Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 298–306
Marseille, 11–16 May 2020

Effort Estimation in Named Entity Tagging Tasks

Inês Gomes, Rui Correia, Jorge Ribeiro, João Freitas
DefinedCrowd Corporation
{ines.gomes, correia, jorge, joao}@definedcrowd.com

Abstract

Named Entity Recognition (NER) is an essential component of many Natural Language Processing pipelines. However, building these language dependent models requires large amounts of annotated data. Crowdsourcing emerged as a scalable solution to collect and enrich data in a more time-efficient manner. To manage these annotations at scale, it is important to predict completion timelines and compute fair pricing for workers in advance. To achieve these goals, we need to know how much effort will be taken to complete each task. In this paper, we investigate which variables influence the time spent on a named entity annotation task by a human. Our results are two-fold: first, the understanding of the effort-impacting factors which we divided into cognitive load and input length; and second, the performance of the prediction itself. On the latter, through model adaptation and feature engineering, we attained a Root Mean Squared Error (RMSE) of 25.68 words per minute with a Nearest Neighbors model.

Keywords: Named Entity Tagging, Human-in-the-loop, Human Effort

1. Introduction

Named Entity Recognition (NER) is a task in the field of Information Extraction that consists of identifying and classifying specific information elements commonly referred to as *named entities* (Marrero et al., 2013). NER is a fundamental component to many Natural Language Processing (NLP) pipelines (Voyer et al., 2010), including the creation and categorization of language resources. However, automatic recognition of entities often needs a large amount of labelled training data and, despite some large dataset being publicly available, they are often associated with specific domains, such as microblogs, news articles or scientific publications (Feyisetan et al., 2015; Lu et al., 2019).

The process of creating training corpora for NER in new domains (or for new languages) is expensive, considering both time and monetary costs (Marrero et al., 2013). Human-in-the-Loop platforms address this scalability problem by assigning Human Intelligence Tasks (HITs) to an existing pool of non-expert contributors. Such contributors are known to be able to approximate expert judgements, after leveraging their answers with other contributors' responses to the same piece of information (Finin et al., 2010; Yin et al., 2014). This information settled on the HIT is completed within a short time, in exchange for a monetary reward (Hassan and Curry, 2013). A group of HITs that shares the same formulation (e.g. for which only the text to be tagged varies) is commonly referred to as *job* (Hirth et al., 2011).

In this paper, we address the problem of predicting the time each HIT takes to be completed in Named Entity Tagging (NET) tasks before the actual annotation takes place. In NET jobs, the contributor's goal is to locate and annotate pre-defined named entities in unstructured text.

From the Human-in-the-Loop platform point-of-view, estimating the human effort of NET tasks has several applications, including:

- Estimating the completion time of a NET job: by combining human effort information with contributor throughput prediction, one can estimate a deadline for the completion of a job (Sautter and Böhm, 2013);
- Awarding a fair payment to contributors: the monetary reward can be adjusted to the expected time-on-task, making micro-payments a more transparent and fair process (Lofi et al., 2012);
- Detecting fraudulent contributors: estimating the time needed by a human for completing a given task establishes a baseline for detecting outlying behaviour, which serves as a data quality indicator (Hirth et al., 2014).

Using a data-driven approach, we address the problem of human effort prediction by answering two research questions. First, *which factors affect human effort for NET tasks?*, and second, *which strategies can be used to predict human effort*? Our major contributions include a study on the variables that influence the time spent on a named entity annotation task by a human, and a strategy to estimate effort for NET tasks in crowdsourcing.

The remainder of this paper is structured as follows: Section 2 describes related work in the area of human effort estimation in general, and NET in particular; Section 3 presents the dataset used in this study; Section 4 explains the experimental setup that includes the target variable definition and data preparation, the evaluation metrics used, and the feature engineering process; Sections 5 and 6 report the experiments carried out at the HIT and job level, respectively, and the corresponding results; and, finally, Sections 7 and 8 present the discussion of the results and the conclusions.

2. Related Work

The concept of *human effort* (or *crowd effort*) is broadly used in the crowdsourcing-related literature. Eickhoff and de Vries (2013), for instance, have used a measure of effort as an indicator for cheater detection: contributors that are trying to take advantage of the system by submitting suboptimal work show outlying effort rates. Another example is the work of Jain et al. (2017), who have used effort as a performance metric for studying the effectiveness of the tasks

themselves, concluding how certain User Interface (UI) elements improve user experience, delivery time and quality. Effort is defined as the amount of time taken to complete a HIT. More concretely, it is the interval between the contributor being shown the HIT and submitting it. Hirth et al. (2014) argue that effort can be divided into two dimensions:

- *Reading time*: the time it will take a human to read the input to be processed;
- *Answering time*: the time involved with decision making and interaction with the UI (according to the author, around five seconds per answer).

