Semantic coherence markers: The contribution of perplexity metrics

Semantic Coherence Markers:
the Contribution of Perplexity Metrics

Davide Colla[a], Matteo Delsanto[a], Marco Agosto[b],
Benedetto Vitiello[b], Daniele P. Radicioni[a,*]

[a] *University of Turin, Computer Science Department*
[b] *University of Turin, Department of Sciences of Public Health and Pediatrics*

Abstract

Devising automatic tools to assist specialists in the early detection of mental disturbances and psychotic disorders is to date a challenging scientific problem and a practically relevant activity. In this work we explore how language models (that are probability distributions over text sequences) can be employed to analyze language and discriminate between mentally impaired and healthy subjects. We have preliminarily explored whether perplexity can be considered a reliable metrics to characterize an individual's language. Perplexity was originally conceived as an information-theoretic measure to assess how much a given language model is suited to predict a text sequence or, equivalently, how much a word sequence fits into a specific language model. We carried out an extensive experimentation with healthy subjects, and employed language models as diverse as N-grams —from 2-grams to 5-grams— and GPT-2, a transformer-based language model. Our experiments show that irrespective of the complexity of the employed language model, perplexity scores are stable and sufficiently consistent for analyzing the language of individual subjects, and at the same time sensitive enough to capture differences due to linguistic registers adopted by the same speaker, e.g., in interviews and political rallies. A second array of experiments was designed to investigate whether perplexity scores may be used to discriminate between the transcripts of healthy subjects and subjects suf-

[*] Corresponding author
Email address: daniele.radicioni@unito.it (Daniele P. Radicioni)

fering from Alzheimer Disease (AD). Our best performing models achieved full accuracy and F-score (1.00 in both precision/specificity and recall/sensitivity) in categorizing subjects from both the AD class, and control subjects. These results suggest that perplexity can be a valuable analytical metrics with potential application to supporting early diagnosis of symptoms of mental disorders.

Keywords: diagnosis of dementia, perplexity, automatic language analysis, language models, early diagnosis, mental and cognitive disorders

1. Introduction

In economically developed societies the burden of mental disturbances is becoming more evident, with negative impact on people's daily life and huge cost for health systems. Whereas for many psychotic disorders no cures have been found yet, the treatment of people at high risk for developing schizophrenia or related psychotic disorders is acknowledged to benefit from early detection and intervention [1]. To this end, a central role might be played by approaches aimed at analyzing thought and communication patterns in order to identify early symptoms of mental disorder [2].

The analysis of human language has recently emerged as a research field that may be helpful to analyze for diagnosing and treating mental illnesses. In fact, in the last decade Natural Language Processing (NLP) techniques have become a common tool to support research on psychotic disorders. Namely, if language and its associated cognitive functions are first impaired before the full signs of mental disorders become apparent, linguistic analysis assisted by computing systems may be helpful for early detection. Recent advances in NLP technologies allow accurate language models (LMs) to be developed. These can be thought of as probability distributions over text sequences, and can be used to estimate in how far a text is coherent with (or, more precisely, predictable through) such language models. In order to measure the distance between an actual sequence of tokens and the probability distribution we propose using *perplexity*, a metric that is well-known in literature for the intrinsic evaluation of LMs. In this work we run experiments targeted at investigating how reliable perplexity is as a tool for investigating individuals' language, and we test whether the perplexity computed employing a language model acquired based on speeches from healthy subjects can be useful in discriminating healthy subjects from people suffering from mental disorders.

The work is structured as follows: in Section 2 we survey related work, and review the approaches that have been proposed for building systems to automatically recognize subjects affected by different forms of psychotic disorders based on linguistic analysis. In Section 3, we provide the essential background to the experiments: we first introduce language models and perplexity (Section 3.1), and illustrate the main traits of the neural architectures actually employed to acquire language models (Section 3.2). We then describe the experiments devised to explore whether perplexity is stable and can be reliably used to detect mental disturbances (Section 4): we first examine whether perplexity can be deemed as reliable to analyze speech transcripts under an intra-subject and discourse-level coherence perspective (Section 4.2); we then assess perplexity reliability under an inter-subjects perspective by considering how stable are perplexity scores for a given speaker when employing language models acquired or refined on other speakers' transcripts (Section 4.3); finally, we test perplexity to discriminate healthy subjects from subjects affected from Alzheimer Disease (Section 4.4). In the final Section we elaborate on the results and illustrate future work to improve the perplexity-based approach and make it a tool practically useful for diagnostic purposes.

Although in literature perplexity is not new as a tool to compare the language of healthy and diagnosed subjects, this work is, to the best of our knowledge, the first attempt at analyzing how suited perplexity is to analyze individuals' spoken language. While in previous reports the reliability of perplexity has been simply taken for granted, we investigate whether and to what extent perplexity scores are reliable before trying to use them to discriminate between mentally impaired and healthy subjects. Moreover, as far as we know, no previous work has compared perplexity scores computed through LMs as diverse as GPT-2 and

N-grams to the ends of discriminating healthy subjects from subjects afflicted by Alzheimer Disease. This difference has practical consequences for applications, mostly due to the different computational effort required both to train and employ such models, and to the descriptive power of the learned models.

2. Related Work

Patients with psychiatric disorders such as schizophrenia show various semantic disturbances, and may suffer from difficulties in handling linguistic meanings at different processing levels such as morphology, syntax, semantics, and pragmatics [3]. The work in [4] provides a rich overview on disturbances at the different levels. As far as we are concerned, disturbances related to schizophrenia typically produce abnormal usage of neologisms and word approximations, disruptions in language cohesion [5], syntactically simpler constructions featured by reduced use of embedded clauses and grammatical dependents [6], inflectional morphology variants and errors [7]. In the last decade, advances in NLP techniques have allowed the construction of approaches to automatically deal with tasks such as linguistic analysis and production, including also many of the aforementioned linguistic levels. These approaches have identified markers that can help differentiate patients with psychiatric disorders from healthy controls, and predict the onset of psychiatric disturbances in high risk groups at the level of the individual patient.

Early work in this area started with generating vectors from co-occurrence matrices [8, 9], treated with latent semantic indexing [10], or point-wise mutual information [11]. Such early distributional representations provided *explicit* (that is, directly meaningful and human-interpretable) information. The number of dimensions of such vectors was determined by the size of the vocabulary. On the other side, in *implicit* or latent representations, features were used resulting from Latent Semantic Analysis (LSA). LSA is a multidimensional associative model based on the distributional hypothesis: word meaning is encoded as a multi-dimensional (usually 300 or 400 dimensions) vector obtained by elaborating large *corpora* to estimate the co-occurrence frequencies for each word. A basic approach based on LSA, such as that described in the seminal work by [12], is as follows. Each input token is represented through a corresponding LSA vector, $W_i = \{I_{i1}, I_{i2}, \dots I_{iN}\}$. In turn, the vector representation for a phrase P is then built as the mean of the vectors representing all words in P: $P_i = \frac{1}{N}\sum_{k=1}^{N} I_{ik}$. The coherence between any two phrases is then computed through the cosine similarity of their corresponding vectors. The assumption underlying this approach is that meaningful texts will be featured by high coherence scores (in that words in the text being considered are semantically related on a distributional perspective), whilst text with some sort of disorder (or 'loose associations' among words) will be featured by reduced coherence scores. In [13] an artificial dataset built by intentionally manipulating existing texts was used to test the described notion of coherence: the minimum semantic distance and the mean semantic distance of adjacent sentences were found to be negatively correlated with the disorder level introduced in the original. In this work LSA (in conjunction with information on grammatical Part-of-Speech function, referred to as POS tags) has been used to predict the transition to psychosis in a clinical high-risk cohort.

More recently, LSA techniques have been superseded by neural approaches aimed at learning latent representations of words called word embeddings [14]. The overall design aimed at characterizing coherence (or, equivalently, the disorder associated with sentences and documents), by comparing vector representations of text excerpts, has remained unchanged. Among the most relevant sets of word embeddings, we mention Word2vec [15], GloVe [16], ConceptNet Numberbatch [17], fastText [18], LessLex [19], and NASARI [20].

A different approach to provide quantitative measures to language coherence and complexity is graph-based: in this setting, nodes represent words, and the word sequence is induced by directed edges. One main assumption underlying these approaches is that in coherent discourse neighboring words refer to connected topics, whilst incoherent discourse is associated with difficulties in making an ordered trajectory or path between topics. By employing tools from graph theory and information science it is possible to extract information on graph properties, such as connectedness, subgraphs or graph components. More specifically, measures such as entropy can be employed to probabilistically define topics and topic transitions [21]. Such graph representations also allowed grasping specific features of the normal and dysfunctional flow of thought (such as divergence and recurrence), and to produce accurate sorting of individuals affected by schizophrenia or mania [22]. In another study, techniques for speech graph analysis were employed to describe formal thought disorder, which has been mathematically defined by the linear combination of connectedness graph attributes and their degree of similarity to randomly generated graphs. Such connectedness attributes were mapped onto a Disorganization Index, and used to classify negative symptom severity [23].

In what follows we survey a set of works employing 'perplexity' that are specifically relevant to introduce our own proposal. Although originally conceived to assess how language models are able to model previously unseen data, perplexity can be used to compare (and discriminate) text sequences produced by healthy subjects or by people suffering from language-related disturbances. To provide a hint of this approach, perplexity is a positive number that —given a language model and a word sequence— expresses how unlikely it is for the model to generate that given sequence. A richer description of the perplexity is provided in Section 3.

In [24] N-grams of part of speech (POS) tags were employed to identify patterns at the syntactic level. Then, two LMs were acquired (one from patients' data and the other from data from healthy controls): the categorization of a new, unseen (that is, not belonging to either set of training data) sample was then performed through the perplexity computed with the two LMs over the sample. The considered sample was then categorized as produced by a healthy subject (patient) if the LM acquired from healthy subjects (patients) data attained smaller perplexity than the other language model. Perplexity has been recently proposed as an indicator of cognitive deterioration [25]; more specifically, the content complexity in spoken language has been recorded in physiological aging and at the onset of Alzheimer's disease (AD) and mild cognitive impairment (MCI) on the basis of interview transcripts. LMs used in this research were built by exploiting 1-grams and 2-grams information; as illustrated in next section (please refer to Equation 2), such models differ in the amount of surrounding information employed. Perplexity scores were computed on ten-fold-cross-validation basis, whereby participants' transcripts were partitioned into ten parts; a model was then built by using nine parts and was tested on the tenth. This procedure was repeated ten times so that each portion of text was used exactly once as the test set. Four examination waves with an observation interval of more than 20 years were performed, and correlations of the perplexity score of transcriptions dating to the beginning of the experiment were found with the score from the dementia screening instrument in participants that lately developed MCI/AD.

Perplexity has been employed as a predictor for Alzheimer Disease (AD) on the analysis of transcriptions from DementiaBank's Pitt Corpus, that contains data from both healthy controls and AD patients [26]. More precisely, in [27] two neural language models, based on LSTM models, were acquired, one built on the healthy controls and the other trained on patients belonging to the dementia group. A leave-one-speaker-out cross-validation was devised and, according to this setting, a language model $\mathcal{M}_{-s}$ was created for each speaker s by using all transcripts from the speaker's group but those of s. Data from speaker s was then tested on both $\mathcal{M}_{-s}$, thus providing a perplexity score p_{own}, and on the language model built upon the transcripts from the whole group to which the speaker did not belong to, thus obtaining the perplexity score p_{other}. The difference between the perplexity scores $\Delta_s = p_{own} - p_{other}$ was computed as a description for the speaker s. The classification of each speaker was then performed by setting a threshold ensuring that both groups obtained equal error rate. The authors achieved 85.6% accuracy on 499 transcriptions, and showed that perplexity can also be exploited to predict a patient's Mini-Mental State Examination (MMSE) scores. The approach adopted in this work is the closest to our own work we could find in literature; however it also differs from ours in some aspects. First, we investigated how reliable perplexity is in assessing the language of healthy subjects. That is, we analyzed how perplexity scores vary within the same individual, as an initial step toward assessing if perplexity is suitable for examining text excerpts/transcripts that (like in the case of the Pitt Corpus) were collected through multiple interviews and tests, spanning over years. Additionally, we were concerned with evaluating all excerpts from a single individual to predict the AD diagnosis at the subject level, rather than in predicting the class for each and every transcript. In order to assess the perplexity as a tool to support the diagnosis, we analyzed only data from subjects for which at least two transcripts were available.

