EVALUATING VERBAL PRODUCTION LEVELS

Fabio Fassetti[1] and Ilaria Fassetti[2]

[1]DIMES Dept., University of Calabria, Italy
[2]Therapeia, Rehabilitation Center, Italy

ABSTRACT

The paper presents a framework to evaluate the adequateness of a written text with respect to age or in presence of pathologies like deafness. This work aims at providing insights about verbal production level of an individual in order for a therapist to evaluate the adequateness of such level. The verbal production is analyzed by several points of view, categorized in six families: orthography, syntax, lexicon, lemmata, morphology, discourse. The proposed approach extract several features belonging to these categories through ad-hoc algorithms and exploits such features to train a learner able to classify verbal production in levels. This study is conducted in conjunction with a speech rehabilitation center. The technique is precisely designed for Italian language, however the methodology is more widely applicable. The proposed technique has a twofold aim. Other than the main goal of providing the therapist with an evaluation of the provided essay, the framework could spread lights on relationship between capabilities and ages. To the best of our knowledge, this is the first attempt to perform these evaluations through an automatic system.

KEYWORDS

Verbal production, Feature Extraction, Deep Learning.

1. INTRODUCTION

This work aims at analyze written texts and to evaluate the level of the text on the basis of several language features whose goal is to highlight different syntactic, morphological and structural characteristics of the discourse. The proposed framework consists in two subsystems: the Feature Extractor and the Classifier. The former is devoted to extract the considered features by the verbal production, while the latter is devoted to exploit extracted features to train a learner in order to build a model able to categorize verbal production in levels.

The main contributions of the work can be summarized in the following points:

1 The definition of relevant features and of associated extractors for building a suited data model for subsequent analysis;
2. The introduction of a novel kind of graph, called action graph, designed to capture the relationships between verbs in the same sentence;
3. The novel notion of asymmetric distribution divergence between distributions to compare feature values associated, respectively, with the analyzed text and with the dictionary;
4. The building of a classification framework to evaluate the level of written text and the relationship between levels and ages and between levels and pathologies;
5. As a further contribution, the framework is designed for Italian language which presents peculiarities that have to be managed and for which, in many cases, NLP, standing for Natural Language Processing, techniques are at a preliminary stage.

Natarajan Meghanathan et al. (Eds) : CoSIT, AIAPP, SIGL, CRIS, DMA, NLPML, CYBI - 2020
pp. 97-106, 2020. © CS & IT-CSCP 2020 DOI: 10.5121/csit.2020.100109

The work is organized as follows. Section 2 provides details about the feature extraction phase, subsequent Section 3 presents the classification phase, Section 4 describes experimental results and Section 5 draws the conclusions.

2. FEATURE EXTRACTION

The considered features are summarized in Table 1 and detailed in the following. In particular, for each feature category, an associated section reports the description and the extraction technique. Preliminarily, it is assumed that a discourse D is provided in input in form of written text T and that the set of lemmata L_D (or, simply, L if D is clear by the context) contained in D has been extracted.

CATEGORY	FEATURES
ORTHOGRAPHY	feat 11: Incorrect words.
SYNTAX	feat 21: Erroneous agreement of gender. feat 22: Erroneous agreement of number. feat 23: Erroneous agreement of person.
LEXICON (WORDS)	feat 31: Distinct words. feat 32: Synonyms. feat 33: Repeated words.
LEXICON (PREPOSITIONS)	feat 41: Distinct prepositions. feat 42: Synonyms. feat 43: Repeated prepositions. feat 44: Prepositional locutions.
LEMMATA	feat 51: Complexity of words. feat 52: Complexity of prepositions. feat 53: Non-common-use words. feat 54: Non-common-use prepositions.
MORPHOLOGY	feat 61: Number of grammatical errors. feat 62: Complexity of grammatical modes feat 63: Complexity of grammatical tenses
DISCOURSE (GRAPH)	feat 71: Number of children. feat 72: Depth. feat 73: Width of levels. feat 74: Tree width.

Table 1: Verbal production features

2.1. Ortography

2.1.1. Description

The feature belonging to this category is the number of incorrect words. The incorrectness of the word is to be intended from the orthographic point of view.