With respect to the first dimension, *reading time*, there is a large body of work related to reading speed both in the fields of Human Computation and Psychology. Rayner et al. (2016) state that "college-educated adults" read at a rate of 200 to 400 wpm. Dyson (2001) have studied the influence of reading speed on comprehension, concluding that fast and slow readers have similar comprehension accuracy. The author also concludes that surface memory, i.e., the recognition of specific wording of statements, is more accurate at regular reading rates. Allen et al. (2014) state that comprehension relies not only on the readers' background knowledge, but also on the cognitive processes necessary to capitalize on the existing knowledge. At fast speeds, the authors found better comprehension amongst readers who pause more often or use more scrolling movements. However, while facing longer segments of text, cues to the location are lost when text is scrolled within a window.
With respect to NET tasks, it is also important to consider the process of *skimming*, that is to quickly browse through the text to find a specific piece of information. Rayner et al. (2016) argue that skimming rates can be as much as two to four times faster than those of typical silent reading. The authors also have studied the patterns of word and character recognition, concluding that, for common words, all the characters are recognized simultaneously. On the other hand, unknown words or uncommon words with more than seven characters require multiple fixations and consequently are less efficient to read.
Regarding the second dimension pointed out in Hirth et al. (2014), *answering time*, we also take into account the cognitive load involved in NET tasks. It is known that high cognitive demand leads to worse performance (Finnerty et al., 2013). From the Cognitive theory, Sweller and Chandler (1994) describe cognitive load as having two sources:

- inherent complexity: temporal demands that contributors need to put into the task completion, reflected by the number of available task elements (e.g. text, images, links, entities) (Yang et al., 2016);
- organization and clarity of the content: elements such as task title, instructions, description and keywords, which have direct impact on how clear a given task is perceived to be (Martin et al., 2017).

On considerations regarding *answering time* related to NET, Feyisetan et al. (2017) have studied how specific features of HITs affect the accuracy and speed of entity annotation. Particularly, the authors focus on the size of the taxonomy (i.e. the number of named entities), the entities' semantics (e.g. person, location) and the input length (text to be tagged). Running an experiment with 7.5K tweets, each annotated by three different contributors, the authors highlight five points:

- contributors are more accurate when having few named entities;
- tasks with fewer named entities are more likely to be selected;
- there is no strong connection between input length and annotation accuracy;
- different categories of named entities have different accuracy and annotation times (categories like *miscellaneous*" tend to take longer to annotate);
- clean, clearly described and properly capitalized text contributes to accuracy.

A final aspect to consider when assessing human effort is motivation. Rogstadius et al. (2011) conclude that contributors' performance and effort are affected by varying the levels of intrinsic motivation (enjoyment, personal improvement or preference) and the extrinsic counterpart (payment, social factors or requirements). The authors find that, while intrinsic motivation do not impact the project completion times, it has a strong positive effect on contributor accuracy. On the other front, the authors show that extrinsic motivation leads to quicker results but not to higher levels of accuracy. These observations were corroborated by Mason and Watts (2010). Sautter and Böhm (2013), however, highlight the fact that increasing the monetary reward may increase the rate of cheating contributors entering the tasks.

3. Dataset

To support this research, we accessed data from 15 Named Entity Tagging jobs in English which ran on *Neevo*[1] (*DefinedCrowd*'s[2] proprietary human-in-the-loop platform) between January and August 2019.
Figure 1 shows an example of a Named Entity Tagging HIT being completed by a contributor. The HIT is composed by a prompt with the text to tag (input) and an ontology composed by multiple categories of named entities. The contributor will iteratively highlight a segment of text with a named entity and select the corresponding category. When the HIT is complete, the contributor clicks the "NEXT" button being then forwarded to the next HIT in the pool.
The dataset is composed of 167,609 unique HITs. Following crowdsourcing best practices (Baba and Kashima, 2013), each HIT was executed by, at least, three different contributors, resulting in a total set of 505,295 answers, completed by 1,489 distinct contributors. All executions are enriched with time-on-task information. The NET taxonomy, i.e., the set of named entities categories, is established at the job level and varies between one and ten distinct categories. The number of contributors and HITs to

[1] https://www.neevo.ai/
[2] https://www.definedcrowd.com/

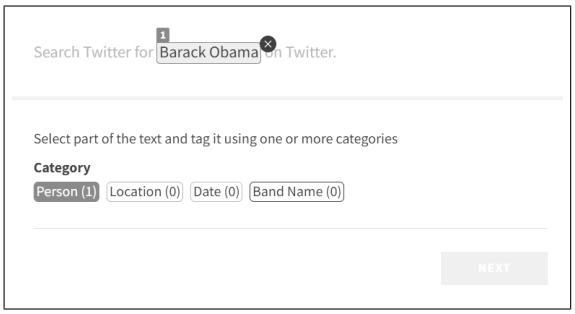


Figure 1: Example of a Named Entity Tagging HIT in *Neevo*.

complete per job varies greatly, with the shortest job having 210 HITs executed by 6 contributors, and the largest having 52,280 HITs executed by 454 distinct contributors. Figure 2 shows the distribution of the number of HITs executed per contributor per job, confined to under 100 executions. Around 40% of the contributors execute less than 15 HITs per job and, although not shown in the chart, 30% execute more than 100 HITs. It is worth highlighting that this discrepancy may affect the average contributor time-on-task, as contributors may have different levels of engagement and expertise.