Following the approach presented in [27], perplexity has been further investigated for the categorization of healthy subjects and AD patients [28]. In particular, different LMs have been acquired on both control and AD subjects' transcriptions from the Pitt Corpus [26]. Such LMs have been employed to evaluate in how far differences in perplexity scores reflect deficits in language use. In order to compute perplexity scores, the authors designed two experimental settings: *interrogation by perturbation*, where LMs were asked to assess corrupted texts so to simulate AD progression; and *interrogation by interpolation*, where the perplexity values obtained by LMs acquired on healthy subjects transcripts were combined with perplexity values computed through LMs trained on the AD patients. In the classification task, the authors achieved their best results by assigning higher relevance to scores computed through LMs acquired on AD class rather than those trained on healthy subjects (AUC 0.941 and 0.872 accuracy at equal error rate). Also interestingly, the experimentation provided evidence about the correlation among perplexity scores and lexical frequency: provided that subjects affected by Dementia of the Alzheimer's type tend to use higher frequency words with less specificity than control individuals, language models and perplexity were proved to be able to capture linguistic manifestations of the cognitive impairment. Our approach differs from this one. Firstly, we explored two different sorts of LMs (N-grams and GPT-2 models, fine tuned with 5, 10, 20 and 30 epochs) so to collect experimental evidence on the level of accuracy recorded by different LMs used to compute the perplexity scores. Secondly, four different decision rules were compared based on average perplexity scores from control and impaired subjects, along with their respective standard deviations. Moreover, while in [28] the categorization is performed at the transcript level, our focus is on the categorization of subjects. More in general, our study is aimed at providing a full account on perplexity, and not only at investigating how to employ it in a categorization task.

3. Background

Most approaches rely on a simple yet powerful descriptive (and predictive) theoretical framework which is known as *distributional hypothesis*. The distributional hypothesis states that words that occur in similar contexts tend to convey similar meanings [29]. For example, if the word w_i and the word w_k often occur in the same context, then they probably have close meanings; if they are interchangeable in the same contexts of occurrence, then they are synonyms. For example, in the sentences 'We used the *board* to shut down the power plant' and 'We used the *panel* to shut down the power plant', the words *board* and *panel* are intended with the same meaning. Several techniques have been devised to acquire the distributional profiles of terms, usually in the form of dense unit vectors of real numbers over a continuous, high-dimensional Euclidean space. In this setting each word can be described through a vector, and each such vector can be mapped onto a multidimensional space where distance (such as, e.g., the Euclidean distance between vectors) acts like a proxy for similarity, and similarity can be interpreted as a metric. As a result, words with similar semantic content are expected to be closer than words semantically dissimilar.

3.1. Language Models and Perplexity

Language Models (LMs) are a statistical inference tool that allows estimating the probability of a word sequence $W = \{w_1, \ldots, w_k\}$ [30, 31]. Such probability can be computed as

$$p(W) = \prod_{i=1}^{k} p(w_i | w_1, \ldots, w_{i-1}), \tag{1}$$

which is customarily approximated as

$$p(W) \approx \prod_{i=1}^{k} p(w_i | w_{i-N+1}, w_{i-N+2}, \ldots, w_{i-1}). \tag{2}$$

In the latter case only blocks of few (exactly N) words are considered to predict the whole W: we can thus predict the word sequence based on N-grams, that are blocks of two, three or four preceding elements (bi-grams, tri-grams, four-grams, respectively). In general N-gram models tend to obtain better performance as N increases, with the drawback of making harder the estimation of $P(w_N | W_{1,N-1})$. Another issue featuring these models stems from the fact that when increasing the context size, it becomes less likely to find sequences with the same length in the training corpus. In order to deal with N-grams not occurring in the training corpus, called out-of-vocabulary N-grams, language models have to add an additional step of regularization to allow a non-zero probability to be associated to previously unseen N-grams [32, 33]. The probabilities assigned by language models are the result of a learning process, in which the model is exposed to a particular kind of textual data. The goal of the learning process is to train the model to predict word sequences that closely resemble the sentences seen during training.

As mentioned, LMs are basically probability distributions of word sequences: perplexity was originally conceived as an intrinsic evaluation tool for LMs, in that it can be used to measure how likely a given input sequence is, given a LM [31]. This measure is defined as follows. Let us consider a word sequence of k elements, $W = \{w_1, \ldots, w_k\}$; since we are interested in evaluating the model on unseen data, the test sequence W must be new, and not be part of the training set. Given the language model LM, we can compute the probability of the sentence W, that is $\text{LM}(W)$. Such a probability would be a natural measure of the quality of the language model itself: the higher the probability, the better the model. The average log probability computed based on the model is defined as

$$\frac{1}{k}\log\prod_{i=1}^{k}\mathrm{LM}(W)=\frac{1}{k}\sum_{i=1}^{k}\log\mathrm{LM}(W),$$

which amounts to the log probability of the whole test sequence W, divided by the number of tokens in sequence. The perplexity of sequence W given the language model LM is computed as

$$\mathrm{PPL(LM,W)}=\exp\{-\frac{1}{k}\sum_{i=1}^{k}\log\mathrm{LM}(w_i|w_{1:i-1})\}. \tag{3}$$

It is now clear why low PPL values (corresponding to high probability values) indicate that the word sequence fits well to the model or, equivalently, that the model is able to predict that sequence.

3.2. Neural Architectures to Acquire Language Models

Modern neural networks organize neurons into layers, each unit being connected to each unit of the subsequent layer through *synapses* or *edges*. Each layer of the network accepts as an input the output of the preceding layer, performs some transformation of the received data, and produces an output according to the layer architecture. Different layers apply different transformations; edges, in turn, are usually provided with a weight, a real-valued number expressing the strength of the connection among two neurons, which is usually exploited to alter data coming from the preceding layer.

Since neural networks deal with real valued representations of data, we have to extract features from data, which is text in our case, and map them onto a numerical vector representation —usually called embedding— able to correctly grasp the main characteristics of the input data. The role of vector representations is central to neural models, and in fact, modern neural networks are provided with an embedding layer, which is responsible for the creation of a fixed-length vector for each element of the input sequence. It is worth noting that these vector representations mitigate the data sparsity problem by building

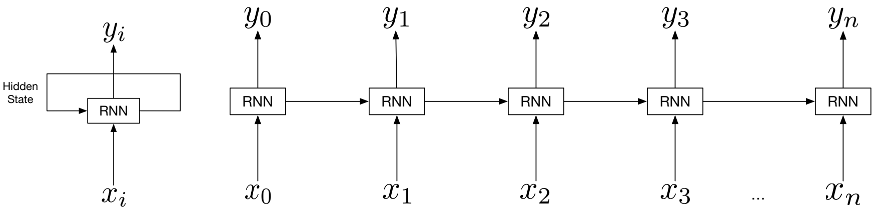


Figure 1: (Left) RNN unit: takes x_i as input, and computes the output representation y_i as a composition of x_i and the hidden state of the preceding time step. (Right) Representation of an RNN unrolled.

a continuous space, each word having its corresponding vector in the network space.

Given the relevance of word order in natural language sentences, neural models for NLP have to account for sequential properties in the input sequence. Recurrent Neural Networks (RNNs) are particularly suited to process sequential data such as natural language texts [34]. A graphical illustration of RNN model is presented in Figure 1: the last hidden state depends on the entire input sequence, that is, the prediction of the next word is conditioned on the previous words in the sentence. The ability of conditioning the prediction of the next word to the preceding context, that is, dealing with sequences, is the most appealing feature of the RNN architectures. Nevertheless, these models struggle to model the context when facing long range dependencies. Unfortunately, however, although they have been conceived to model sequential information, RNNs are not able to model broad range dependencies [35].

Given the difficulties in seizing long range dependencies, RNNs were replaced by the Long Short-Term Memory networks (LSTM) [36]. LSTMs are RNNs specifically designed to learn long range dependencies. This is obtained by providing units with an explicit context memory that conveys the information about the preceding context through the time steps. The context representation is achieved through two main operations: (i) forgetting information no longer needed from the context, and (ii) adding new information probably needed for next word prediction. Both sub-tasks are addressed through specialized neural

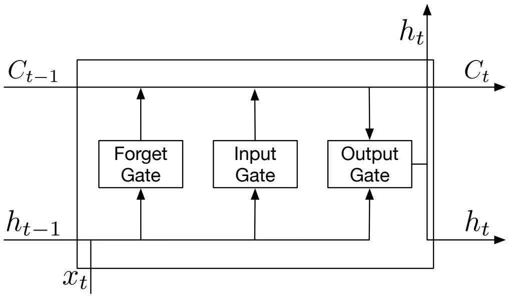


Figure 2: Representation of an LSTM unit. Here, C_{t-1} and h_{t-1} are the context representation and the hidden state coming from the preceding unit respectively. The input token is represented by x_t. The output of the cell corresponds to its hidden state at the current time step h_t. The updated representation of the context C_t and the hidden state h_t are then forwarded to the next LSTM unit.

units called gates, which manage the flow of information through the memory state and the output of the LSTM cell. A graphical illustration of an LSTM unit is depicted in Figure 2.

The described structure makes LSTMs particularly suited to deal with sequences and long range dependencies. However, simple LSTM models cannot naturally handle tasks in which input and output lengths are not equal, such as machine translation or speech recognition involve dealing with sequences whose length is not fixed beforehand. The Sequence-to-Sequence (S2S) model has been proposed to overcome such limitations [37]. The S2S model relies on LSTMs to map an arbitrary length sequence $x_1, \ldots, x_n$ to another sequence $y_1, \ldots, y_k$ where k may be different from n. In this setting, the input sequence is processed by an encoder, which compresses the sequence to a fixed length vector representation C. The decoder is then initialized on the C vector, and predicts the output token by token, accounting for the previously predicted token at each time step. A graphical representation of the S2S architecture is depicted in Figure 3.

Despite the ability of LSTM architectures to deal with long range dependencies, these models still struggle in representing larger pieces of text and suffer from high training time due to the recurrent connections which build these

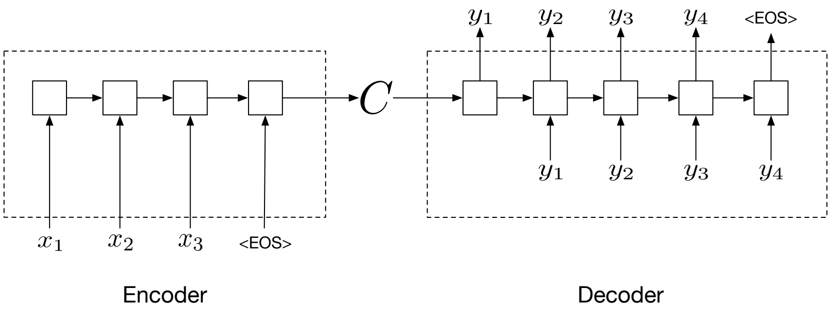


Figure 3: Representation of an S2S setting. Here <EOS> represents the end of the sentence. The input sequence x_1, x_2, x_3 is processed by the encoder and compressed to the context vector representation C. The context vector is then forwarded to the decoder which predicts the output sequence y_1, y_2, y_3, y_4 by taking as input the previously predicted token at each time step.

units. Additionally, the S2S architecture suffers from the loss of informative load in compressing the whole input sequence into a single fixed length vector representation. Transformers [38], together with the attention mechanism [39], alleviate these problems by both increasing the amount of exploited information from the context, and getting rid of the recurrent connections. The attention mechanism has been designed to alleviate the difficulties in S2S models; this is done by allowing the decoder to directly exploit the encoder's hidden states rather than just using the final context representation provided by the encoder itself. Adopting an attention mechanism allows the model to selectively focus on parts of the input that are likely to be the most useful for the task at hand. The attention mechanism is particularly suited to address tasks which need to take decisions relying on specific parts of the input data. Attention mechanisms plays a key role in the Transformer architecture; in particular, this model follows the S2S design pattern where the encoder processes the input sequence, the output is then forwarded to the decoder which is concerned with the output predictions. In this Section we will refer to the encoder-decoder model as to the Transformer block. Since Transformers get rid of recurrent connections, thereby allowing models to deal with sequences, the encoder represents the input through a combination of word embeddings and information about the position of words in the input sentence: in so doing, the model is able to account for ordering information. After this first operation, the encoder unit is made of an attention layer followed by a simple neural layer which is charged to compute the context representation. The decoder unit combines the previously predicted word representations with the positional information to keep track of the order of the words, and sends forward these vectors through an attention layer that is aimed at selecting the most useful information among the predictions. After these first steps the decoder combines the information from previously predicted tokens with the context representation, coming from the encoder, through another attention layer. Lastly, a simple neural layer is concerned with computing the output representation. Most popular models consist of several Transformer blocks stacked one on another: this allows the model to increase its abstraction capabilities (as the number of stacked blocks grows, the representation that can be calculated is more and more abstract [40, 41]). A graphical illustration of a transformer block is provided in Figure 4.

Transformers have been widely adopted and improved to address diverse Natural Language Understanding benchmarks, such as those in the GLUE [42] and SuperGLUE [43] benchmarks. The successful adoption of models such as BERT [44] and GPT-2 [45] in different application settings has attracted considerable efforts on improving such models [46, 47, 48, 49, 50]. One of the main applications of Transformers is the language modeling task: predicting the next word given the preceding context.

