2 and 3.

As it is possible to ascertain looking at the plots, *Age* target variable is the one which presents a more balanced distribution among the classes, even if the *'20-29'* label is the majority class. As for *Gender*, it is clear that sentences classified as written by men are absolutely more than the ones written by women, determining a strongly unbalanced distribution of the two classes. The last variable, *Topic* , presents 11 labels, 3 of which (ANIME, SPORTS and AUTO-MOTO) result particularly frequent compared to the others.

It has been decided to use the data in a slight different way, trying to predict the age, gender and topic, not of a collection of posts, but of a single one. This made the task more difficult, but it created a lot more data to work on.

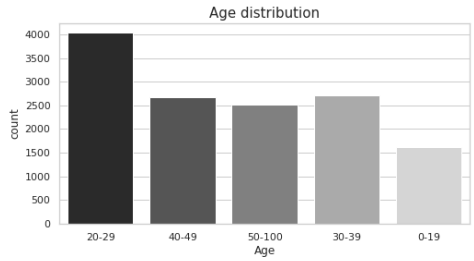

Figure 1: Age distribution

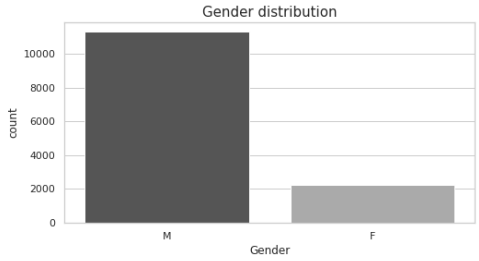

Figure 2: Gender distribution

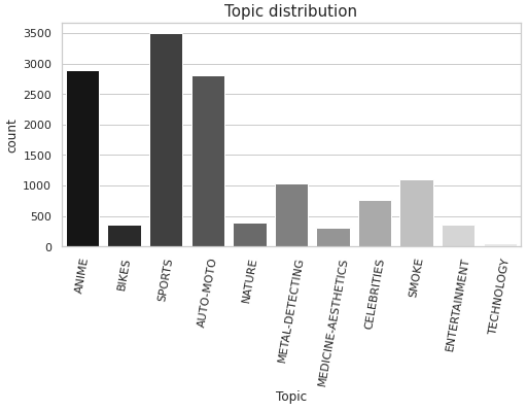

Figure 3: Topic distribution

Attribute	Data type	Description	Values
Age	String	Age of the writer	Captured in five ranges: 0-19, 20-29, 30-39, 40-49 and 50-100
Gender	String	Gender of the writer	M, F
Topic	String	Topic of the post	Eleven possible categories: ANIME, AUTO-MOTO, BIKES, CELEBRITIES, ENTERTAINMENT, NATURE, MEDICINE-AESTHETIC, METAL-DETECTING, SMOKE, SPORTS, TECHNOLOGY

Table 1: TAG-it semantic

Since a lot of the sentences were very short and with little information to work out even the topic of the sentence, it has been decided to use Stanza (Qi et al. 2020) to tokenize the sentences and remove the ones which were shorter than 10 tokens.
After these operations the training set (composed of 13553 sentences) has been used to train the models and the test set (5055 sentences) to evaluate the obtained results.

For what data preparation is concerned, in order to use BERT the three target variables have been converted to numeric labels, whereas for T5 they have been pre-processed as follow:

- *Gender* values have been transformed in *uomo* and *donna*;
- *Topic* categories have been translated in italian, written lowercase and, in cases in which were composed by more than one word, truncated in a single one. New labels are the ones that follow: *anime, automobilismo, bici, sport, natura, metalli, medicina, celebrità, fumo, intrattenimento, tecnologia*;
- *Age* has been left as it was.

2 Models

2.1 BERT

BERT (Bidirectional Encoder Representations from Transformers) is a machine learning framework for natural language processing (Devlin et al. 2018). From an architectural point of view, BERT is a multi-layer bidirectional Transformer encoder based on Transformers, a deep learning model whose main characteristic is the connection between each input and output element and the dynamical calculation of weightings between them based on their connection. Thanks to the bidirectional training of Transformers, BERT encoder reads the input sequence of words all at once, being able to learn the context of a word based on the surrounding ones in both directions, not only from the left-context of the word. For this experiment it has been used the BERT base, which presents 12 layers of Transformer Encoder, to carry out a sequence classification task.