2.1.2. Technique

The developed framework exploits GNU Aspell, a free and open source software available at the official website aspell.net, which performs the spell check of the input essay and provides the number of orthographic errors.

2.2. Syntax

2.2.1. Description

The features belonging to this category are related the number of syntactic and morpho-syntactic errors. In particular, the system considers three features by counting the number of errors in agreement of three different categories: (i) gender, (ii) number and (iii) person. Note that the agreement is much more significant in Italian than in English since Italian is highly inflected. Indeed, as for gender and numbers there must be concord between

- noun and articles

il gatto [the cat (male)], *la gatta* [the cat (female)];
i gatti [the cats (males)], *le gatte* [the cats (females)];

- noun and adjectives

bel gatto [nice cat (male)], *bella gatta* [nice cat (female)];
bei gatti [nice cats (males)], *belle gatte* [nice cats (females)];

- verbs (past participle with the auxiliary verb to be, intransitive verbs and passive forms)

il gatto è scappato [the cat (male) has escaped], *la gatta è scappata* [the cat (female) has escaped]; *i gatti sono scappati* [the cats (males) have escaped], *le gatte sono scappate* [the cats (females) have escaped].

As for person, there must be concord between verbs and subjects and, in Italian, verb inflections are different for any person.

io cammino [I walk], *tu cammini* [you walk], *egli/ella cammina* [he/she walks],
noi camminiamo [we walk], *voi camminate* [you walk], *essi/esse camminano* [they walk].

2.1.2. Technique

In order to extract the above described features, a grammar checker that provides the number of errors is exploited.

2.3. Lexicon

2.3.1. Description

As far as lexicon richness is concerned, the technique takes into account both the use of prepositions and the use of terms (noun, verbs, adjectives, ...). For each of them, the technique considers three main aspects detailed next.

- *The number of distinct terms*. This feature aims at describing the wideness of the dictionary the individual knows.

- *The number of prepositional locutions*. This indicates the capability in using more complex forms of linking propositions.

- *The number of repeated words/synonyms*. This describes the capability of an individual in using different terms to represent the same semantical concept, in order to make the discourse richer and pleasant.

2.3.2. Technique

From the algorithmic point of view, the extraction of the former two features is straightforward, while the feature concerning synonyms is extracted exploiting EuroWordNet [Vossen, 1998], which is a multilingual database similar to WordNet but developed for several European languages, including Italian. Provided that this software is able to return for a given lemma ℓ the set of synonyms of ℓ, denoted as synset(ℓ), the algorithm, aimed at computing the following counters:

- the counter C_{rep}^{ℓ} representing the number of repetitions, in each form, of the lemma ℓ in the input text T

- the counter C_{ref}^{ℓ} representing the number of synonyms of the lemma ℓ in the whole language dictionary

- the counter C_{syn}^{ℓ} representing the number of synonyms of the lemma ℓ in the input text T