GPT-2 is a large language model based on Transformers and trained to predict the next word given the preceding context [45]. GPT-2, like traditional language models, predicts one token at a time, and the new prediction is appended to the input sequence for next time step. Inspired by the work in [51], where the Transformer-Decoder architecture was proposed, GPT-2 is made of stacked decoder units only. More precisely, while the Transformer-Decoder block is very similar to the decoder of the Transformer architecture, it gets rid of the encoder unit and of the contextual attention layer in the decoder. The GPT-2 model has

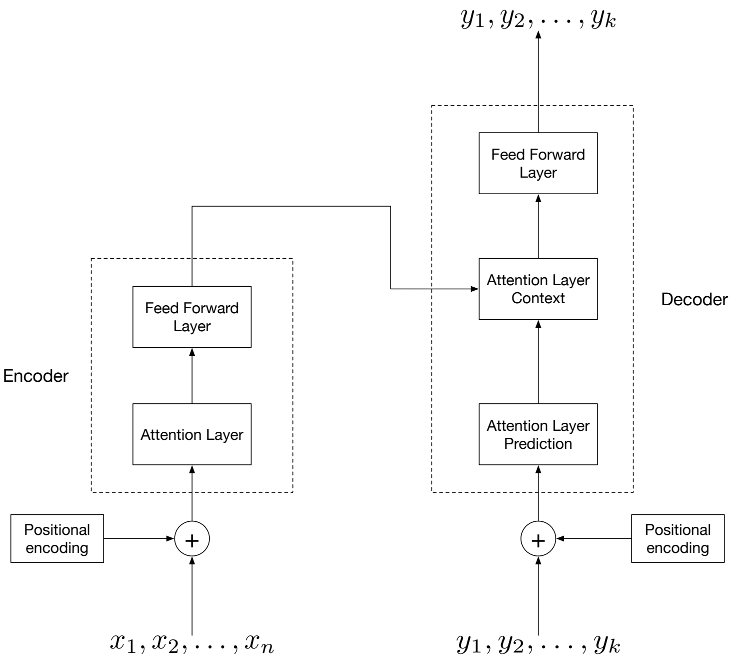


Figure 4: High level representation of transformer block. The input sequence $x_1, x_2, \ldots x_n$ is combined with positional information to account for ordering properties. The input is then processed from the attention layer of the Encoder and a simple neural layer aimed at representing the whole input sentence. The Encoder output is then combined with previously predicted tokens from the Decoder through another attention layer, and then, the last layer computes the output representation for each input token.

been trained on 40GB of Internet text carefully selected for quality, that is a selection of documents curated or supervised by humans. One main trait featuring the training data selection is that many different domains have been exploited as data sources; this allows the neural network to model language properties avoiding a strong polarization towards a specific domain. Additionally, it is worth noting that the number of stacked decoder units impacts performance, in that increasing the number of levels produces an improvement on the language modeling capabilities.

Neural language models are language models based on neural networks. Such models improve on the language modeling capabilities of N-grams by exploiting the ability of neural networks to deal with longer histories. Additionally, neural models do not need regularization steps for unseen N-grams and address the data sparsity curse of N-grams by dealing with distributed representation. The predictive power of neural language models is higher than N-grams language models given the same training set. Despite the great improvement of neural language models on NLP tasks, these models are affected by training time higher than N-grams language models.

4. Experiments

After having introduced the notion of perplexity and a brief description on modern neural architectures, we explore whether —and to what extent— the perplexity of LMs attained through such architectures can be used as a linguistic marker to detect language anomalies. Language anomalies detection may be helpful in recognizing mental disturbances and other disorders.

Before exploring perplexity as a tool suitable to discern linguistic anomalies in impaired subjects, we perform a preliminary step, consisting of checking whether perplexity scores can be considered as reliable. Informally stated, by reliable we intend that similar text documents —such as repeated interviews to the same subject over a limited time span, or descriptions by different subjects about the same scene— should be featured by analogous perplexity scores (by employing the same language model). We designed two experiments, the first one aimed at exploring the intra-subject reliability of perplexity scores, and the second one aimed at exploring inter-subjects reliability. In the former case we recorded the coefficient of variation (CV) —that is the ratio between standard deviation and average perplexity scores—, and in the latter one we measured the intraclass correlation coefficients (ICC) [52], that are two popular measures in the psychiatric psychometry community.

The whole experimentation presented in this Section is thus concerned with answering to two focal questions: 1) Whether perplexity scores are reliable within the same subject, but still sensitive enough to account for different sorts of speech forms produced by a given speaker (Experiment 1), and across subjects (Experiment 2); 2) whether the language of a specific class of subjects, diagnosed as suffering from disorders impacting on common linguistic abilities, can be automatically distinguished from that of healthy controls solely based on perplexity accounts (Experiment 3). In the first experiment we analyzed whether the LMs acquired by training both N-grams and GPT-2 on transcriptions of two different kinds of speech (two classes: political rallies *vs.* interviews) from a single subject produce different perplexity scores when the LM is used for analyzing similar (taken from same class) and different (from the other class) documents. In the second experiment we have measured the perplexity scores featuring discourses by 8 well-known political figures: in this case our aim was to assess whether perplexity scores computed based on models acquired on the other 7 speakers' transcripts are coherent in assessing the eight speaker. Finally, for the third experiment we have used the Pitt Corpus, from which we selected the transcripts of responses to the Cookie Theft stimulus picture [53], and investigated whether the perplexity score allows discriminating patients with dementia diagnosis (n = 194) from healthy controls (n = 99).

The code for replicating the experiments is available at `https://github.com/davidecolla/semantic_coherence_markers`.

4.1. Compared LMs

Three different experimental setups have been designed in order to compare perplexity as computed by language models acquired by training with two different sorts of architectures: N-grams, and GPT-2.

4.1.1. N-grams

Since N-grams implement the simplest language model with context, where each word is conditioned on the preceding N-1 tokens only, we adopted N-grams for the first experimental setup. For the sake of clarity we introduce the formalization for Bigrams; such formulation can be further generalized to any N. We define the probability of a sequence of words $W_{1,n} = \{w_1, w_2, \ldots, w_n\}$ as:

$$P(W_{1,n}) = \prod_{i=1}^{n} P(w_i|w_{i-1})$$

where the probability of each Bigram is estimated by exploiting the Maximum Likelihood Estimation (MLE) [54, Chap. 3].[1] According to the MLE, we can estimate probability of the Bigram (w_{i-1}, w_i) as:

$$P(w_i|w_{i-1}) = \frac{C(w_i|w_{i-1})}{C(w_{i-1})} \tag{4}$$

where $C(w_i|w_{i-1})$ is the number of occurrences of the Bigram (w_{i-1}, w_i) in the training set, while $C(w_{i-1})$ counts the occurrences of the word w_{i-1} only. It is worth mentioning that training Bigrams on a limited vocabulary may lead to cases of out-of-vocabulary words, i.e., unseen words during the training process. Out-of-vocabulary words pose a problem in calculating the probability of the sentence in which they are involved: in such cases we are not able to compute the probability of the Bigram involving the unknown word, thus undermining the probability of the whole sequence. In order to deal with out-of-vocabulary words, each token occurring only once in the training set can be replaced with the ‘unknown’ tag, UNK. In so doing, during the test phase we are allowed

[1] In this setting, stopwords are customarily not filtered, as providing useful sequential information.

to map each out-of-vocabulary word to the unknown word tag. Of course this procedure entails that the probability mass associated to UNK tokens tends to overestimate the role of such tokens, badly affecting the behavior of N-gram based models. Conversely, modern architectures such as GPT-2 are less impacted from out-of-vocabulary (OOV) issues: in fact, such models are acquired by employing huge amounts of data (in the order of 40 GB of text [45]) by using sub-word tokenizers and encoding strategies.

Notwithstanding the strategy for handling out-of-vocabulary words, we may still end up with unseen N-grams, formally occurring zero times in the training set, thereby resulting in a null probability. We addressed the unseen N-grams issue through the interpolated Kneser-Ney Smoothing technique [33]. The most effective smoothing techniques for N-grams involve exploiting lower-order representations so to improve the precision of higher-order N-grams whenever is needed. For example if the 3-gram (w_{i-2}, w_{i-1}, w_i) has zero evidence, we may either rely solely on the probability of its lower-order components, that are the bigrams (w_{i-2}, w_{i-1}), (w_{i-1}, w_i) and the unigrams (w_i), (w_{i-1}), (w_{i-2}) or combine the scores of its lower-order components to obtain the higher-order representation. The Kneser-Ney algorithm belongs to the family of interpolation strategies, and is based on the absolute discounting technique: to compute a precise probability distribution, we may need to *discount* the counts for frequent N-grams to save some probability mass to deal with unseen N-grams: in so doing, we subtract a small discounting factor d from the counts of N-grams to employ such discount as probability for unseen N-grams.

In the present setting we experimented with N-grams ranging from 2- to 5-grams; the Kneser-Ney discounting factor d was set to 0.1.[2]

The vocabulary was closed on each experiment: that is, the N-grams models employed in each experiment were acquired with the vocabulary obtained from the concatenation of the transcripts herein. Since the perplexity is bounded

[2] To compute N-grams we exploited the Language Modeling Module (*lm*) package from NLTK version 3.6.1, `https://www.nltk.org/api/nltk.lm.html`.

by the vocabulary size, fixing the cardinality of the vocabulary allows obtaining comparable perplexity scores from N-gram models trained across different corpora.

4.1.2. GPT-2

The second experimental setup that we designed exploits the GPT-2 neural model, in particular we used the GPT-2 pre-trained model available via the Hugging Face Transformers library.[3] In this setting, the input text has been preprocessed by the pre-trained tokenizer and grouped into blocks of 1024 tokens. The pre-trained model is specialized as Causal Language Model (CLM) on the input texts, that is, predicting a word given its left context. Since the average log-likelihood for each token is returned as the loss of the model, the perplexity of a text is computed according to Equation 3.

4.2. Experiment 1: Intra-subject and discourse-level coherence

The first experiment is aimed at investigating whether perplexity scores computed based on a given LM are stable, and whether perplexity scores are able to grasp factors specific to a given sort of speech. We have then targeted transcripts of two different kinds of discourse: the interview and the political rally. While in the former case both the questions put to the interviewee and his answers may be featured by different topics political rallies are events where people sharing similar political beliefs gather to support their candidate, whose language is in principle more regular, as not concerned with answering to specific questions. As regards as the linguistic register differentiating such transcripts, interviews should convey a sense of poise, balance, and posture, while the language adopted in rallies is expected to be more emphatic, direct, uniform and vehement. Our second research question was then whether the employed language models were able to recognize the two different linguistic registers.

[3] `https://huggingface.co/gpt2`

Table 1: Statistics describing the transcripts employed in Experiment 1: for all considered samples we report time duration, number of tokens, number of unique tokens, average number of tokens and of unique tokens, and type-token ratio (TTR).

Category	Transcript	Duration	Tokens	Unique Tokens	AVG Tokens	AVG Unique Tokens	TTR
Interview	I	1 : 28 : 52	7, 278	1, 098	8, 953	1, 185	0.13
	II	1 : 28 : 23	6, 471	922			
	III	1 : 31 : 34	18, 514	1, 926			
	IV	0 : 45 : 40	6, 702	1, 032			
	V	1 : 01 : 51	5, 933	946			
Rally	I	1 : 17 : 37	15, 200	1, 967	15, 051	1, 944	0.13
	II	0 : 56 : 17	10, 501	1, 614			
	III	1 : 43 : 43	20, 865	2, 300			
	IV	1 : 13 : 01	14, 056	1, 945			
	V	1 : 18 : 19	14, 806	1, 896			

4.2.1. Materials

We selected 10 transcripts by the former US President Donald Trump (this choice is mostly due to the large availability of his transcripts): 5 interviews and 5 campaign rallies were downloaded from the Rev platform.[4] Interviews were recorded between June 2019 and November 2020, while campaign rallies date to September and October 2020. The duration of both interviews and rallies varies between 45 minutes and one hour and 43 minutes. The statistics describing all transcripts employed in the first experimental setting, including time duration, token counts and type-token ratio (TTR, computed as the ratio between the types, that is the total number of different tokens occurring in a text divided by the total number of tokens) are reported in Table 1. While the initial choice of the transcripts was random within each category, we tried to select text excerpts of similar duration.

4.2.2. Procedure

Two types of model were acquired, one for Political Rallies and one for Interviews, and this schema was replicated for both N-grams and GPT-2. Each

[4] `https://www.rev.com`.

LM was then tested on *leave-one-out* basis on transcripts in the same category as the training/fine-tuning, and in direct fashion on transcripts from the other category. In the following we will simply refer to training, even though in a strict sense training procedures were employed to acquire N-gram models, while fine-tuning[5] is associated to the refinement step of the base GPT-2 model (more on fine-tuning in [55]). For example, in order to compute the perplexity score for excerpts from the Rally category with a language model obtained by training/fine-tuning on the same category, 5 models were built by using 4 of the 5 available transcripts (the fifth one was used for testing); results were then averaged over these 5 runs. Conversely, to compute the perplexity score on excerpts from the Interview category one LM was acquired from the Rally class, and used to test on all 5 transcripts. The same procedure was followed for the training/fine-tuning on the Interview category: leave-one-out schema for testing on transcripts from the same class, and only one model to compute the perplexity of transcripts in the other class.