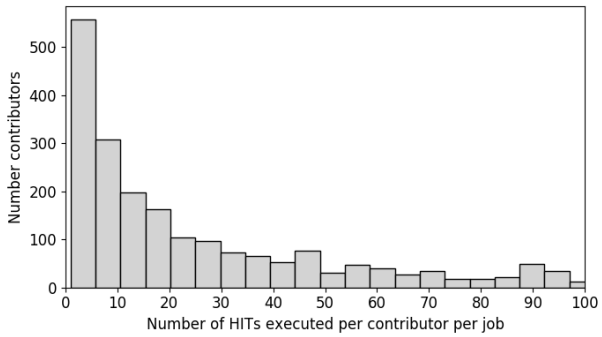


Figure 2: Distribution of number of HITs executed per contributor per job, up to 100 HITs for visualization purposes.

In Figure 3 we observe the distribution of the input length in number of tokens over HITs. It is possible to identify two main groups of tasks: one with text length around 15 tokens and the other around 65. Additionally, we observe that 75% of the HITs in the dataset have less than 50 words. Figure 4 shows the distribution of the time needed to complete a HIT, bounded to executions of under two minutes. This visualization represents 93.4% of the dataset. The median of time-on-task is 22 seconds.

The distributions presented in the figures above are the first step to understand the dataset, define the target variable and the next section experiments.

4. Experimental Setup

As mentioned in the introductory section, we aim to understand which factors impact the human effort when annotating named entities and how accurate can we be when trying to predict such effort before the actual annotation takes

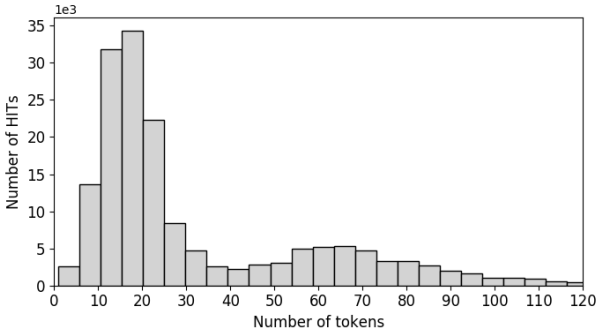


Figure 3: Input length distribution in number of tokens.

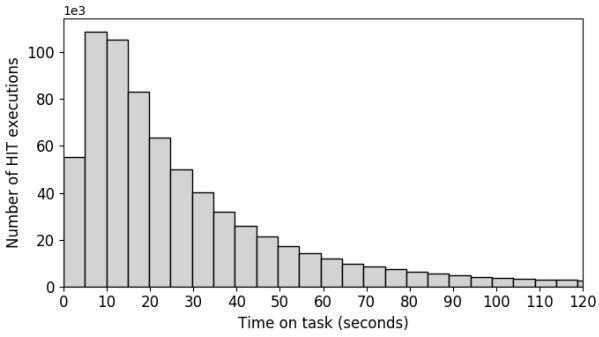


Figure 4: Distribution of time-on-task over HIT executions, confined to under two minutes for visualization purposes.

place. When executing a HIT, external factors including contributor carelessness (such as leaving the HIT open for an unreasonable amount of time) or contributor expertise (where an experienced contributor completes tasks at a very fast pace) may impact the completion time. Given the goal of the present research, these are not desirable data points to train (and test) an effort predictor, so we must prepare the dataset accordingly. This section describes the experimental setup, taking into consideration the data distributions described in Section 3.

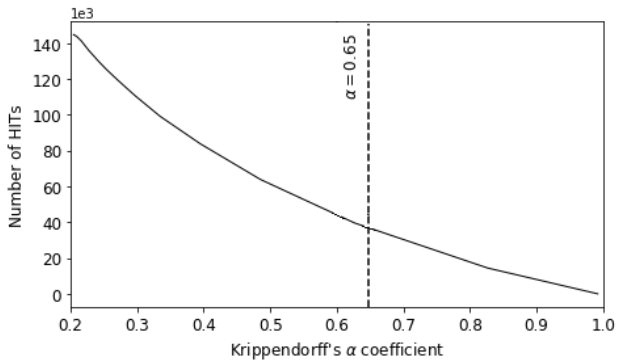
4.1. Target Variable

As described in Section 2, human effort represents the time needed by a human for completing a given HIT. For representability and comprehension purposes, and following the approaches found on the literature (Feyisetan et al., 2017), we normalize time-on-task by the number of tokens present in the input. This transformation results in speed-on-task, which is measured in words per minute (wpm) and given by:

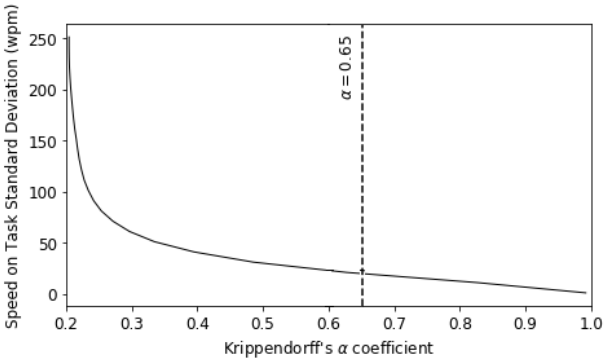
$$Speed - on - task = \frac{NumberTokens}{Time - on - Task}(wpm) \quad (1)$$

The average speed-on-task is an effort normalization, calculated per each HIT that has a redundancy of at least three executions. The average speed-on-task is our ultimate variable to predict.