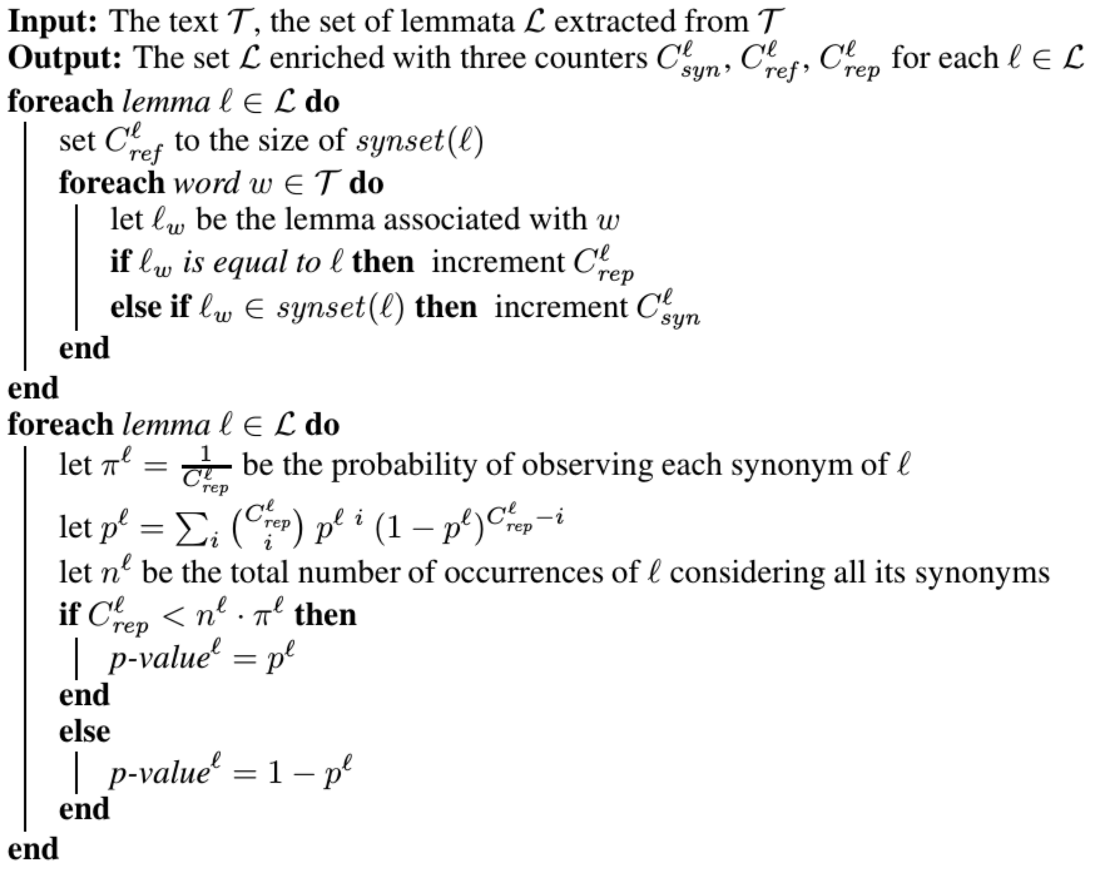
for each lemma ℓ, consists in the steps presented in the Algorithm 1.

After evaluating counters, the features are extracted by considering the probability of the observed number of occurrences of a certain synonym of a certain lemma, assuming an uniform distribution for the occurrences of the synonyms of a lemma. Thus, the p-value can be computed by considering the repetitions as a sequence of trials in a binomial distribution.

2.4. Lemmata

2.4.1. Description

An other important category of features is that of lemmata. Here, two different main aspects are taken into account, which are described next.



```
Input: The text $\mathcal{T}$, the set of lemmata $\mathcal{L}$ extracted from $\mathcal{T}$
Output: The set $\mathcal{L}$ enriched with three counters $C^{\ell}_{syn}$, $C^{\ell}_{ref}$, $C^{\ell}_{rep}$ for each $\ell \in \mathcal{L}$
foreach lemma $\ell \in \mathcal{L}$ do
    set $C^{\ell}_{ref}$ to the size of $synset(\ell)$
    foreach word $w \in \mathcal{T}$ do
        let $\ell_w$ be the lemma associated with $w$
        if $\ell_w$ is equal to $\ell$ then increment $C^{\ell}_{rep}$
        else if $\ell_w \in synset(\ell)$ then increment $C^{\ell}_{syn}$
    end
end
foreach lemma $\ell \in \mathcal{L}$ do
    let $\pi^{\ell} = \frac{1}{C^{\ell}_{rep}}$ be the probability of observing each synonym of $\ell$
    let $p^{\ell} = \sum_i \binom{C^{\ell}_{rep}}{i} \, p^{\ell\ i} \, (1 - p^{\ell})^{C^{\ell}_{rep} - i}$
    let $n^{\ell}$ be the total number of occurrences of $\ell$ considering all its synonyms
    if $C^{\ell}_{rep} < n^{\ell} \cdot \pi^{\ell}$ then
        $p\text{-}value^{\ell} = p^{\ell}$
    end
    else
        $p\text{-}value^{\ell} = 1 - p^{\ell}$
    end
end
```

Algorithm 1: Analysis of synonyms

- *Length of words/prepositions.*

- *Complexity of words/prepositions in terms of alphabet symbol sequence there contained.*Sequences consisting in alternation c-v (consonant-vowel), like c-a-s-a [house] (c-v-c-v), are easier than sequences containing groups of adjacent vowels, like c-u-o-c-o [cook] (c-v-v-c-v), that, in turn, are easier than sequences containing groups of adjacent consonants, like a-l-b-e-r-g-o [hotel] (v-c-c-v-c-c-v). This aspects is, also, related to difficulties in speaking and/or hearing, indeed individuals with this problems, often tend to avoid the use of lemmata with non-easy pronunciation.