As regards as the LMs acquired through GPT-2, the selected transcripts were employed for the fine-tuning. Provided that most systems cited in related literature adopt 20 epochs (an epoch being the hyperparameter that governs the number of complete runs all throughout the training dataset), we explored if either it is possible to obtain similar results with models tuned with less epochs, or higher categorization through further fine-tuning epochs. We thus experimented with 5, 10, 20 and 30 epochs, together with the pre-trained model. In order to compute the perplexity we adopted a *sliding window* of 50 tokens;

[5] Our distinction is compatible with a definition provided in literature: “In fine-tuning, we begin with off-the-shelf embeddings like word2vec, and continue training them on the small target corpus” [54, p.399]. Another popular description of the goals of fine-tuning comes from the work in [44]: “During pre-training, the model is trained on unlabeled data over different pre-training tasks. For fine-tuning, the [. . .] model is first initialized with the pre-trained parameters, and all of the parameters are fine-tuned using labeled data from the downstream tasks. Each downstream task has separate fine-tuned models, even though they are initialized with the same pre-trained parameters”.

Table 2: Perplexity (PPL) scores along with standard deviation scores obtained with fine-tuning on the transcripts from the Rally and Interview categories, averaged values for perplexity (PPL) scores, standard deviations and coefficient of variation (CV). The top four rows illustrate the results on the N-gram models, while the bottom rows show the results obtained by employing GPT-2 models, varying from 0 (pre-trained model) to 30 fine-tuning epochs.

Model		Rally						Interview					
		Rally			Interview			Rally			Interview		
		avg-PPL	avg-stdev	CV	avg-PPL	avg-stdev	CV	avg-PPL	avg-stdev	CV	avg-PPL	avg-stdev	CV
N-grams	2-gr	296.81	2.71	0.01	304.18	15.78	0.05	282.77	4.19	0.01	270.35	13.17	0.05
	3-gr	525.59	8.86	0.02	545.30	33.25	0.06	489.08	11.02	0.02	457.84	26.05	0.06
	4-gr	709.04	11.89	0.02	730.75	43.77	0.06	645.93	14.49	0.02	592.29	33.68	0.06
	5-gr	914.90	16.94	0.02	931.04	70.41	0.08	817.71	19.84	0.02	734.16	50.72	0.07
GPT-2	0 ep	27.16	1.20	0.04	24.15	1.55	0.06	27.16	1.20	0.04	24.15	1.55	0.06
	5 ep	18.29	0.66	0.04	17.64	1.21	0.07	20.22	1.00	0.05	18.44	1.34	0.07
	10 ep	16.36	0.55	0.03	17.02	1.35	0.08	18.59	1.01	0.05	17.22	1.19	0.07
	20 ep	15.16	0.51	0.03	17.16	1.61	0.09	18.18	1.06	0.06	16.80	1.12	0.07
	30 ep	14.86	0.50	0.03	18.05	2.02	0.11	18.50	1.14	0.06	17.12	1.19	0.07

window sizing was motivated by the fact that on average, the sentence length for transcripts is around 50 words (namely, 53.63 for interviews and 51.67 for rallies).

We then expected to observe analogous perplexity scores on all transcripts (as capturing common features underlying the language of the same speaker); and to observe slightly higher perplexity scores with models trained/fine-tuned on Interviews (Rallies) and used to test on Rallies (Interviews). In order to assess the reliability of PPL scores, we recorded the coefficient of variation (CV), which is computed as the ratio between standard deviation of the perplexity scores and the average of perplexity scores. A CV $\leq .1$ would contribute to support the hypothesis that perplexity provides stable and reliable scores.

4.2.3. Results

The results are presented in Table 2, where we recorded the average perplexity scores, their standard deviation and the coefficient of variation.

We observe that PPL scores grow monotonically from 2-grams to 5-grams; as regards as GPT-2 based models, PPL scores tend to decrease from the base model up to 20 fine tuning epochs, while they grow when computed though models acquired by 30 training epochs. In this case also standard deviation grows, which means that such models are overfitting to the fine-tuning data. The coefficient of variation is on average lower than 0.1 (we recorded .06 CV on the scores from GPT-2-based models, and .04 for the scores from N-gram models).

As regards as our second research question, whether perplexity allows recognizing different linguistic registers, we recorded a twofold result. In fact, while PPL scores acquired from rallies show reduced CV scores when tested on transcripts from the same class (Table 2), models acquired on interviews provide lower CV values when tested on rallies. This result may be explained to some extent with the observation that data available to train/fine-tune such models were roughly half of data available for rallies. We defer to future work a deeper investigation and experimentation on this point.

4.3. Experiment 2: Inter-subject coherence on different speakers

The second experiment was aimed at assessing whether perplexity scores are stable across subjects. Five transcripts with no specific topic for eight well-known past and present political figures were selected, and a language model for each subject was trained/fine tuned. The perplexity score was then computed for the speeches from each speaker, based on the others' language models (thus 7 LMs were used to compute the PPL scores for each one of the 8 speakers). In this case we expected to record analogous PPL scores by employing the models trained on the other speakers: a good agreement through models trained on different speakers would support the reliability of the PPL metrics.

4.3.1. Materials

In this case the context was less uniform than in the previous experiment, in that we collected political rallies, speeches on spot topics, such as economy, health systems, general challenges for the Western economy, a talk given in the frame of the World Economic Forum in Davos (Switzerland), civil rights, and so forth. Statistics describing time duration, number of tokens, number of unique tokens and type-token ratio describing the transcripts employed in this experiment are presented in the Appendix, in Table A.6.

4.3.2. Procedure

A speaker *vs.* speaker setup was implemented, that is all transcripts for each subject were employed to fine-tune a GPT-2 model or to acquire N-grams. The models obtained from each subject were then used to compute perplexity scores for the transcripts from other subjects. Similar to the former experiment, in all experimental conditions involving language models based on GPT-2 we compared results obtained through models refined with 5, 10, 20 and 30 fine-tuning epochs, with a sliding window sized to 50 tokens.

The transcripts of each speaker were 'rated' (with PPL scores) through the models acquired from the other seven speakers. The set of ratings collected for all speakers were then compared, to investigate to what extent the series of PPL scores can be deemed as reliable. To explore the reliability of the perplexity scores we employed the Intraclass correlation coefficients (ICC) [52]. In this setting, ICC values above 0.9 are recognized to indicate excellent reliability [56]. Six ICC variants may be overall considered, according to the choice of raters, and to whether a single measurement or the average of 2 or more measurements are employed [52], so that ICC models are featured by two parameters, as in ICC(X,Y). The former variable specifies the model, that is how raters are chosen. Model 1: each subject is assessed by a different set of randomly chosen raters; model 2: each subject is assessed by each rater, and raters are randomly sampled; model 3: each subject is assessed by each rater, and the set of raters is fixed. The second variable reports whether reliability should be computed based on a single measurement, or by employing the average of 2 or more measurements provided by different raters. We therefore chose the ICC(3,1) metric, that is each subject was assessed by all raters, and the set of raters kept constant (as indicated by the first argument, '3'); also, a single measurement was employed (as indicated by the second argument, '1').

4.3.3. Results

The detailed PPL scores and standard deviations recorded in the second experiment are presented in Table A.7 in Appendix A.2, which reports figures averaged over the 5 transcripts available for each subject. In Table 3 we provide the ICC scores obtained from those runs. The ICC scores show high (> 0.8) correlation for N-gram based models, and very high correlation (> 0.9) for GPT-2 based models. Thus we obtained good to optimal reliability for perplexity scores computed through the models at stake. As regards as N-gram models, the implementation employing closed dictionary over all subjects obtained substantially increased reliability scores with respect to the naïve implementation employing a dictionary closed given a single speaker. By inspecting the results obtained by both GPT-2 and N-grams-based models, we observe high ICC scores, that reduce as long as fine-tuning proceeds. This trend may be explained by noticing that when we extend fine-tuning, language models tend to be less general and to over-fit the language of an individual speaker, thereby becoming progressively less able to account for the language of all other ones. On the whole, these scores show that different GPT-2 based models do provide reliable PPL scores when used to assess the speeches of individual subjects. In this task there is no need for biasing models towards a specific subject's language, and fine tuning turns out to be detrimental to the reliability of PPL scores.

4.4. Experiment 3: Predictive and discriminative features of PPL

For this experiment we used publicly available data from the Pitt Corpus.[6] These data were gathered as part of a larger protocol administered by the Alzheimer and Related Dementias Study at the University of Pittsburgh School of Medicine [26]. In particular, we selected the descriptions provided to the Cookie Theft picture, which is a popular test used by speech-language pathologists to assess expository discourse in subjects with disorders such as dementia. This experiment was designed to investigate whether perplexity scores on the

[6] `https://dementia.talkbank.org/access/English/Pitt.html`.

Table 3: Intraclass correlation coefficients characterizing the perplexity scores obtained in the second experiment, in which each speaker was rated through LMs acquired/fine tuned based on transcripts from all other speakers.

Model	ICC(3,1) score
2-grams	0.88
3-grams	0.86
4-grams	0.83
5-grams	0.80
GPT2-5	0.98
GPT2-10	0.97
GPT2-20	0.94
GPT2-30	0.91

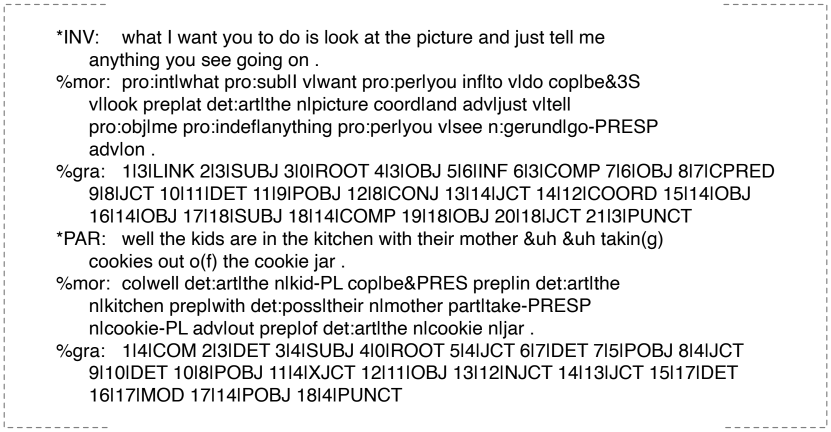
collected descriptions allow discriminating patients from healthy controls.

4.4.1. Materials

The dataset is composed of 552 files arranged into Control (243 items) and Dementia (309 items) directories. These correspond to multiple interviews to 99 control subjects, and to 219 subjects with dementia diagnosis. Text documents herein were transcribed according to the CHAT format,[7] so we pre-processed such documents to extract text. In so doing, the original text was to some extent simplified: e.g., pauses were disregarded, like hesitation phenomena, that were not consistently annotated [57, 58].

In Figure 5 we illustrate an excerpt, encoded in the CHAT format, taken from the Pitt Corpus. The transcriptions of the subject were selected —i.e., we retrieved only the lines starting with *PAR—, discarding the texts from the investigator. The CHAT transcription format is very rich and informative; for example, incomplete words are completed in a post-processing stage and marked through brackets, "bro" is represented as "bro(ther)"; pauses of the speaker are

[7] `https://talkbank.org/manuals/CHAT.pdf`.



```
*INV:    what I want you to do is look at the picture and just tell me
     anything you see going on .
%mor:  pro:int|what pro:sub|I v|want pro:per|you inf|to v|do cop|be&3S
     v|look prep|at det:art|the n|picture coord|and adv|just v|tell
     pro:obj|me pro:indef|anything pro:per|you v|see n:gerund|go-PRESP
     adv|on .
%gra:   1|3|LINK 2|3|SUBJ 3|0|ROOT 4|3|OBJ 5|6|INF 6|3|COMP 7|6|OBJ 8|7|CPRED
     9|8|JCT 10|11|DET 11|9|POBJ 12|8|CONJ 13|14|JCT 14|12|COORD 15|14|OBJ
     16|14|OBJ 17|18|SUBJ 18|14|COMP 19|18|OBJ 20|18|JCT 21|3|PUNCT
*PAR:   well the kids are in the kitchen with their mother &uh &uh takin(g)
     cookies out o(f) the cookie jar .
%mor:  co|well det:art|the n|kid-PL cop|be&PRES prep|in det:art|the
     n|kitchen prep|with det:poss|their n|mother part|take-PRESP
     n|cookie-PL adv|out prep|of det:art|the n|cookie n|jar .
%gra:   1|4|COM 2|3|DET 3|4|SUBJ 4|0|ROOT 5|4|JCT 6|7|DET 7|5|POBJ 8|4|JCT
     9|10|DET 10|8|POBJ 11|4|XJCT 12|11|OBJ 13|12|NJCT 14|13|JCT 15|17|DET
     16|17|MOD 17|14|POBJ 18|4|PUNCT
```

Figure 5: Excerpt from the Pitt Corpus: first interview with the subject #6 of the control group encoded in the CHAT format. The lines beginning with *INV and *PAR refer to the transcriptions for *Investigator* and *Participant*, respectively. Lines starting with %mor and %gra report both morphological and grammatical analysis of the transcript line. Interjections such as "uh" are marked with &, while incomplete words such as "takin" are completed in the transcript as "takin(g)".

marked through dots in brackets, for example (.) indicates a short pause while (...) refers to a longer pause; interjections are marked with the symbol &, for example "&uh" or "&ehm". Such elements were discarded; experiments exploiting this sort of information were left for future work. In particular lengthened syllables, long pauses and interruption symbols were eliminated, alongside a wide variety of sounds such as cries, sneezes, and coughs. Other meaningful aspects were preserved in the final file, such as repetitions, interjections and retracings, considering these events as important features for the model to capture. No information on intonational contours and other markers of the utterance planning process was available in the input files.