Given the previously referred external factors that affect the HITs execution, we pre-process the dataset to discard



(a) Dataset size versus dataset agreement.



(b) Speed-on-task standard deviation versus dataset agreement.

Figure 5: Krippendorff's alpha and dataset size evolution given the allowed speed-on-task standard deviation threshold.

unreasonable speeds. For this, we investigate the speed-on-task's standard deviation (SD) for each HIT: a value of zero represents perfect alignment, i.e., all contributors performed the task at the exact same speed (optimal condition); larger values for the standard deviation (see Figure 5b for detailed information) imply discrepant speeds and, therefore, are not good candidates to be used for training nor testing the prediction of human effort.

To make the process of data curation systematic, we use the concept of Inter-Annotator Agreement (IAA) applied to the speed-on-task values, measuring the Krippendorff's alpha (α) coefficient of the dataset (Krippendorff, 1980). A value of one for the α coefficient means that there is total agreement, while a value of zero represents the agreement that can be attributed to chance. Landis and Koch (1977) defined IAA $\alpha \in [0.6, 0.8]$ as "substantial agreement" and such range is consistently used in the literature for data quality (Nowak and Rüger, 2010).

Figure 6 describes the iterative data curation procedure carried out. In sum, starting from the highest standard deviation observed for a HIT in the dataset, and in steps of -10, we remove all HITs below that threshold and compute the IAA of the remaining dataset. This process is repeated until reaching a defined threshold of $\alpha = 0.65$. This threshold takes into consideration both the state-of-the-art recommendations and our sense of the data.

Speed-on-task SD threshold = MAX_SD_OBSERVED
While speed-on-task SD threshold > 0
    Get dataset whose HITs speed-on-task SD is lower than the established threshold.
    Compute new dataset agreement using Krippendorff's alpha
    Reduce the speed-on-task SD threshold in 10 wpm
End while

Figure 6: Pseudo code for data curation process.

Figure 5 plots the tradeoffs between the value of IAA with respect to both number of HITs remaining in the dataset (5a) and standard deviation of speed-on-task (5b). The dotted vertical line on both plots represents the point where $\alpha = 0.65$. Through the procedure described in Figure 6, the first setting where the ideal condition is met ($\alpha > 0.65$), corresponds to accepting HITs for which their speed-on-task standard deviation is below 20 wpm, resulting on a total dataset size of 37,061 HITs.

Figure 7 shows the speed-on-task distribution considering the curated dataset. Rayner et al. (2016) states that an average reader reads between 200 wpm and 400 wpm. In our dataset, an average HIT executed by a contributor has average speed-on-task of 53.8 wpm. The average speed-on-task is lower, as expected, since it includes not only the task reading time, but also the task comprehension and entity tagging times.

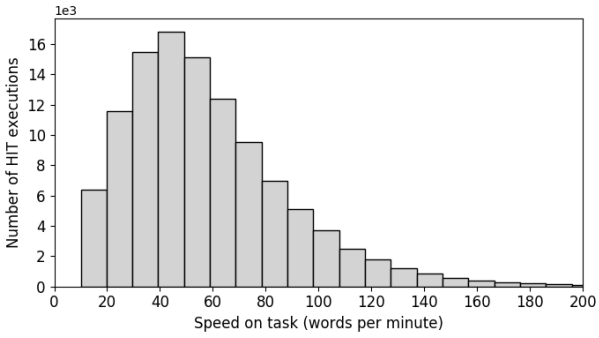


Figure 7: Speed-on-task distribution over HIT executions after dataset preparation.