- *Frequency of the lemma in the language.*This aspect consists in evaluating the refinement of the vocabulary the individual knows in terms of knowledge of less common lemmata. This is strictly related to the age, since the vocabulary grows with the child and to culture level.

2.4.2. Technique

For all of the above three features, five measures are considered: the *maximum*, the *average*, the *median*, the *standard deviation* and the *asymmetric distribution divergence*, a modified version of the Hellinger distance, between distribution associated with the feature at hand and the reference distribution taken from Italian dictionary.

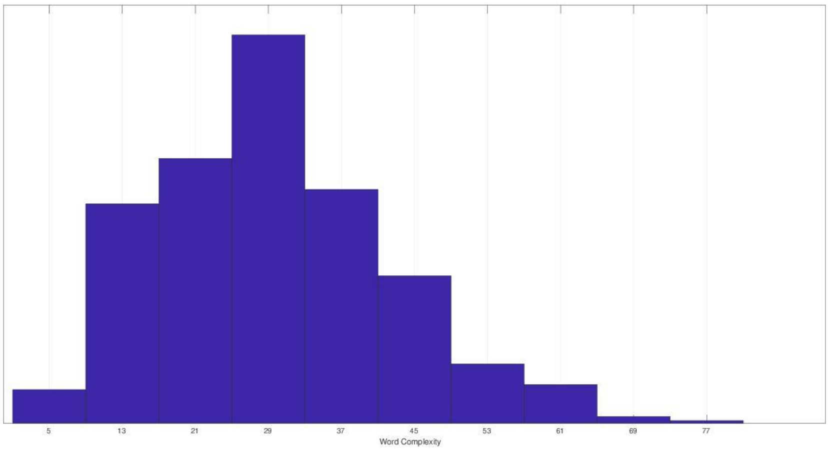


Figure 1: Complexity of words for Italian Language

In this section, details about the *asymmetric distribution divergence* are provided. Details about other features are omitted since the techniques for their extraction are straightforward. As starting point, the Hellinger distance is consider which is a well-known distance between statistical distribution and is introduced next. Given two probability functions P$_1$ and P$_2$, the *Hellinger distance* is

$$H^2(P_1,P_2) = \frac{1}{2}\int_{-\infty}^{+\infty}(\sqrt{dP_1}-\sqrt{dP_2})^2. \tag{1}$$

The aim is to consider separately contributions *high* and *low*, namely to measure if P_2 agree with P_1 (value 0), if P_2 values is meanly lower than P_1 (value ≥ -1), if P_2 values is meanly higher than P_1 (value ≤ 1). This is accomplished by separating contributions *low* and *high* of Equation (1).

By letting $E[P_1]$ be the expected value of P_1, Equation (1) can be rewrite as

$$H^2(P_1,P_2) = \frac{1}{2}\left(\int_{-\infty}^{E[P_1]}(\sqrt{dP_1}-\sqrt{dP_2})^2 + \int_{E[P_1]}^{+\infty}(\sqrt{dP_1}-\sqrt{dP_2})^2\right).$$

The modified version of this equation, called *asymmetric distribution divergence* and denoted as K^2, considers, instead of the sum between the two contributions, the difference, namely

$$K^2(P_1,P_2) = \frac{1}{2}\left(\int_{-\infty}^{E[P_1]}(\sqrt{dP_1}-\sqrt{dP_2})^2 - \int_{E[P_1]}^{+\infty}(\sqrt{dP_1}-\sqrt{dP_2})^2\right). \tag{2}$$

Theorem 1. The values assumed by the *asymmetric distribution divergence*, K^2, defined in Equation (2) are in the range $[-1,1]$.