To the ends of collecting enough text to be analyzed, we dropped the interviews of subjects that participated in only one interview. We ended up with material relative to 74 control subjects (for which overall 218 transcripts were collected), and to 77 subjects with dementia diagnosis (overall 192 transcripts). The statistics describing number of tokens, number of unique tokens and

Table 4: Statistics describing the transcripts employed in Experiment 3. For each class we report the average number of tokens per interview, the average number of unique tokens per interview, the number of participants, the overall number of transcripts and the type-token ratio (TTR).

Class	AVG Tokens	AVG Unique Tokens	Participants	Transcripts	TTR
Control	437	26	74	218	0.07
Alzheimer's Disease	409	25	77	192	0.08

type-token ratio for the transcripts employed in the Experiment 3 are presented in Table 4.

4.4.2. Procedure

This experiment is aimed at testing the discriminative features of perplexity scores: more specifically, we tested a simple categorization algorithm to discriminate between mentally impaired and healthy subjects. We adopted the experimental setup from the work in [27]: two language models LM_C and LM_{AD} were acquired by employing all transcripts from Control and Alzheimer's disease groups, respectively. Such models are supposed to grasp the main linguistic traits of both groups speeches, thus representing the typical language adopted by subjects belonging to Control and AD classes. For both groups we adopted a leave-one-subject-out setting, whereby language models were refined with files from all other subjects within the same group except for one, which was used for testing. For each subject s we acquired the model LM_s on the transcripts from the same group of s, except for those of the subject s. Each transcript in the corpus was then characterized by two perplexity scores P_C and P_{AD}, expressing the scores obtained through language models acquired on Control and AD groups, respectively. More precisely, if a subject s was a member of the AD class, the scores P_C for its transcripts were obtained through LM_C, while the scores P_{AD} were computed by exploiting LM_s. Vice versa, if the subject s was from the Control group, the scores P_C for her/his transcripts were obtained through LM_s, while the scores P_{AD} were computed by exploiting LM_{AD}. Additionally, since we were interested in studying the scores featuring each subject, we synthesized the perplexity scores P_C and P_{AD} of each subject with the average of her/his transcripts scores, thus obtaining $\overline{P}_C$ and $\overline{P}_{AD}$.

In order to discriminate AD patients from healthy subjects, we adopted a threshold-based classification strategy. Three different approaches were explored to estimate such threshold:

(i) in the first setting we used the average perplexity scores characterizing all control subjects employed in the training process;

(ii) in the second setting we computed the threshold as the average perplexity score of all the subjects belonging to the AD class;

(iii) in the third setting we estimated two different thresholds by exploiting the difference $\overline{P}_{AD} - \overline{P}_C$, by initially following the approach reported in [27] and [28].

For each subject, the threshold estimation process was computed through a leave-one-subject-out setting, and repeated for the three approaches from (i) to (iii). In the first setting the threshold was estimated on all the subjects from the control group except for the test subject s: for each subject s we computed the threshold as the average of $\overline{P}_C$ scores for all subjects in the control group except for s —if s was from the healthy controls group—. In case the perplexity score $\overline{P}_C$ for the subject s was higher than the healthy controls threshold, we marked the subject as suffering from AD; as healthy otherwise. Similarly, in the second setting we computed the threshold as the average of $\overline{P}_{AD}$ scores for all subjects in the AD group except for s. In case the perplexity score $\overline{P}_{AD}$ for the subject s was higher than the average of AD class threshold, we marked the subject as healthy; as suffering from AD otherwise. The rationale underlying the first two settings is that each subject may be characterized more accurately by LMs acquired on transcript from the same group: in other words, we expected lower perplexity scores to be associated to control (AD) subjects, rather than subjects belonging to the other class, with LMs trained or fine-tuned on transcripts from control (AD) subjects.

Following the literature, in the third setting we characterized each subject with the difference $D = \overline{P}_{AD} - \overline{P}_C$. We defined two thresholds, $\overline{D}_{AD}$ which was computed as the average of all the difference scores from patients in the AD group and $\overline{D}_C$, defined as the average of all the difference scores from healthy controls. In both cases we considered all the patients belonging to the group except for the test subject s (s was held out with the only purpose to rule out her/his contribution from $\overline{D}_{AD}$ or $\overline{D}_C$). Different from literature —where equal error rate is used—, we employ $\overline{D}_{AD}$ and $\overline{D}_C$ as compact descriptors for the classes AD and C, respectively. The rationale underlying this categorization schema is that a subject is associated to the class that exhibits most similar perplexity score to her/his own. We categorize a subject s by choosing the class associated to the threshold (either $\overline{D}_{AD}$ or $\overline{D}_C$) featured by smallest margin with the PPL score computed based on a given LM for the transcripts from s, T_s according to the following formula:

$$\text{class}(s) = \underset{x \in \{C, AD\}}{\text{argmin}} \left| \text{PPL}(\text{LM}, T_s) - \overline{D}_x \right| . \tag{5}$$

This setting (involving $\overline{D}_{AD}$ and $\overline{D}_C$) will be referred to as $\overline{D}$.

Furthermore, we refined the decision rule $\overline{D}$ to account for standard deviation information. Together with the average $\overline{D}_{AD}$ and $\overline{D}_C$, we computed also σ_{AD} and σ_C as the standard deviations of the difference scores D for impaired and control groups. We explored the 3σ rule, which is a popular heuristic in empirical sciences: it states that in populations that are assumed to be described by a normally distributed random variable, over 99.7% values lie within three standard deviations of the mean, 95.5% within two standard deviations, and 68.3% within one standard deviation [59]. On this basis we explored the three options by adding 1, 2 and 3 standard deviations to average scores: the best results were obtained by employing 2 standard deviations. Our thresholds were then refined as follows:

$$\begin{aligned} \overline{D}^*_{AD} &= \overline{D}_{AD} + 2 \cdot \sigma_{AD}, \text{ and} \\ \overline{D}^*_C &= \overline{D}_C - 2 \cdot \sigma_C. \end{aligned}$$

The updated decision rule for categorization was then reshaped as

$$\text{class}(s) = \underset{x \in \{C, AD\}}{\text{argmin}} \left| \text{PPL}(\text{LM}, T_s) - \overline{D}^*_x \right|. \tag{6}$$

This setting, involving $\overline{D}^*_{AD}$ and $\overline{D}^*_C$, will be referred to as $\overline{D}^*$.

A twofold experimental setting has been devised, including experiments with N-grams and GPT-2, adopting a window size set to 20 in order to handle shorter text samples (the shortest text in the training data contains only 23 tokens). In the case of N-grams, the models were acquired for 2-grams to 5-grams; the GPT-2 model was fine-tuned employing 5, 10, 20 and 30 epochs.

4.4.3. Evaluation Metrics

To evaluate the results we adopted the Precision and Recall metrics (specificity and sensitivity) along with their harmonic mean, F1 score, and accuracy. Precision (specificity) is defined as $P = \frac{TP}{TP+FP}$, while Recall (sensitivity) is defined as $R = \frac{TP}{TP+FN}$. Informally stated, Precision computes the fraction of results that are actually correct: it is computed as the number of correct results (true positives, TP) divided by the sum of correct results (TP) and items mistakenly returned as results (false positives, FP). Recall computes how many correct results were individuated. In Recall, we have the number of correct results divided by the sum of correct results (TP) and items mistakenly not recognized as results (false negative, FN). While precision provides an estimation of how precise a categorization system is, recall indicates how many results were identified out of all the possible ones. F_1 measure is then used to provide a synthetic value of Precision and Recall, whereby the two measures are evenly weighted through their harmonic mean: $F_1 = 2 \cdot \frac{P \cdot R}{P+R}$

Accuracy was computed as $ACC = \frac{TP+TN}{P+N}$, that is as the fraction of correct predictions (the sum of TP and TN) over the total number of records examined (the sum of positives and negatives, P and N).

Finally, in order to record a synthetic index to assess accuracy and F1 scores on the two groups at stake, we used the harmonic mean among these three values. It was computed as

$$\mathrm{HM}(\mathrm{Acc.}, \mathrm{F1}_{AD}, \mathrm{F1}_{C}) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{x_i}$$

where n was set to the number of x_i values being averaged.

4.4.4. Results

The overall accuracy scores are presented in Figure 6, while detailed figures across different experimental conditions are presented in Table A.9, in Appendix A.4.

Let us start by reporting the results from N-gram models. The overall most effective strategy is $\overline{D}^*$ (Eq. 6), based on a threshold using the difference between AD patients and healthy controls, extended with the 3σ rule. The best performing model is based on Bigrams, and obtained .93 accuracy, .92 F1 score on the AD class, and .93 F1 score on the C class. The models employing PPL scores from the control group (indicated as P_C in Figure 6 and in Table A.9) obtained the lowest accuracy scores in all conditions, well below the random guess, while the accuracy yielded by the $\overline{P}_{AD}$ strategy is always above .5. In general we observe that increasing the length of the Markovian assumption reduces the accuracy of N-gram models for all decision rules (employing more context seems to be slightly detrimental for such models), with the exception of the $\overline{D}$ strategy.

The results obtained by the GPT-2 models reveal overall higher accuracy, ranging from .71 for the best model acquired with 5 epochs of fine-tuning to 1.00 for all further fine-tuning steps. The same profile describes the F1 scores recorded on the sub-tasks focused on AD and control subjects, respectively, varying from around 0.69 for the best model acquired with 5 epochs of fine-tuning ($\overline{D}$ strategy on the AD class) to 1.00 for all other models and sub-tasks. If we consider the efficacy of thresholding strategies and associated decision rules, the refined difference rule $\overline{D}$ is the best performing strategy for GTP-2 based models, as witnessed by the rightmost column in Table A.9. Such scores report the harmonic mean among accuracy, F1 score on categorization of AD

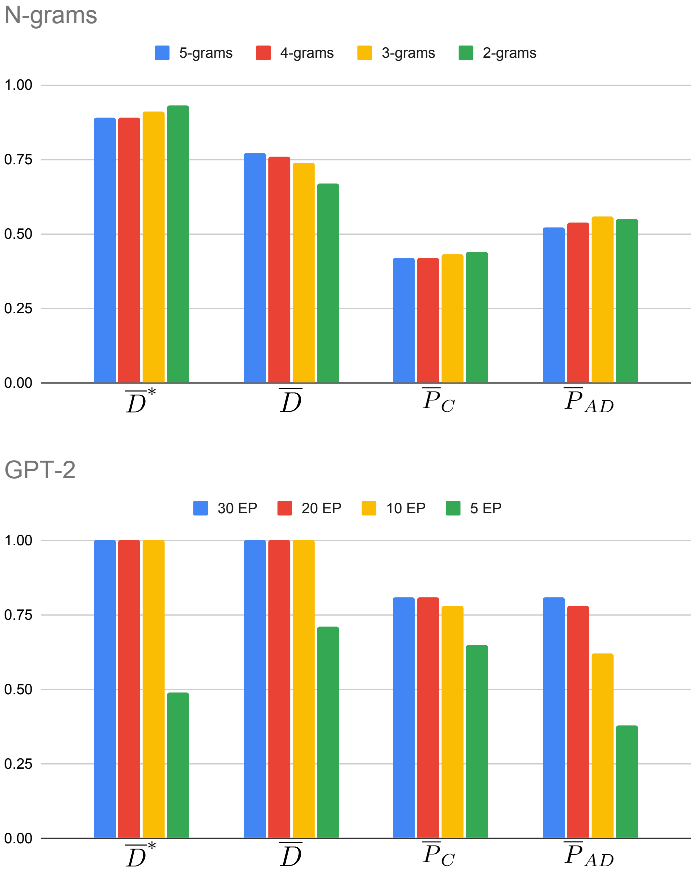


Figure 6: Plot of the accuracy scores for the third experiment on the categorization of AD/control subjects. The histograms in the top sub-figure show the accuracy on N-grams, while the histograms at the bottom report results obtained through GPT-2 models. Different colors correspond to N-gram of differing order and to different fine-tuning epochs, respectively. The histograms illustrate the scores obtained through $\overline{D}^*$, $\overline{D}$, $\overline{P}_C$ and $\overline{P}_{AD}$ decision rules, respectively.

Table 5: Study to compare the effectiveness of the thresholding and categorization strategies for each LM. The top scoring strategy is reported for each model.

N-gram models	categorization strategy	mean HM score
2-grams	$\overline{D}^*$	0.93
3-grams	$\overline{D}^*$	0.91
4-grams	$\overline{D}^*$	0.89
5-grams	$\overline{D}^*$	0.89
GPT-2 models: epochs	categorization strategy	mean HM score
5 epochs	$\overline{D}$	0.71
10 epochs	$\overline{D}, \overline{D}^*$	1.00
20 epochs	$\overline{D}, \overline{D}^*$	1.00
30 epochs	$\overline{D}, \overline{D}^*$	1.00

subjects and on categorization of control subjects. A compact view on data from the same column is provided in Table 5, illustrating the best strategy for each model at stake.