2.2 T5

T5 (Text-To-Text Transfer Transformer) is an encoder decoder Transformer model, based on the text-to-text paradigm (Raffel et al. 2020). In this experiment it has been used the T5 base version fine-tuned on Italian language (Sarti and Nissim 2022), which presents 12 encoder layers and 12 decoder layers. The challenging part of this experiment has been to study this model behavior in a classification task, in which it had to predict a single word corresponding to the target label.

To each input it has been added a prefix, following the *Fixed-prompt LM tuning* approach (for a survey see Liu et al. 2021), which uses prompts with fixed parameters to specify the model behavior. This prompting engineering implies to provide a textual template that is applied to every training and test example. Fixed-prompt LM tuning has been already successfully explored for text classification, allowing more efficient learning, more completely specifying the task. In this experiment, three different prompts have been used, one for each classification task: *"Classifica argomento"*, *"Classifica età"* and *"Classifica genere"*.

3 Workflow

To better assess T5 classification's performances, they have been defined different baselines: two Dummy classifiers and BERT.

3.1 Dummy baseline

As a baseline they have been created two different classifier for each task. These two use very simple strategy classification to compare T5 classification models against. The first one is the *most frequent dummy classifier* which always predict the most frequent label for each row; the second is the *stratified dummy classifier* that generates predictions by respecting the class distribution of the training data. The stratified differs from the first one because, instead of always predicting the majority class, it gives off random predictions trying to respect the class distribution of the target variable.

3.2 BERT baseline

BERT has been fine-tuned three times to create three different single-task sequence classification models, one for each variable: *Topic*, *Gender* and *Age*. To create a Multi task BERT model it has been necessary to slightly change the structure of the model. After the encoder layers, the linear classification layer has been substituted with three different, and parallel, classification linear layers. Each of these represents one of the three tasks and has an input size equal to the hidden layer dimension of the BERT encoder, and an output size equal to the number of classes for the associated tasks. For the fine-tuning it has been necessary to tweak the way the loss is calculated, adding each of the output layers' losses to the total model loss.

In the end it has been obtained a BERT model that, given an input sequence, predicts three different labels at the same time.

These four different BERT models have been used to see how well T5 performed for classification.

3.3 T5 for text-to-text sequence classification

Regarding T5 the followed procedure has been similar to the BERT's one, creating three different single-task classification models and a multi-task classification one.

For the single tasks, T5 has been fine-tuned using as inputs two different sequences: the first one, which is the prompt, has been composed by the task prefix followed by the sentence to predict the class of, and the second sequence, which is the expected generation output, was the label associated with the sentence.

For the multi-task T5, each sentence has been presented to the model during the training three times, one for each task, each time with the appropriate prefix and label for the task. This means that, even if the input sequences were not exactly the same, the multitask model saw part of the input sequence, the sentence itself, three times instead of just one.

4 Results and Evaluations

As explained in Raffel et al. 2020 to compare different architectures (like T5 and BERT) it would be ideal that they present meaningful similarities as, for example, the number of parameters or the amount of computation to process a input-output sequence. Unfortunately, considering the two adopted models, it hasn't been possible to guarantee both these aspects, since a T5 with L layers has approximately the same number of parameters as a BERT with 2L layers, but also the same amount of computational cost as a BERT with L layers.
For the experiments it has been decided to compare T5 base (220 million parameters) against BERT base (110 million parameters).

After the training phase, each model has been evaluated with F-Scores measures. The choice of the metric has been due to the strongly unbalanced distributions of the three target variables (reported in Section 1), reason why other evaluation metrics, as accuracy or precision, seemed to be less suitable for the considered data.

The following tables summarize results obtained on the three different target variables by the six tested classifiers taking into consideration F1-score.