Proof. Since the properties

$$\int_{-\infty}^{E[P_1]}(\sqrt{dP_1}-\sqrt{dP_2})^2 \geq 0 \quad and \quad \int_{E[P_1]}^{+\infty}(\sqrt{dP_1}-\sqrt{dP_2})^2 \geq 0$$

hold and since their sum H^2 satisfies the property $0 \leq H^2 \leq 1$, the properties

$$\int_{-\infty}^{E[P_1]} (\sqrt{dP_1} - \sqrt{dP_2})^2 \leq 2 \quad \mathit{and} \quad \int_{E[P_1]}^{+\infty} (\sqrt{dP_1} - \sqrt{dP_2})^2 \leq 2$$

hold and, then, the statement follows.

By construction, the next stated properties follow.

Property 1 [$K^2 = -1$]. The *asymmetric distribution divergence* approaches -1 if and only if for each x such that $P_2(x) > 0$ the properties $x < E[P_1]$ and $P_1(x) \approx 0$ hold, where $E[P_1]$ denotes the expected value of P_1.

Property 2 [$K^2 = 1$]. The *asymmetric distribution divergence* approaches 1 if and only if for each x such that $P_2(x) > 0$ the properties $x > E[P_1]$ and $P_1(x) \approx 0$ hold, where $E[P_1]$ denotes the expected value of P_1.

Property 3 [$K^2 \approx 0$]. The *asymmetric distribution divergence* approaches 0 if and only if for each x $P_2(x) \approx P_1(x)$.

2.5. Morphology

2.5.1. Description

Morphological aspects are an other relevant category of features that are considered. In details, the following items are evaluated (*i*) grammatical errors mainly concerning verb conjugations; (*ii*) verbs in terms of tenses and modes is evaluated in order to measure the ability in using "complex tenses" and "complex modes" of Italian verbs.

2.5.2. Technique

In order to extract the above described features, here the approach is to resort to a grammar checker and to a lemmatizer, provided by the *Linguistic Annotation Pipeline* Software[2], developed by *ItaliaNLP*, the Italian Natural Processing Laboratory www.italianlp.it. This tool provides also the part of speech tagging which allows to recognize modes and tenses of verbs.

2.6. Discourse

2.6.1. Description

Discourse analysis represents the more innovative part of the feature extraction phase. Here, the aim is to evaluate the richness of discourse structure, in terms of ability in joining propositions, in using subordinates or abbreviate forms like gerunds, and contextually to build a fluent discourse.

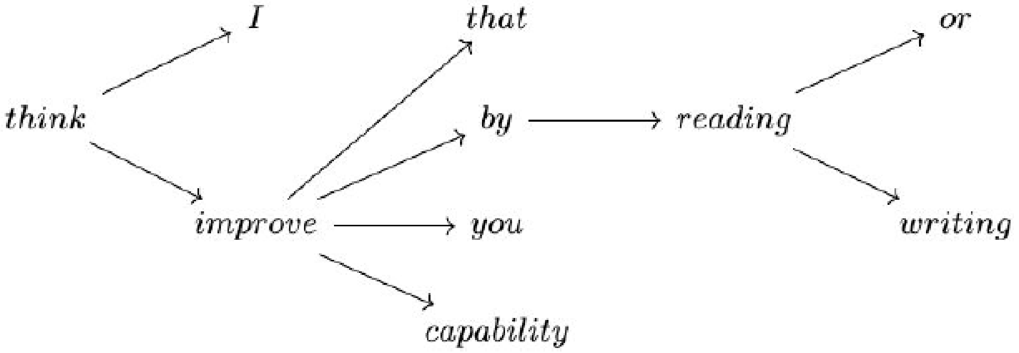
2.6.2. Technique

Basically, in order to evaluate discourse structure, we exploit again the *Linguistic Annotation Pipeline* Software, which provides the syntactic trees of the discourse, which here is called *discourse graph* and denoted as *DG*. However, the developed idea is to build two different directed acyclic graphs, the *discourse graph DG* and a *verb (action) graph VG* whose building is introduced in this work.