To frame our results with respect to literature, let us start from the accuracy of the baseline clinical diagnosis obtained in the first version of the study by Becker and Colleagues [26]: it was 86%, and after considering follow-up clinical data this datum raised to 91.4%, with a 0.988 sensivity and 0.983 specificity. This is what subsequent literature considered as the gold standard against which to compare experimental outputs. We recall that such data are particularly relevant as human evaluation included various analytical steps, such as medical and neurologic history and examination, semistructured psychiatric interview, and neuropsychological assessments. Experimental results provided in subsequent work approach those ratings by employing solely transcripts of descriptions to a rather simple picture. A relevant work attained 85.6% accuracy through LSTM based models [27] in the categorization of individual transcripts. Such results were then replicated and improved in the work by [28], where the best reported model experimentally obtained a 0.872 accuracy.

General Discussion

Provided that our experimental results seem to outperform the accuracy scores reported in literature, we feel that one main relevant result of this experimentation is that evidence was provided that perplexity scores are reliable at both intra-subject and inter-subject levels, and suited to categorize the language of subjects affected by cognitive impairments. In doing so, only speakers' transcripts were used.

Additionally, we realized that a short, controlled elicitation task can potentially outperform natural linguistic data obtained from speakers. The quality of our results needs be checked in different settings (further languages, varied experimental conditions: much experimental work still needs to be done), but this fact provides evidence that specialists may be effectively assisted by systems employing a technology based on language models and perplexity scores. Also, by comparing language models as different as N-grams and models based on the more recent GPT-2, we observed that Bigrams outperform a GPT-2 model fine tuned for 5 epochs. This fact may provide insights on the possible trade-off between accuracy of the results and computation time and costs.

While perplexity proved to be overall a viable tool to investigate human language, we found consistent differences in the outputs of the models at stake, mostly stemming from intrinsic properties of the LMs, from the amount of context considered by the models, from the size of available training data, and from the amount of training employed to refine models themselves. One first datum is that even though N-grams can be hardly compared to GPT-2-based models, nonetheless it may be helpful trying to discern the scenarios in which such models provide better results. In Experiment 1, which can be considered as a rather favorable experimental setting for N-gram models, recorded CV scores are on par or smaller than those obtained through GPT-2 based models. In Experiment 2 the ICC scores characterizing N-gram-based model output (ranging from 0.88 to 0.80) show valuable reliability. As anticipated, this is probably the most challenging setting for N-grams, in that the samples are featured by a consistent number of unique tokens and nearly doubled TTR with respect to documents employed in Experiment 1. Also, selected documents span various topics and a significant time frame, going from the mid Sixties to 2020. It was somehow surprising, then, that in Experiment 3 the accuracy level attained by the best-performing N-gram model (2-grams) achieved a 0.93 harmonic mean improving on the best GPT-2-based model (HM=0.73; please refer to Table A.9), fine tuned for 5 epochs and employing the $\overline{D}$ decision rule.

This result may be understood in the light of the rather regular language used for the descriptions to the Cookie Theft picture, that thereby turned out to be less demanding for the N-gram LMs. In these respects, a lesson learned is that N-grams can be employed in scenarios where the task is less difficult on lexical and linguistic accounts (recorded TTR values roughly range on 0.08, 0.25 and 0.13 for the Experiments 1, 2 and 3, respectively): in some instances of such problems adopting N-gram models may be convenient (considering both training and testing efforts) with respect to the more complete and computationally expensive Transformer models. Few data may be useful to complete this note on the trade-off between accuracy and computational effort. Our experiments were performed on machinery provided by the Competence Centre for Scientific Computing [60]. In particular, we exploited nodes with 2x Intel Xeon Processor E5-2680 v3 and 128GB memory. The first experiment took on average 12 hours for each GPT-2 LM, and about 5 minutes for all the N-gram models. The second experiment lasted about 32 hours for each GPT-2 LM and about 12 minutes for all the N-gram models, while the third experiment took around 8 hours for each GPT-2 setting and about 12 minutes for all the N-gram models.

5. Conclusions

The studies reported in this article have explored how suited perplexity is to function as a marker for measuring coherence in spoken language, and whether it can be used to support automatic linguistic analysis for clinical diagnoses. The diagnosis of dementia is a complex process that is long and labor intensive, involving a neuropsychiatric evaluation that includes medical and neurologic history and examination, semistructured psychiatric interview, and neuropsychological assessments [61, 62]. Being able to define a linguistic marker to detect symptoms of mental disorders would thus provide clinicians with automatic procedures for language analysis that can contribute to the early diagnosis and treatment of mental illnesses in an efficient and noninvasive fashion.

Two main research questions were addressed in this work. First, we have been exploring whether perplexity can be considered as a reliable metrics to analyze spoken language at large. To answer this question we designed an experiment to compare perplexity scores for different speeches from the same speaker (transcripts from an healthy subject were considered in this phase): two sorts of language —political rallies and interviews— were analyzed. In the second experiment we investigated the coherence of perplexity scores by comparing the speeches of eight well-known politicians. Each speaker's perplexity was rated through perplexity scores based on LMs acquired from the other seven speakers. The results of these studies seem to corroborate the hypothesis that perplexity can be measured in a reliable manner for the individual subject, while at the same time accounting for different linguistic registers. Differences in scores obtained through the application of different language models were detected and discussed. The perplexity computed through simpler LMs may be a good option when either language variability is reduced or training data ensure good coverage of the considered language. Conversely, simpler models may be misled by out-of-vocabulary terms: interestingly enough, however, even in these cases perplexity scores were consistent with the individual subject language characteristics. Reliability is a precondition to employ perplexity scores to assess trends in language production by a given subject, and also to compare perplexity scores across subjects. In turn, being able to compare perplexity scores associated to different subjects speeches may reveal in how far their language is accounted for by a given language model. Furthermore, the perplexity scores obtained by employing the base model GPT-2 were compared to those computed through the fine-tuned GPT-2 model, confirming that the fine-tuning step is a valuable tool for obtaining more accurate and reliable perplexity scores.

As to our second research question, we investigated whether and to what extent perplexity scores allow categorizing transcripts of healthy subjects and subjects suffering from Alzheimer Disease (AD). In this experiment we used a publicly available dataset, the Pitt Corpus. A widely varied experimental setting was designed to investigate the predictive and discriminative power of perplexity scores, and to assess how the resulting categorization accuracy varies in function of the amount of training/fine-tuning employed to acquire the LMs. We compared (2, 3, 4 and 5) N-gram models, 0 to 30 (GPT-2) fine-tuning epochs, and four different thresholding strategies, as well. Novel thresholds were proposed, and compared to those reported in literature: the newly proposed categorization strategies ensure consistent improvements over state-of-the-art results.

A final remark relates to an outlook on future work. Different language models can attain results possibly featured by analogous accuracy with a fraction of training/fine-tuning efforts: e.g., we conducted preliminary tests, not reported here for brevity, also on LSTMs that revealed poor performance, paired with a computational load higher than for the GPT-2 architecture.[8] Also, different categorization algorithms may be adopted to discriminate patients from control subjects; refinements to both employed LMs and overall categorization strategy may result in substantial improvements. Yet, further experiments are needed to assess perplexity on larger samples, and on different sorts of spoken language: as mentioned, the language required to comment the Cookie Theft picture is quite a regular one. A richer, fuller characterization of the discriminative power of perplexity scores will involve experimenting also on different languages, and the associated language models. However, the findings from this proof-of-concept study have several implications: perplexity has been exten-

[8] More specifically, perplexity scores computed through LSTMs were highly volatile (with standard deviation values often overcoming mean perplexity values), even increasing the number of training epochs, which required almost twice the time necessary to train the GPT-2 base model.

sively used to carry out experiments on general language from healthy subjects. Experimental evidence seems to support the hypothesis that perplexity scores can be reliably employed to assess how much language excerpts are consistent with a given language model. Also, when we moved to the challenging task of predicting whether the author of a transcript was afflicted by dementia or a healthy subject, we obtained valuable results, especially if we consider that our predictions were based solely on perplexity scores, with a substantial reduction in the amount of information with respect to the clinical evidence collected all throughout the diagnosis steps employed by human experts to face the same categorization task [26].

Appendix A. Sources of experimental material and detailed results

Appendix A.1. Material used in Experiment 1

The list of transcripts employed for training/fine tuning and testing, along with links to the `www.rev.com` platform can be found in the bundle containing the whole project, available at the URL `https://github.com/davidecolla/semantic_coherence_markers`.

Appendix A.2. Material used in Experiment 2

The list of transcripts employed for training/fine tuning and testing, along with links to the www.rev.com platform can be found in the bundle containing the whole project, available at the URL https://github.com/davidecolla/semantic_coherence_markers.

Appendix A.3. Statistics describing data and detailed results for Experiment 2

Appendix A.4. Detailed results for Experiment 3

References

[1] M. Marshall, S. Lewis, A. Lockwood, R. Drake, P. Jones, T. Croudace, Association between duration of untreated psychosis and outcome in cohorts of first-episode patients: a systematic review, Archives of general psychiatry 62 (9) (2005) 975–983.

[2] M. K. Larson, E. F. Walker, M. T. Compton, Early signs, diagnosis and therapeutics of the prodromal phase of schizophrenia and related psychotic disorders, Expert review of neurotherapeutics 10 (8) (2010) 1347–1359.

[3] J. N. de Boer, S. G. Brederoo, A. E. Voppel, I. E. Sommer, Anomalies in language as a biomarker for schizophrenia, Current opinion in psychiatry 33 (3) (2020) 212–218.

[4] M. A. Covington, C. He, C. Brown, L. Naçi, J. T. McClain, B. S. Fjordbak, J. Semple, J. Brown, Schizophrenia and the structure of language: the linguist's view, Schizophrenia research 77 (1) (2005) 85–98.

[5] N. M. Docherty, M. DeRosa, N. C. Andreasen, Communication disturbances in schizophrenia and mania, Archives of General Psychiatry 53 (4) (1996) 358–364.

[6] D. Çokal, G. Sevilla, W. S. Jones, V. Zimmerer, F. Deamer, M. Douglas, H. Spencer, D. Turkington, N. Ferrier, R. Varley, et al., The language profile of formal thought disorder, npj Schizophrenia 4 (1) (2018) 1–8.

[7] M. Walenski, T. W. Weickert, C. J. Maloof, M. T. Ullman, Grammatical processing in schizophrenia: Evidence from morphology, Neuropsychologia 48 (1) (2010) 262–269.

[8] D. Harman, Overview of the first trec conference, in: Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval, 1993, pp. 36–47.

[9] H. Schütze, J. O. Pedersen, A cooccurrence-based thesaurus and two applications to information retrieval, Information Processing & Management 33 (3) (1997) 307–318.

[10] T. K. Landauer, P. W. Foltz, D. Laham, An introduction to latent semantic analysis, Discourse Processes 25 (2-3) (1998) 259–284.

[11] D. Hindle, Noun classification from predicate-argument structures, in: 28th Annual meeting of the Association for Computational Linguistics, 1990, pp. 268–275.

[12] B. Elvevåg, P. W. Foltz, D. R. Weinberger, T. E. Goldberg, Quantifying incoherence in speech: an automated methodology and novel application to schizophrenia, Schizophrenia research 93 (1-3) (2007) 304–316.

[13] G. Bedi, F. Carrillo, G. A. Cecchi, D. F. Slezak, M. Sigman, N. B. Mota, S. Ribeiro, D. C. Javitt, M. Copelli, C. M. Corcoran, Automated analysis of free speech predicts psychosis onset in high-risk youths, npj Schizophrenia 1 (1) (2015) 1–7.

[14] R. Navigli, F. Martelli, An overview of word and sense similarity, Natural Language Engineering 25 (6) (2019) 693–714.

[15] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Advances in neural information processing systems, 2013, pp. 3111–3119.

[16] J. Pennington, R. Socher, C. D. Manning, Glove: Global vectors for word representation, in: Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1532–1543.

[17] R. Speer, J. Chin, An ensemble method to produce high-quality word embeddings (2016).

[18] P. Bojanowski, É. Grave, A. Joulin, T. Mikolov, Enriching word vectors with subword information, Transactions of the Association for Computational Linguistics 5 (2017) 135–146.

[19] D. Colla, E. Mensa, D. P. Radicioni, LessLex: Linking multilingual embeddings to sense representations of lexical items, Computational Linguistics 46 (2) (2020) 289–333.

[20] J. Camacho-Collados, M. T. Pilehvar, R. Navigli, NASARI: a novel approach to a semantically-aware representation of items, in: Proceedings of NAACL, 2015, pp. 567–577.

[21] Á. Cabana, J. C. Valle-Lisboa, B. Elvevåg, E. Mizraji, Detecting order–disorder transitions in discourse: Implications for schizophrenia, Schizophrenia Research 131 (1-3) (2011) 157–164.