4.2. Evaluation Metrics

Regarding the performance metrics, we report Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE). MAE measures the average of the absolute difference between each true and predicted values (see Equation 2). It weights equally small and large errors. RMSE measures the square root of the average of the squared difference between the predicted and the true values (see Equation 3). By definition, this performance metric gives more weight to larger errors. As a result, we consider RMSE to be more appropriate for our goal. All performance metrics are expressed in the same units of the target variable and, as they measure error, lower values represent higher performance

rates.

$$MAE = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y_j}| \quad (2)$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y_j})^2} \quad (3)$$

4.3. Feature Engineering

The last step of our data preparation pipeline was to engineer new features motivated by the literature review presented in Section 2. As described, these are expected to carry information that can help to predict effort.
The extracted features are:

- *Number of categories of named entities*: the size of the named entity taxonomy involved in the job;
- *Input length in number of tokens*: the length of the text to be tagged;
- *Number of sentences*: the number of sentences along with the input length are an indicator of readability (Kincaid et al., 1975), that may impact both reading and answering time;
- *Number of punctuation tokens*: punctuation influences speed-on-task as it affects reading speed. An unambiguous sentence will generally be read faster when it contains internal punctuation (Hirotani et al., 2006);
- *Number of stopwords*: according to the state-of-the-art, non common words or words unknown to the contributor make require more effort to interpret. As a result, we use *stopwords*[3], i.e. words that are most common in a language, to study the hypothesis that the number of stopwords in a given text influences the reading speed. We expect that the higher the number of stopwords, the lower the cognitive effort, resulting in a higher speed-on-task (Barrouillet et al., 2007);
- *Average word length*: as the size of the words impacts reading efficiency (Rayner et al., 2016), we expect that texts with higher average word length are more difficult to read;
- *Average word length, without stopwords*: as the size and regularity of the words impacts reading efficiency, this feature excludes stopwords to compute the average word length.

After computing the Pearson correlation between the features with the target variable, we found that the number of sentences ($\rho = 0.27$), input length in number of tokens ($\rho = 0.27$), number of stopwords ($\rho = 0.30$) and number of punctuation tokens ($\rho = 0.24$) are the features which correlate the most with the target variable. The average word length and number of entities have negative correlations with the target variable, but positive correlations among them ($\rho = 0.29$).

[3]In this research we use `https://pythonspot.com/nltk-stop-words/` set of stopwords

5. Predicting human effort at the HIT level

Using the curated dataset, we can now move to investigate the performance of predicting the human effort at the HIT level. The experiments described in the remainder of this section follow a 10-fold cross validation process. We evaluate the model using the evaluation metrics described in Section 4.2 (MAE and RMSE), and for the solutions that require a validation set, we tune the models hyperparameters using a grid search approach.
In this section, we carry out experiments considering the established experimental setup (see Section 4). In Section 5.1, we set a baseline (Ordinary Least Squares) and compare it to three nonlinear models using two features represented in the state-of-the-art. In Section 5.2, we improve the nonlinear models by expanding the set of explanatory variables, to understand in more detail the impact of the cognitive load features on the effort prediction.

5.1. Baseline

As a first approach to predict human effort, we start by establishing a linear model baseline with two features. The features used represent the dimensions described in the state-of-the-art: *input length in number of tokens*, as a basic representation of the amount of information to be processed, and *number of categories of named entities*, as a representation of cognitive load. The linear model used herein was the Ordinary Least Squares (OLS) estimator.
Although we believe that the *input length in number of tokens* may evolve linearly with the effort needed to complete a task, we do not expect the taxonomy size to contribute linearly to the time contributors spend while annotating. Therefore, we additionally explore the impact of using nonlinear approaches to the experiment.
Table 1 shows this experiment results. For reference, we include the performance of always assigning the average speed-on-task computed in the entire dataset (*53.8 wpm*). The OLS Model scores 27.33 wpm RMSE, 5.7% greater than assigning the average value. Moreover, Random Forest yields the best results in both evaluation metrics with 26.16 wpm RMSE, 4.3% better than the linear model. The set of hyperparameters that optimize the Random Forest model are *maximum depth = 25*, *number of estimators = 700*, *minimum sample split = 200* and *minimum samples leaf = 2*.

Model	MAE (wpm)	RMSE (wpm)
Average Speed-on-task (*53.8 wpm*)	21.78	28.98
Linear Model (baseline)	20.09	27.33
Nearest Neighbours	19.10	26.39
Random Forest	**19.06**	**26.16**
Gradient Tree Boosting	19.11	26.39

Table 1: Performance of predicting human effort with Nonlinear Models compared with the Linear model Baseline.

Given the similarity among models' results, we performed the Wilcoxon signed rank test at the 0.01 level, to verify the statistical significance of the results attained (Wilcoxon,

1946). Comparing the Random Forest with Nearest Neighbors and Gradient Tree Boosting, the p-values were bellow the significance level proving that the models predictions are statistically significant.

5.2. Exploring Additional Features

The results of the experiments described in the previous section show that nonlinear models are more suitable to capture the human effort complexity. Building up on that end, in this experiment we expand the set of explanatory variables to understand if they can contribute to outperform the previous experiments and, consequently, better explain the target variable. Under those circumstances, we add text preprocessing features that could model the cognitive load associated with the task (see Section 4.3): *number of sentences*, *number of punctuation tokens*, *number of stopwords* and *average word length* (with and without stopwords). Later, we accomplish an ablation study to understand if a reduced set of features would improve the models accuracy. Table 2 shows the experiment results when expanding the set of features according to the recommendations studied in the state-of-the-art, and after reducing the set of features according to the feature ablation. We observe that both the Nearest Neighbors and Random Forest improve the performance by reducing the set of features. The former removes the *count of sentences* and the *average word length without stopwords*, while the latter removes the *count of punctuation tokens* and the *input length in number of words*.