Topic classification		
	macro avg f1-score	weighted avg f1-score
Dummy (stratified)	0.09	0.17
Dummy (most frequent)	0.04	0.10
BERT	0.50	0.64
T5	0.19	0.41
Multi task BERT	0.51	0.65
Multi task T5	0.31	0.52

Table 2: Macro and weighted average f1-score on Topic classification

Age classification		
	macro avg f1-score	weighted avg f1-score
Dummy (stratified)	0.20	0.22
Dummy (most frequent)	0.09	0.14
BERT	0.32	0.33
T5	0.09	0.16
Multi task BERT	0.29	0.31
Multi task T5	0.16	0.23

Table 3: Macro and weighted average f1-score on Age classification

Gender classification		
	macro avg f1-score	weighted avg f1-score
Dummy (stratified)	0.50	0.68
Dummy (most frequent)	0.44	0.69
BERT	0.76	0.84
T5	0.31	0.70
Multi task BERT	0.74	0.84
Multi task T5	0.33	0.71

Table 4: Macro and weighted average f1-score on Gender classification

As is possible to ascertain from the previous Tables, BERT results the best model in classification task of all the three target variables, reaching really similar results in its single task and multi task versions.

For what T5 is concerned, the model achieves satisfying results compared to BERT on simpler tasks as

Gender and *Topic* classification, but returned very bad results in *Age* one. Indeed, on this task the Dummy classifier trained with the stratified strategy attained better results that T5, obtaining 0.11 more on macro average f1-score and 0.06 more on weighted average f1-score. It has to be considered that *Age* classification results to be the most difficult for all the six models, from the moment that BERT, which reaches the highest value of weighted average f1-score on this task, returned only 0.33.

From f1-scores reported in the Tables, it is clear that in its multi-task version T5 presents surely better results than the single task model, this may just due to the fact that the fine-tuned multitask model saw more times the sentences than the single task one and thus had a longer training, or that learning multiple tasks at a time, improved its generalization abilities (For a survey on multitask learning see Zhang and Yang 2017).

Even if T5 performances do not reach high results as BERT ones, future works might re-evaluate these results taking into consideration that some labels generated by T5 have been defined as erroneous because weren't exactly the one presented in the training set, but were actually more accurate than the original ones. For example a sentence has been classified wit the label *tabacco* (*tobacco*) insted of *fumo* (*smoke*). This aspect will be better analyzed in the following Section.

4.1 Errors analysis

As expected from a text-to-text model, sometimes T5 generated an unexpected output that wasn't one of the labels learned during the fine-tuning. Through error analysis for the Topic classification task, it turned out that the model appeared to have been able to generalize the task presented in the prompt *"Classifica argomento"* (*Classify topic*). In fact, sometimes the model returned more appropriate labels than the ones it has been trained with.

Sentence	Predicted label
Che bell'acqua e che bei vitellini! Grande Pres.!	animali
Perchè non l'alcool alimentare essendo neutro? E costa pure meno	alcol
terza miscela svizzera champagne eccellente! non vedo l'ora di tornare da two lions per altre miscele	bevande

Table 5: Exemples of T5 specific behaviour

As can be seen in Table 5, in the first row, the expected label was *celebrità* (*celebrities*) but the model generated *animali* (*animals*), which could be seen as a more appropriate topic label for the sentence.
To further prove that the model actually generalized the task, and didn't commit a mistake based on random noise token found in the sentences, it has been conducted a specific experiment detailed in the next Section.

It has also been noticed that sometimes IT5 is not able to generate meaningful labels at all, returning as output punctuation marks (as . or :) or single letters. Such cases are an exiguous number (less than 5 for each topic) and didn't have a real impact in the overall performance of the model.

5 Label representation experiments

As mentioned in the previous section, to actually evaluate the nature of obtained errors, it has been carried out a labels representation experiment, testing the connection between the semantic context of a sentence with the one of the label. In order to do this, it has been produced a shuffled version of both the train and the test set in which *Topic* labels have been randomly replaced with each other. This experiment produced fascinating results (reported in Tables 6), showing that the model lost 0.12 of macro average f1-score and 0.24 of weighted average f1-score on the shuffled *Topic* classification.