[22] N. B. Mota, N. A. Vasconcelos, N. Lemos, A. C. Pieretti, O. Kinouchi, G. A. Cecchi, M. Copelli, S. Ribeiro, Speech graphs provide a quantitative measure of thought disorder in psychosis, PloS One 7 (4) (2012) e34928.

[23] N. B. Mota, M. Copelli, S. Ribeiro, Thought disorder measured as random speech structure classifies negative symptoms and schizophrenia diagnosis 6 months in advance, npj Schizophrenia 3 (1) (2017) 1–10.

[24] A. Stolcke, E. Shriberg, Statistical language modeling for speech disfluencies, in: 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings, Vol. 1, IEEE, 1996, pp. 405–408.

[25] C. Frankenberg, J. Weiner, T. Schultz, M. Knebel, C. Degen, H.-W. Wahl, J. Schroeder, Perplexity –a new predictor of cognitive changes in spoken language?– results of the Interdisciplinary Longitudinal Study on Adult Development and Aging (ILSE), Linguistics Vanguard 5 (2) (2019) 1–10.

[26] J. T. Becker, F. Boiler, O. L. Lopez, J. Saxton, K. L. McGonigle, The natural history of Alzheimer's disease: description of study cohort and accuracy of diagnosis, Archives of Neurology 51 (6) (1994) 585–594.

[27] J. Fritsch, S. Wankerl, E. Nöth, Automatic diagnosis of Alzheimer's disease using neural network language models, in: ICASSP 2019-2019 IEEE Inter-

national Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2019, pp. 5841–5845.

[28] T. Cohen, S. Pakhomov, A tale of two perplexities: Sensitivity of neural language models to lexical retrieval deficits in dementia of the Alzheimer's type, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 1946–1957. `doi:10.18653/v1/2020.acl-main.176`.
URL `https://aclanthology.org/2020.acl-main.176`

[29] Z. S. Harris, Distributional structure, Word 10 (2-3) (1954) 146–162.

[30] C. Manning, H. Schutze, Foundations of statistical natural language processing, MIT press, 1999.

[31] Y. Goldberg, Neural network methods for natural language processing, Synthesis Lectures on Human Language Technologies 10 (1) (2017) 1–309.

[32] W. A. Gale, K. W. Church, What's wrong with adding one, Corpus-based research into language: In honour of Jan Aarts (1994) 189–200.

[33] R. Kneser, H. Ney, Improved backing-off for m-gram language modeling, in: 1995 international conference on acoustics, speech, and signal processing, Vol. 1, IEEE, 1995, pp. 181–184.

[34] J. L. Elman, Finding structure in time, Cognitive science 14 (2) (1990) 179–211.

[35] Y. Bengio, P. Simard, P. Frasconi, Learning long-term dependencies with gradient descent is difficult, IEEE transactions on neural networks 5 (2) (1994) 157–166.

[36] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation 9 (8) (1997) 1735–1780.

[37] I. Sutskever, O. Vinyals, Q. V. Le, Sequence to sequence learning with neural networks, in: Advances in neural information processing systems, 2014, pp. 3104–3112.

[38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems, 2017, pp. 5998–6008.

[39] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, arXiv preprint arXiv:1409.0473.

[40] I. Tenney, D. Das, E. Pavlick, BERT rediscovers the classical NLP pipeline, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 4593–4601. `doi:10.18653/v1/P19-1452`.
URL `https://aclanthology.org/P19-1452`

[41] I. Tenney, P. Xia, B. Chen, A. Wang, A. Poliak, R. T. McCoy, N. Kim, B. Van Durme, S. R. Bowman, D. Das, E. Pavlick, What do you learn from context? probing for sentence structure in contextualized word representations, arXiv preprint arXiv:1905.06316.

[42] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, S. R. Bowman, Glue: A multi-task benchmark and analysis platform for natural language understanding, arXiv preprint arXiv:1804.07461.

[43] A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, S. R. Bowman, Superglue: A stickier benchmark for general-purpose language understanding systems, arXiv preprint arXiv:1905.00537.

[44] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805.

[45] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al., Language models are unsupervised multitask learners, OpenAI blog 1 (8) (2019) 9.

[46] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, R. Soricut, Albert: A lite BERT for self-supervised learning of language representations, arXiv preprint arXiv:1909.11942.

[47] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, arXiv preprint arXiv:1907.11692.

[48] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, Q. V. Le, XLNet: Generalized Autoregressive Pretraining for Language Understanding, in: H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, R. Garnett (Eds.), Advances in Neural Information Processing Systems, Curran Associates, Inc.

[49] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, arXiv preprint arXiv:1910.10683.

[50] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, arXiv preprint arXiv:2005.14165.

[51] P. J. Liu, M. Saleh, E. Pot, B. Goodrich, R. Sepassi, L. Kaiser, N. Shazeer, Generating Wikipedia by summarizing long sequences, arXiv preprint arXiv:1801.10198.

[52] P. E. Shrout, J. L. Fleiss, Intraclass correlations: uses in assessing rater reliability., Psychological bulletin 86 (2) (1979) 420.

[53] H. Goodglass, E. Kaplan, Boston diagnostic aphasia examination booklet, Lea & Febiger, 1983.

[54] D. Jurafsky, J. H. Martin, Speech and language processing. Vol. 3, Prentice Hall, 2014.

[55] K. W. Church, Z. Chen, Y. Ma, Emerging trends: A gentle introduction to fine-tuning, Natural Language Engineering 27 (6) (2021) 763–778.

[56] D. Liljequist, B. Elfving, K. Skavberg Roaldsen, Intraclass correlation–a discussion and demonstration of basic features, PloS one 14 (7) (2019) e0219854.

[57] B. MacWhinney, The CHILDES project: Tools for analyzing talk, Volume II: The database, Psychology Press, 2014.

[58] B. MacWhinney, Tools for analyzing talk part 1: The CHAT transcription format, Carnegie (2017) 1–115.

[59] J. R. Helms, Mathematics for Health Sciences: A Comprehensive Approach, Cengage Learning, 2009.

[60] M. Aldinucci, S. Bagnasco, S. Lusso, P. Pasteris, S. Rabellino, S. Vallero, OCCAM: a flexible, multi-purpose and extendable HPC cluster, Journal of Physics: Conference Series 898 (8) (2017) 082039.

[61] F. J. Huff, J. Becker, S. Belle, R. Nebes, A. Holland, F. Boller, Cognitive deficits and clinical diagnosis of Alzheimer's disease, Neurology 37 (7) (1987) 1119–1119.

[62] O. L. Lopez, A. Swihart, J. T. Becker, O. Reinmuth, C. Reynolds, D. Rezek, F. Daly, Reliability of NINCDS-ADRDA clinical criteria for the diagnosis of Alzheimer's disease, Neurology 40 (10) (1990) 1517–1517.

Table A.6: Figures describing the transcripts employed in Experiment 2: time duration, number of tokens, number of unique tokens (along with average number of tokens and average number of unique tokens) and type-token ratio (TTR) are reported for each such speech transcript.

Subject	Transcript	Duration	Tokens	Unique Tokens	AVG Tokens	AVG Unique Tokens	TTR
Joe Biden	I	0 : 32 : 23	4,647	1,074	6,315	1,343	0.21
	II	0 : 41 : 39	5,446	1,140			
	III	0 : 25 : 00	9,490	1,895			
	IV	0 : 43 : 36	6,801	1,381			
	V	0 : 34 : 05	5,211	1,226			
Donald Trump	I	1 : 17 : 37	15,200	1,967	15,051	1,185	0.13
	II	0 : 56 : 17	10,501	1,614			
	III	1 : 43 : 43	20,865	2,300			
	IV	1 : 13 : 01	14,056	1,945			
	V	1 : 18 : 19	14,806	1,896			
Barack Obama	I	0 : 56 : 39	5,594	1,479	5,957	1,271	0.21
	II	0 : 38 : 15	6,298	1,252			
	III	0 : 38 : 45	5,526	1,153			
	IV	0 : 45 : 55	6,981	1,312			
	V	0 : 36 : 07	5,390	1,159			
Bernie Sanders	I	0 : 35 : 33	4,164	969	4,458	1,046	0.23
	II	0 : 29 : 51	3,785	849			
	III	0 : 34 : 54	4,451	1,088			
	IV	0 : 43 : 27	5,387	1,039			
	V	0 : 44 : 46	4,501	1,286			
Bill Gates	I	0 : 35 : 53	3,503	944	2,514	812	0.32
	II	0 : 17 : 20	1,679	577			
	III	0 : 24 : 07	2,350	779			
	IV	0 : 22 : 04	2,152	744			
	V	0 : 30 : 07	2,896	1,018			
Nelson Mandela	I	0 : 40 : 17	3,844	1,113	6,403	1,410	0.22
	II	0 : 29 : 45	1,740	617			
	III	3 : 00 : 00	15,682	2,702			
	IV	1 : 43 : 21	7,741	1,654			
	V	0 : 40 : 16	3,020	963			
Martin Luther King	I	0 : 42 : 51	5,197	1,102	6,508	1,379	0.21
	II	0 : 46 : 56	6,471	1,315			
	III	0 : 43 : 48	6,287	1,456			
	IV	0 : 40 : 38	8,256	1,697			
	V	0 : 47 : 54	6,332	1,324			
Boris Johnson	I	0 : 51 : 42	4,397	1,123	3,202	943	0.29
	II	0 : 20 : 35	2,758	764			
	III	0 : 17 : 47	1,960	659			
	IV	0 : 17 : 00	2,375	896			
	V	0 : 38 : 22	4,530	1,273			

Table A.7: Detailed results obtained in Experiment 2 (between subjects reliability): each sub-table reports results for N-gram models. For each experiment we report perplexity scores along with their standard deviations. More specifically, we report the scores obtained by employing 2-grams to 5-grams models. Each row reports the scores obtained through the LM trained on speeches by the subject in the first column and tested on the other speakers.

2-grams

Subject	J. Biden		D. Trump		B. Obama		B. Sanders		B. Gates		N. Mandela		M. L. King		B. Johnson	
	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev
J. Biden	—	—	262.42	4.03	223.76	6.94	199.50	5.86	209.92	17.74	181.64	16.15	191.22	13.16	197.47	26.94
D. Trump	276.71	18.05	—	—	268.72	7.02	230.47	4.43	241.55	23.36	205.11	21.21	219.05	19.62	225.97	36.59
B. Obama	242.32	14.15	281.05	3.99	—	—	202.07	5.96	219.87	17.27	188.89	18.43	198.19	16.07	205.93	29.94
B. Sanders	202.26	9.93	231.57	4.33	197.19	5.50	—	—	191.61	10.02	166.92	14.74	172.84	11.60	179.69	18.10
B. Gates	221.97	11.75	253.93	3.69	215.63	5.47	197.46	3.48	—	—	177.97	16.40	186.61	12.56	194.60	24.05
N. Mandela	215.75	14.05	245.86	5.17	200.55	8.16	192.64	7.10	199.97	7.89	—	—	183.76	16.17	190.32	24.37
M. L. King	232.65	12.95	267.86	3.33	225.78	5.87	199.83	6.48	215.69	14.60	187.93	18.35	—	—	201.88	26.07
B. Johnson	222.69	12.78	253.65	3.46	215.32	7.38	197.24	6.29	207.91	13.94	179.03	17.05	187.25	12.98	—	—

3-grams

Subject	J. Biden		D. Trump		B. Obama		B. Sanders		B. Gates		N. Mandela		M. L. King		B. Johnson	
	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev
J. Biden	—	—	406.05	8.04	335.23	22.01	262.33	8.07	263.84	32.53	210.22	17.30	247.03	37.22	244.37	37.09
D. Trump	454.56	40.38	—	—	461.33	37.33	332.98	8.51	329.39	50.45	250.27	25.36	310.56	63.55	301.00	54.22
B. Obama	358.21	27.74	445.25	9.07	—	—	266.61	14.83	278.93	33.74	217.98	19.41	257.84	44.90	256.96	40.79
B. Sanders	271.69	16.91	323.58	5.99	273.11	14.96	—	—	228.20	19.08	185.62	14.71	209.38	26.40	212.36	24.03
B. Gates	310.96	21.20	377.88	6.84	316.11	19.48	252.86	6.72	—	—	202.51	16.94	235.63	32.85	236.38	32.09
N. Mandela	279.08	21.42	330.54	7.52	265.66	10.72	233.46	9.95	235.10	15.99	—	—	219.70	29.87	220.96	30.05
M. L. King	333.91	24.66	410.89	6.77	339.45	23.40	257.78	13.92	269.14	28.06	215.84	19.19	—	—	248.40	35.49
B. Johnson	301.58	20.95	357.49	5.50	301.67	15.95	247.23	6.66	250.34	25.24	201.36	17.66	230.76	28.22	—	—