Model	MAE (wpm)	RMSE (wpm)
Random Forest (2 features)	19.06	26.16
Nearest Neighbors (7 features)	18.38	25.72
Nearest Neighbors (5 features)	**18.36**	**25.68**
Random Forest (7 features)	18.82	25.73
Random Forest (5 features)	18.76	25.70
Gradient Tree Boosting (7 features)	18.95	25.90

Table 2: Performance of predicting human effort using Nonlinear Models with the complete set of features (7), the reduced set of features for the Nearest Neighbors and Random Forest (5), compared with the Random Forest with a set of two features.

As the performances described are very similar, we carry out the Wilcoxon statistical test at the 0.01 level to compare both the Random Forest and Nearest Neighbors models, with the full and reduced set of features. Nearest Neighbors achieved a p-value bellow the significance level, concluding that reducing the set of features improves model accuracy. On the counter part, Random Forest attained a p-value of *0.083*, above the significance level, concluding that the predictions are similar apart from using the complete or reduced set of features.

Additionally, Table 2 shows that Nearest Neighbors Regressor outperforms the remaining models considering both evaluation metrics, attaining 25.68 wpm RMSE, 6% better than the linear model baseline. The set of hyperparameters that optimize the Nearest Neighbour model are *number of neighbors = 95* and *minkowski distance metric*. Comparing the reduced set of features Nearest Neighbors with the Random Forest and Gradient Tree Boosting with the complete set of features, we observe p-values bellow the 0.01 significance level, proving that the models' results are statistically different. Ultimately, the Nearest Neighbors with the reduced set of features is the model that better predicts the target variable.

In the final analysis, when comparing the Nearest Neighbors to the Baseline Experiment Random Forest, we observe an RMSE improvement of 1.8%, statistically significant at the 0.01 level. So, in short, adding the features *number of punctuation tokens*, *number of stopwords* and *average word length*, to represent the cognitive load, reduces the prediction error.

6. Predicting Human Effort at the Job Level

By using the human effort at task level, we can estimate the total effort required by the crowd to complete a job. We define it as the job effort. This metric is valuable as it provides insights that are useful for pricing and crowd payments, as well as crowd management.

To calculate the job effort, we take into consideration the HITs available and the redundancy required for quality control. Since in Section 5.2 we predict the effort as the average speed-on-task, we must denormalize the results to estimate the job effort in hours. Equation 4 reflects the job effort estimation, where *T* is the number of HITs available in the job, the *Number of Tokens* is the input length per task, the *PredictedSoT* is the predicted mean speed-on-task and *Redun* is the redundancy required.

$$JobEffort = \sum_{t=0}^{T} (\frac{NumberTokens_t}{PredictedSoT_t} * Redund) \quad (4)$$

To verify the accuracy of our estimations, we compare the predicted job effort with our ground truth, that is, the sum of all executions' time-on-task. Then, to evaluate the results, we use the Mean Absolute Percentage Error (MAPE). This evaluation metric is the Mean Absolute Error (see Section 4.2) in percentage and allows to compare jobs with different number of tasks and/or total effort. The job effort estimation achieved 18.1% MAPE.

Figure 8 shows the ground truth versus estimated job effort per each job. Results show that our model is underpredicting that is, the predicted effort is inferior to the ground truth. Considering that we employ the predicted efforts at the HIT level to compute the job effort, and also that the number of tokens and redundancy are equal to the ground truth, we hypothesize that these underpredictions are the result of overpredictions at the HIT level.

7. Discussion

In this paper, we address the problem of predicting the effort to complete a given NET task, before the actual annotation takes place. Nearest Neighbors with five features is the model that attain the best performance (see Table 2) estimating human effort at the task level. Adding three out of the five proposed text preprocessing features to the model,

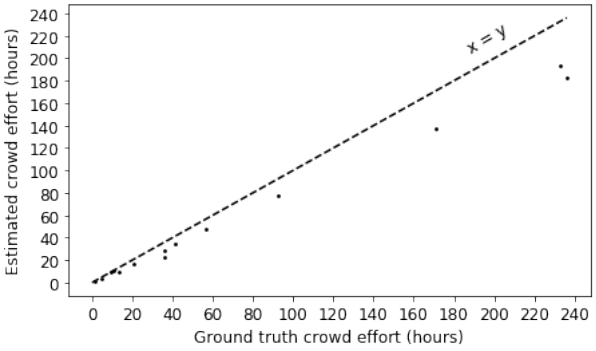


Figure 8: Estimated job effort in hours (y axis) as a function of the ground truth in hours. Each point in space represents one job. The x = y line is the identity line.

has a statistically significant improvement of 1.8% when comparing to the Baseline Experiment. The final set of features used by the Nearest Neighbors is *input length in number of tokens*, *number of categories of named entities*, *number of stopwords*, *number of punctuation tokens* and *average word length*.