Error analysis on the shuffled *Topic* T5 model confirmed that T5 was able to generalize the task of *Topic* classification. This is corroborated by the evidence that, when presented with shuffled labels, the models stopped generating new and more specific labels for the sentences but simply got worse at predicting.
This is probably due to the fact that the T5 model actually learned during the fine-tuning process some semantic correlations between the encoding of the tokens in the prompt sentences and the encoding of the labels. This correlation went missing when the labels were shuffled, making the model worse at predicting and unable to generalize the task.

Topic classification		
	macro avg f1-score	weighted avg f1-score
Topic T5	0.19	0.41
Topic T5 sfuffled	0.07	0.17

Table 6: Macro and weighted average f1-score on Topic classification comparing T5 with its 'shuffled' version

The same experiment has been conducted also on the *Gender* classification, a task which does not present a clear semantic connection between the sentence and the label as *Topic* one does. Indeed, results reported in Table 7, shows that shuffling *Gender* binary labels does not have a significant effect on the f1 scores.

Gender classification		
	macro avg f1-score	weighted avg f1-score
Gender T5	0.31	0.70
Gender T5 shuffled	0.29	0.69

Table 7: Macro and weighted average f1-score on Gender classification comparing T5 with its 'shuffled' version

The last assessed experiment has been inspired from the study conducted by Chen et al. on the effect that the labels representation has on the performances of the model (Chen et al. 2020). Tests illustrated in the paper have been implemented using T5 and show that the choice of strings used to represent labels does not have a significant impact on overall classification results. Following this intuition, it has been tested to replace *Gender* labels (*uomo* and *donna*) with the original TAG-IT ones m (*maschio*) and f (*femmina*).

Gender classification		
	macro avg f1-score	weighted avg f1-score
T5 (m and f labels)	0.32	0.70
T5 (uomo and donna labels)	0.31	0.70

Table 8: Macro and weighted average f1-score on Gender classification with different labels representation

As shown in Table 8, changing label representation did not affect performances significantly, confirming Chen et al. 2020 findings that when the semantics of the label isn't relevant to the task even random strings function well as label representations. We saw however how in a different task like Topic classification, the label representation had a huge impact on performance.

6 Conclusions

From the conducted work it is possible to assert that a text-to-text framework can be used for a classification task, even if, in this work, T5 does not reach BERT performances. The most enthralling part of the study has been to notice that the model is actually able to apprehend the semantic connection that exists between input and output, as testified by the results provided on the shuffled data. T5 is, indeed, able to generalize the classification task and provide more specific labels which were not part of the training set.

References

Chen, Xinyi et al. (2020). "Label Representations in Modeling Classification as Text Generation". In: *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing: Student Research Workshop*, pp. 160–164.

Cimino, Dell'Orletta, and Nissim (2020). "TAG-it – Topic, Age and Gender Prediction". In: *EVALITA*.

Devlin, Jacob et al. (2018). "Bert: Pre-training of deep bidirectional transformers for language understanding". In: *arXiv preprint arXiv:1810.04805*.

Liu, Pengfei et al. (2021). "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing". In: *arXiv preprint arXiv:2107.13586*.

Qi, Peng, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning (2020). "Stanza: A Python Natural Language Processing Toolkit for Many Human Languages". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. URL: https://nlp.stanford.edu/pubs/qi2020stanza.pdf.

Raffel, Colin et al. (2020). "Exploring the limits of transfer learning with a unified text-to-text transformer." In: *J. Mach. Learn. Res.* 21.140, pp. 1–67.

Sarti, Gabriele and Malvina Nissim (Mar. 2022). "IT5: Large-scale Text-to-text Pretraining for Italian Language Understanding and Generation". In: *ArXiv preprint 2203.03759*. URL: https://arxiv.org/abs/2203.03759.

Zhang, Yu and Qiang Yang (2017). *A Survey on Multi-Task Learning*. DOI: 10.48550/ARXIV.1707.08114. URL: https://arxiv.org/abs/1707.08114.