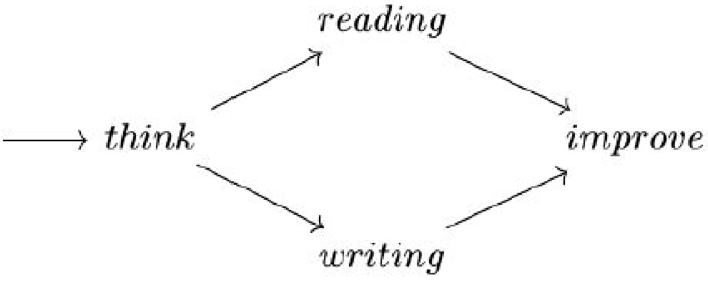
The discourse graph *DG* represents the whole discourse and consists in a pair ⟨*V*,*E*⟩, where *V* is the set of vertices and *E* is the set of edges. Given a discourse D and the set of lemmata *L* extracted from *D*, *DG*=⟨*V*,*E*⟩ is build as follows. There is one vertex v_*i*∈*V* for each lemma l_*i*∈*L* and there is an edge (*v_i*,*v_j*)∈*E* from vertex v_i to vertex *v_j* if and only if there is a semantic relationship between the lemma l_i and the lemma l_j. Conversely, the verb graph *VG* represents the relationships between verbs and takes into account "parallel" actions (coordinate propositions) and "sequential" actions (subordinate propositions). Formally, given a discourse *D*, the set of verbs V is extracted from *D*. Then, *VG*=⟨*V*,*E*⟩ is build as follows. There is one vertex *v*_i^v in V for each verb *v_i*∈*V* and there is an edge (*v_i*^v,*v_j*^v)∈*E* for each pair (*v_i*,*v_j*) of verbs semantically linked in the discourse.

Example 1. Consider the discourse *D* consisting in the single sentence: "I think that by reading or by writing you improve your speech capability".

Then, V_D={"*I*", "*think*", "*that*", "*by*", "*reading*", "*or*", "*by*", "*writing*", "*to*", "*improve*", "*speech*", "*capability*"} and the associated DG=⟨V_D, E_D⟩ is reported in the following figure.



Also, V_V = {"*think*", "*reading*", "*writing*", "*improve*"} and the associated $VG = \langle V_V, E_V \rangle$ is reported in the following figure.



Features. Several features are extracted by the two graphs aimed at measuring how much the discourse is articulated. In particular, the following graph properties are taken into account: (i) the number of children, (ii) the depth of the graph, (iii) the width of graph levels and (iv) the tree width , which evaluates how much the graph is distant from a tree. Note that the graphs are stratified, namely composed by linked layers.

3. VERBAL PRODUCTION CLASSIFICATION

The second phase of the framework consists in the classification step that provides the level of the written text at hand. The classifier exploits a *deep neural network* to accomplish the task. The proposed network consists in the following layers:

Input Layer.

The input layer is a *convolutional 1D* layer. In particular, consider the features as grouped by categories as reported in Table [table:features]. For each category C_i, consider the set $\{feat_1^i, \ldots, feat_{k_i}^i\}$, with k_i denoting the number of features in C_i, and let k be the least common multiple of integers k_i (then, $k = 12$ for the scenario in Table [table:features]). Features $feat_j^i$ in C_i are, then, replicated $\frac{k}{k_i}$ times, so that for each category there are exactly k features. Thus, convolutions of size k without stride and with $nf = 16$ filters are performed.

Hidden Layers.

There are three *dense* hidden layers each consisting in $nf \cdot k \cdot nc$ neurons, where nf is the number of filters, k is the number of features in each category and nc is the number of categories.

Output Layer.

The output layer is a *dense* layer with nl neurons, where nl is the desideratum number of levels the text has to be associated with.

4. IMPLEMENTATION AND EXPERIMENTS

All the parts of the framework, sketched in the subsections related to technical aspects about feature extraction, have been implemented in Python.