To understand the importance of each feature, we decide to carry out an ablation study. Table 3 shows the rank of the features ordered by the highest impact on the model performance. The RMSE column is the score of the model without the given feature, and the difference is the discrepancy between the score and the initial model performance (*25.68 wpm*). That is to say that the *number of categories of named entities* is the feature with greater influence on the model performance. This feature is intuitively more relevant to explain the human effort, as the more entities to tag, the longer the task comprehension and the answering time. At the same time, the *number of stopwords* and the *number of punctuation tokens* prove the hypothesis formulated in Section 4.3 that these features impact the speed-on-task, the former by lowering the cognitive effort and the latter by improving the sentence comprehension. Finally, the *input length in number of tokens* and *average word length* are the features with minor impact on the model performance and consequently are less relevant to explain the target variable.

Removed Feature	RMSE (wpm)	Diff. (wpm)
Number categories named entities	27.01	1.33
Number stopwords	25.92	0.24
Number punctuation tokens	25.77	0.09
Input length in number of tokens	25.73	0.05
Average word length	25.70	0.03

Table 3: RMSE obtained when removing each feature from the Nearest Neighbors model and the difference between the performance attained without each feature and the performance when using the set of five features (25.68 wpm).

Another important result is the shortage of high speeds predictability. In Table 2, we observe a discrepancy between the MAE and RMSE results. RMSE, by definition, gives a relatively high weight to large errors, suggesting that our predictions have several outliers. Under those circumstances and upon further investigation, we searched for similarities among HITs whose ground truth speed-on-task surpass 250 wpm, a very large speed-on-task according to the data exploration detailed in Section 3. Observing the tasks' input, we noticed that most texts had short sentences with low complexity and without named entities to tag, e.g. "*I'm very sorry to hear that*" or "*Yes, I'll remember.*". Ultimately, these evidences indicate that there are uncertain factors, beyond the contributor carelessness or expertise (see Section 4.1) that impact human effort.

Figure 9 studies the speed-on-task aggregated by the number of instances of named entities by means of an error plot. An instance corresponds to the tagging result, for example, in Figure 1 there is one instance, i.e., the annotation of *"Barack Obama"* with the named entity category *"Person"*. The plot shows the average speed-on-task (represented with a black circle) and the standard deviation (represented by the whiskers). Observing the mean speed-on-task, we notice a decreasing trend when the number of instances of named entities increases. We expected this correlation as the number of instances influences the answering time, that in turn influence the speed-on-task. Nevertheless, we must take into consideration the high variance among the number of instances of named entities. This variance is the consequence of other factors impacting the human effort, such as the contributor profile and the HIT cognitive load. In either case, although the number of instances of named entities is a factor that affects human effort, we cannot use it as a predictor variable, as we do not have access to this information before the actual annotation takes place.

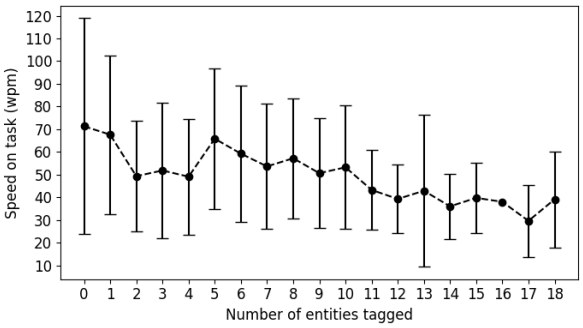


Figure 9: Mean and Standard Deviation of speed-on-task per number of instances of named entities.

In Section 4.1, we refer that the contributor profile may impact the HIT completion time, so during the dataset preprocessing step we discard HITs whose executions do not agree on the effort needed to complete the task. However, different efforts may be explained by distinct motivation or expertise (Rogstadius et al., 2011). Therefore, we study the contributor speed-on-task variance per each contributor and analyze its distribution. Observing Figure 10, we notice that the contributor mean speed distribution has a long tail with several contributors having speeds above 100 wpm. Given the speed-on-task distribution (see Figure 7), we recognize that contributor average speeds above 90 wpm are

very large. Figure 10 distribution can be explained by the contributors' set of skills, leading us to the conclusion that the contributor characteristics affects human effort. On the counterpart, we cannot use the contributor characteristics as predictor variables as the contributor assignment to the task is undetermined before the actual annotation takes place. However, if using crowd segmentation, some particularities as the contributor age, gender, country or fluent language may be investigated.

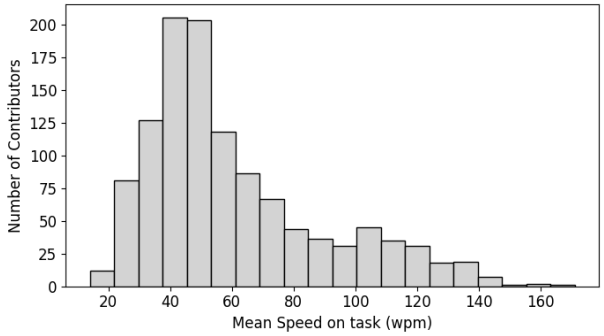


Figure 10: Distribution of the average speed-on-task over contributors.