4-grams

Subject	J. Biden		D. Trump		B. Obama		B. Sanders		B. Gates		N. Mandela		M. L. King		B. Johnson	
	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev
J. Biden	—	—	511.58	10.26	408.86	29.02	313.72	8.29	308.91	44.85	239.39	17.26	294.40	49.96	283.25	47.23
D. Trump	595.73	59.19	—	—	601.17	52.71	425.11	11.01	407.96	74.91	297.95	28.24	393.33	91.56	366.84	72.73
B. Obama	434.45	36.32	553.74	10.74	—	—	314.58	19.09	325.33	46.01	247.72	19.14	305.16	58.41	297.27	51.33
B. Sanders	316.38	20.87	385.39	6.63	318.39	18.59	—	—	257.53	25.42	205.20	13.84	239.21	32.71	238.26	30.06
B. Gates	371.64	27.30	463.54	8.60	378.35	24.81	296.05	7.08	—	—	227.73	16.36	275.74	41.42	270.49	40.51
N. Mandela	319.85	24.56	387.17	8.18	308.12	13.59	264.24	10.35	263.71	21.80	—	—	249.01	34.04	245.99	35.71
M. L. King	403.33	32.13	509.82	8.58	409.59	29.81	302.47	17.95	312.81	39.24	244.41	18.76	—	—	286.18	45.07
B. Johnson	357.56	26.52	435.98	7.46	359.13	20.68	287.57	7.20	286.85	33.98	224.76	17.14	268.16	35.35	—	—

5-grams

Subject	J. Biden		D. Trump		B. Obama		B. Sanders		B. Gates		N. Mandela		M. L. King		B. Johnson	
	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev
J. Biden	—	—	624.81	13.72	491.81	35.69	369.76	9.27	360.01	58.55	273.18	18.93	346.38	59.97	327.28	58.93
D. Trump	750.16	76.77	—	—	765.22	68.36	529.26	14.38	499.48	103.94	354.69	34.56	486.85	115.52	444.11	95.07
B. Obama	516.13	44.54	671.90	14.05	—	—	367.86	24.24	378.35	59.87	282.25	20.26	357.79	69.76	343.16	63.57
B. Sanders	364.42	25.11	452.64	8.10	369.41	22.24	—	—	290.97	32.57	227.70	13.57	272.37	38.10	267.51	37.06
B. Gates	438.15	33.74	558.68	11.50	449.35	30.61	344.39	7.59	—	—	256.98	16.87	320.80	49.22	309.31	50.31
N. Mandela	367.01	28.55	452.77	9.31	357.29	17.00	299.51	10.97	296.75	28.68	—	—	282.82	39.02	274.89	42.54
M. L. King	478.32	39.58	618.46	11.63	489.60	36.37	352.03	23.17	362.62	51.94	277.48	19.59	—	—	329.21	56.26
B. Johnson	418.83	32.46	523.31	10.17	424.61	25.77	332.46	7.98	328.72	43.92	252.02	17.48	310.22	41.90	—	—

Table A.8: Detailed results obtained in Experiment 2 (between subjects reliability): each subtable reports results for a GPT-2-based language model (differences stem from the number of fine tuning epochs employed to acquire each such model). For each experiment we report perplexity scores along with their standard deviations. Each row reports the scores obtained through the LM trained on speeches by the subject in the first column and tested on the other speakers.

GPT-2 **5 epochs**

Subject	J. Biden		D. Trump		B. Obama		B. Sanders		B. Gates		N. Mandela		M. L. King		B. Johnson	
	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev
J. Biden	—	—	22.48	0.90	21.28	1.94	26.32	3.74	18.39	2.58	29.23	2.41	28.17	3.05	42.49	7.85
D. Trump	23.27	1.72	—	—	20.42	2.37	26.39	3.84	17.96	2.24	31.01	2.43	28.48	3.15	42.93	8.04
B. Obama	24.61	1.79	22.58	0.98	—	—	26.37	3.75	18.28	2.56	29.39	2.36	28.10	3.07	43.03	7.87
B. Sanders	27.47	1.84	25.60	1.16	23.40	1.83	—	—	19.92	2.83	29.61	2.51	29.54	3.32	46.23	8.42
B. Gates	25.16	1.73	23.26	0.95	21.82	1.96	26.84	3.83	—	—	28.71	2.32	27.96	2.92	43.30	7.93
N. Mandela	28.54	1.97	26.38	1.25	24.45	1.89	28.62	3.84	19.63	2.51	—	—	28.63	3.12	45.80	8.45
M. L. King	25.39	1.76	23.51	1.09	21.68	1.76	26.21	3.63	18.45	2.46	28.03	2.50	—	—	42.86	7.86
B. Johnson	26.20	1.79	24.75	0.99	22.76	1.90	27.30	3.80	19.34	2.84	28.93	2.40	28.62	3.01	—	—

GPT-2 **10 epochs**

Subject	J. Biden		D. Trump		B. Obama		B. Sanders		B. Gates		N. Mandela		M. L. King		B. Johnson	
	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev
J. Biden	—	—	21.24	0.87	20.15	1.94	25.72	3.84	17.84	2.51	30.23	2.35	28.11	3.18	41.71	7.89
D. Trump	23.46	1.88	—	—	20.49	2.79	28.04	4.17	18.33	2.26	34.65	2.64	30.39	3.74	45.40	8.72
B. Obama	23.60	1.89	21.41	0.97	—	—	26.10	3.92	17.84	2.49	30.54	2.32	28.41	3.29	43.01	8.01
B. Sanders	26.06	1.65	24.30	1.11	22.39	1.68	—	—	19.27	2.66	29.25	2.38	28.66	3.30	44.52	8.13
B. Gates	24.34	1.67	22.31	0.96	20.99	1.94	26.41	3.94	—	—	29.53	2.21	28.04	3.07	43.13	8.07
N. Mandela	28.78	1.99	26.37	1.31	24.54	1.87	28.60	3.98	19.65	2.40	—	—	28.74	3.09	46.15	8.66
M. L. King	25.22	1.78	22.88	1.19	21.33	1.70	26.13	3.66	18.27	2.32	28.54	2.54	—	—	43.32	8.20
B. Johnson	25.03	1.61	24.00	0.88	21.89	1.77	26.54	3.77	18.85	2.87	28.91	2.33	27.93	2.90	—	—

GPT-2 **20 epochs**

Subject	J. Biden		D. Trump		B. Obama		B. Sanders		B. Gates		N. Mandela		M. L. King		B. Johnson	
	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev
J. Biden	—	—	20.60	0.94	19.69	2.11	26.36	4.10	17.79	2.42	32.29	2.42	29.16	3.58	42.70	8.33
D. Trump	25.38	2.42	—	—	21.91	3.62	32.55	4.86	20.23	2.91	43.10	2.76	35.18	5.08	52.92	10.60
B. Obama	24.10	2.24	21.44	1.07	—	—	27.46	4.25	18.25	2.51	33.17	2.50	30.18	3.81	45.92	8.77
B. Sanders	26.19	1.59	24.29	1.18	22.30	1.46	—	—	19.36	2.58	30.44	2.40	29.34	3.59	45.62	8.44
B. Gates	24.90	1.92	22.21	1.08	21.18	2.16	27.42	4.19	—	—	31.54	2.13	29.53	3.57	45.88	8.75
N. Mandela	29.64	2.11	26.70	1.40	25.15	1.99	29.26	4.18	20.06	2.31	—	—	29.64	3.21	48.01	9.48
M. L. King	27.10	2.09	23.72	1.48	22.55	1.90	28.16	3.88	19.34	2.28	31.31	2.83	—	—	48.12	9.61
B. Johnson	24.68	1.59	24.20	0.82	21.69	1.75	26.68	4.02	18.92	3.16	29.87	2.34	28.31	3.02	—	—

GPT-2 **30 epochs**

Subject	J. Biden		D. Trump		B. Obama		B. Sanders		B. Gates		N. Mandela		M. L. King		B. Johnson	
	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev	PPL	stdev
J. Biden	—	—	21.18	1.03	20.14	2.36	27.98	4.49	18.42	2.53	35.04	2.59	31.11	4.09	45.47	9.18
D. Trump	28.07	3.07	—	—	24.07	4.63	37.73	5.76	22.67	3.76	52.84	3.02	41.10	6.74	62.44	13.05
B. Obama	26.10	2.71	23.14	1.24	—	—	30.12	4.73	19.69	2.78	37.07	2.85	33.41	4.54	51.18	9.91
B. Sanders	27.16	1.73	24.88	1.32	22.91	1.44	—	—	19.93	2.55	31.91	2.48	30.70	3.95	48.29	9.21
B. Gates	26.83	2.35	23.47	1.31	22.57	2.55	29.68	4.58	—	—	34.83	2.11	32.38	4.28	51.16	9.75
N. Mandela	31.40	2.33	27.94	1.53	26.47	2.15	30.84	4.47	21.01	2.43	—	—	31.60	3.59	51.74	10.70
M. L. King	30.26	2.50	25.79	1.82	24.78	2.25	31.23	4.33	21.32	2.51	36.00	3.20	—	—	55.62	11.67
B. Johnson	25.43	1.71	25.17	0.82	22.33	1.79	27.75	4.26	19.60	3.50	31.33	2.51	29.53	3.20	—	—

Table A.9: Detailed results for Experiment 3. The table reports Accuracy (Acc.) scores, Precision (P), Recall (R) and F1 for both tasks of identifying AD and Control subjects. The rightmost column reports the harmonic mean (HM) of the accuracy, F1 score on the AD and C classes. Best results are marked in boldface.

Model		Acc.	Dementia (AD)			Control (C)			HM(acc,F1$_{AD}$,F1$_{C}$)
			P	R	F1	P	R	F1	
2-grams	$\overline{P}_C$	0.44	0.41	0.21	0.28	0.46	0.69	0.55	0.39
	$\overline{P}_{AD}$	0.55	0.54	0.75	0.63	0.57	0.34	0.42	0.52
	$\overline{D}$	0.67	0.68	0.66	0.67	0.66	0.68	0.67	0.67
	$\overline{D}^*$	**0.93**	0.99	0.87	**0.92**	0.88	0.99	**0.93**	**0.93**
3-grams	$\overline{P}_C$	0.43	0.40	0.22	0.28	0.44	0.65	0.53	0.39
	$\overline{P}_{AD}$	0.56	0.55	0.70	0.62	0.57	0.41	0.47	0.54
	$\overline{D}$	0.74	0.76	0.71	0.74	0.72	0.77	0.75	0.74
	$\overline{D}^*$	**0.91**	1.00	0.83	**0.91**	0.85	1.00	**0.92**	**0.91**
4-grams	$\overline{P}_C$	0.42	0.38	0.23	0.29	0.43	0.61	0.51	0.38
	$\overline{P}_{AD}$	0.54	0.54	0.65	0.59	0.54	0.43	0.48	0.53
	$\overline{D}$	0.76	0.81	0.70	0.75	0.73	0.82	0.77	0.76
	$\overline{D}^*$	**0.89**	1.00	0.78	**0.88**	0.81	1.00	**0.90**	**0.89**
5-grams	$\overline{P}_C$	0.42	0.38	0.23	0.29	0.43	0.61	0.51	0.38
	$\overline{P}_{AD}$	0.52	0.53	0.62	0.57	0.52	0.42	0.46	0.52
	$\overline{D}$	0.77	0.86	0.66	0.75	0.72	0.89	0.80	0.77
	$\overline{D}^*$	**0.89**	1.00	0.79	**0.88**	0.82	1.00	**0.90**	**0.89**
GPT-2 5 epochs	$\overline{P}_C$	0.65	0.64	0.70	0.67	0.66	0.59	0.62	0.65
	$\overline{P}_{AD}$	0.38	0.42	0.58	0.49	0.29	0.18	0.22	0.33
	$\overline{D}$	**0.71**	0.76	0.62	**0.69**	0.67	0.80	**0.73**	**0.71**
	$\overline{D}^*$	0.49	0.50	0.09	0.15	0.49	0.91	0.64	0.30
GPT-2 10 epochs	$\overline{P}_C$	0.78	0.70	0.99	0.82	0.98	0.57	0.72	0.77
	$\overline{P}_{AD}$	0.62	0.63	0.58	0.61	0.60	0.65	0.62	0.62
	$\overline{D}$	**1.00**	1.00	1.00	**1.00**	1.00	1.00	**1.00**	**1.00**
	$\overline{D}^*$	**1.00**	1.00	1.00	**1.00**	1.00	1.00	**1.00**	**1.00**
GPT-2 20 epochs	$\overline{P}_C$	0.81	0.73	1.00	0.84	1.00	0.61	0.76	0.80
	$\overline{P}_{AD}$	0.78	0.91	0.64	0.75	0.71	0.93	0.81	0.78
	$\overline{D}$	**1.00**	1.00	1.00	**1.00**	1.00	1.00	**1.00**	**1.00**
	$\overline{D}^*$	**1.00**	1.00	1.00	**1.00**	1.00	1.00	**1.00**	**1.00**
GPT-2 30 epochs	$\overline{P}_C$	0.81	0.73	1.00	0.84	1.00	0.61	0.76	0.80
	$\overline{P}_{AD}$	0.81	0.96	0.65	0.78	0.73	0.97	0.83	0.80
	$\overline{D}$	**1.00**	1.00	1.00	**1.00**	1.00	1.00	**1.00**	**1.00**
	$\overline{D}^*$	**1.00**	1.00	1.00	**1.00**	1.00	1.00	**1.00**	**1.00**