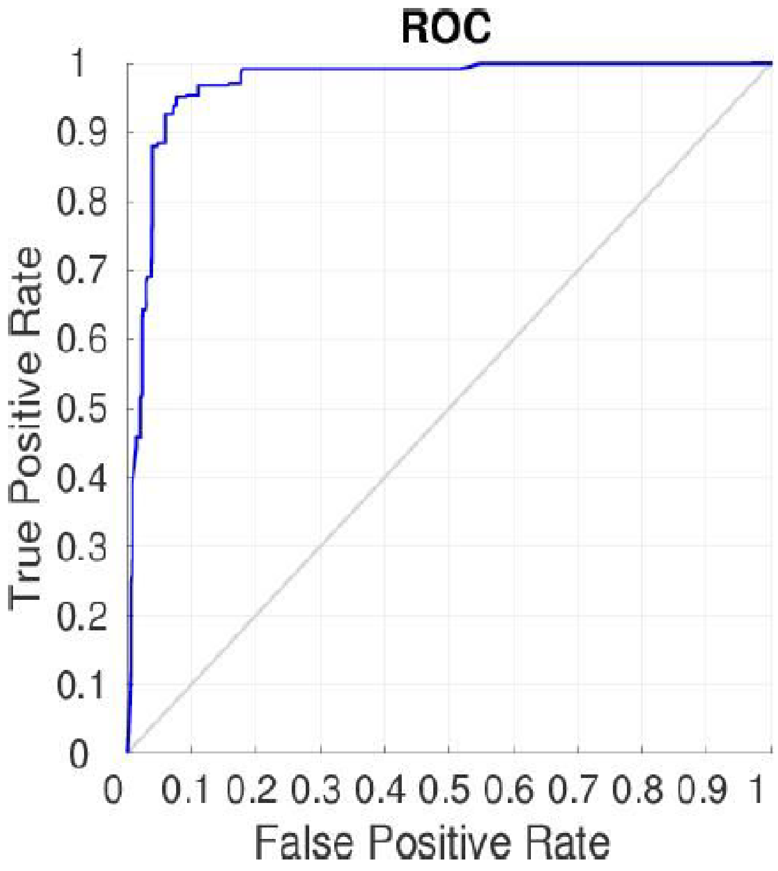


Figure 2: ROC Analysis

As far as the deep learning phase is concerned, *Keras*[3] , a high-level neural networks framework for Python running on top of the open source framework *TensorFlow*[4] , is exploited. Experiments have been conducted on a machine equipped with an Intel I7 processor with eight cores and 16 GB of RAM, running Linux operating system.

For this work, agreements with some Italian schools have been made in order to collect essays of students of different levels. In Italy, the school system includes 13 years of pre-university studies: primary school (ages 6 to 10), junior high school (ages 11 to 13), high school (ages 14 to 19). As for high schools, contacted students attend *lyceum*, which is the more theoretical school. Also, students of the former two university years have been involved (ages 20 to 21). Ages are then gathered in 5 groups each of them associated with 3 ages. Thus, the classification phase is designed with nl=5 classes. About 70 essays for each age have been collected till now and, then, for each group about 210 essays have been collected, even if this activity is still in progress by involving much more schools. Students involved in data collection are asked to write an essay with a fixed number of words using a plain editor without spell checker, grammar checker or word prediction. For very young children, essays have been reported in electronic form by teachers.

Figure 2 reports the ROC analysis conducted on the dataset at hand. After the classical phase of validation, the trained model is adopted to classify inadequate texts coming from foreign people, then with few mastery of Italian language. In this case, the accuracy computed by assigning the level of the input texts with the age and the schooling of the individuals, the system is drastically inaccurate as expected.

5. CONCLUSIONS

The proposed approach seems to be promising to evaluate the level of an input text. Experiments show that the classification accuracy is high and, importantly, it seems to recognize inadequate levels.

REFERENCES

[1] Nicholas Asher and Alex Lascarides. Logics of Conversation. Cambridge University Press, 2005.

[2] G. Attardi, F Dell'Orletta, M. Simi, and J. Turian. Accurate dependency parsing with a stacked multilayer perceptron. In Evaluation of NLP and Speech Tools for Italian (EVALITA), 2009.

[3] Giuseppe Attardi and Felice Dell'Orletta. Reverse revision and linear tree combination for dependency parsing. In Human Language Technologies, pages 261–264, 2009.

[4] Umberto Bertele and Francesco Brioschi. Nonserial Dynamic Programming. Academic Press, Inc., 1972.

[5] Felice Dell'Orletta. Ensemble system for part-of-speech tagging. In Evaluation of NLP and Speech Tools for Italian (EVALITA), 2009.

[6] Piek Vossen, editor. EuroWordNet: A Multilingual Database with Lexical Semantic Networks. Kluwer Academic Publishers, Norwell, MA, USA, 1998. ISBN 0-7923-5295-5.