When estimating the human effort at the job level, we noticed underpredictions, caused by overpredictions at the HIT level, that is, predicting higher speeds compared to the ground truth. This outcome corroborates the hypothesis that uncertain factors, before the actual annotation takes place, affect the human effort predictability.

Considering that in Section 2 we argue that the task completion time is divided into *answering* and *reading* times, by the end of this section we understand that poor effort predictions may relate to the shortage of the answering/reading times explainability, due to uncertain factors before the annotation takes place. We believe that this shortage is twofolded: the reading time depends on the contributor assigned, and the answering time depends on the number of instances of named entities.

8. Conclusion and Future Work

In this research, we used a data-driven approach to understand which factors affect the human effort for NET tasks and which strategies can be used to predict it. In the state-of-the-art, we found several factors impacting the human effort, such as the HIT cognitive load and the contributor's motivation. Based on this study, we extracted a set of features: *number of categories of named entities, input length in number of tokens, number of sentences, number of punctuation tokens, number of stopwords* and *average word length* (both with and without stopwords). Then, we performed a set of experiments with different models and feature combinations. We concluded that the Nearest Neighbors Regressor outperforms the remaining models, attaining 25.68 wpm RMSE, 6% better than the Linear Model Baseline. We also concluded that adding features related to the HIT cognitive load, specifically the *number of stopwords*, *number of punctuation tokens* and *average word length*, improve the models' performances. In the end, we expanded this estimation to the job level, achieving 18.1% MAPE when comparing to the ground truth.

When exploring the results of the experiments, we found some factors that cannot be measured before the annotation takes place, i.e., the instances of named entities and the contributors' profile, impacted the human effort predictability. Ultimately, we correlated these factors with effort overpredictions at the HIT level.

For future work, we would extend the effort estimation to other languages besides English. There is also room for improvement regarding the investigation of additional features that model the task cognitive load to improve effort predictions, e.g. investigate the impact of the named entities categories on the human effort. Finally, since the number of instances of named entities and the contributor's particularities impacts the human effort, we would further investigate these variables predictors.

9. Bibliographical References

Allen, L. K., Snow, E. L., Crossley, S. A., Jackson, G. T., and McNamara, D. S. (2014). Reading comprehension components and their relation to writing. *Annee Psychologique*, 114(4):663–691.

Baba, Y. and Kashima, H. (2013). Statistical quality estimation for general crowdsourcing tasks. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Part F1288:554–562.

Barrouillet, P., Bernardin, S., Portrat, S., Vergauwe, E., and Camos, V. (2007). Time and Cognitive Load in Working Memory. *Journal of Experimental Psychology: Learning Memory and Cognition*, 33(3):570–585.

Dyson, M. C. (2001). The influence of reading speed and line length on the effectiveness of reading from screen. *International Journal of Human Computer Studies*, 54(4):585–612.

Eickhoff, C. and de Vries, A. P. (2013). Increasing cheat robustness of crowdsourcing tasks. *Information Retrieval*, 16(2):121–137.

Feyisetan, O., Luczak-Roesch, M., Simperl, E., Ramine, T., and Shadbolt, N. (2015). Towards Hybrid NER: A Study of Content and Crowdsourcing-Related Performance Factors. *Springer International Publishing Switzerland 2015*, 9088:525–540.

Feyisetan, O., Simperl, E., Luczak-Roesch, M., Tinati, R., and Shadbolt, N. (2017). An extended study of content and crowdsourcing-related performance factors in named entity annotation. *Semantic Web*, 9(3):355–379.

Finin, T., Murnane, W., Karandikar, A., Keller, N., Martineau, J., and Dredze, M. (2010). Annotating named entities in Twitter data with crowdsourcing. *Human Language Technologies Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, 2010(June):80–88.

Finnerty, A., Kucherbaev, P., Tranquillini, S., and Convertino, G. (2013). Keep it simple: Reward and Task Design in Crowdsourcing. *Proceedings of the Biannual Conference of the Italian Chapter of SIGCHI on - CHItaly '13*, (September):1–4.

Hassan, U. u. and Curry, E. (2013). A Capability Requirements Approach for Predicting Worker Performance in Crowdsourcing. *Proceedings of the 9th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing*, pages 429–437.

Hirotani, M., Frazier, L., and Rayner, K. (2006). Punctuation and intonation effects on clause and sentence wrap-up: Evidence from eye movements. *Journal of Memory and Language*, 54(3):425–443.

Hirth, M., Hoßfeld, T., and Tran-Gia, P. (2011). Anatomy of a crowdsourcing platform - Using the example of microworkers.com. *Proceedings - 2011 5th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, IMIS 2011*, pages 322–329.

Hirth, M., Scheuring, S., Hossfeld, T., Schwartz, C., and Tran-Gia, P. (2014). Predicting result quality in Crowdsourcing using application layer monitoring. *2014 IEEE 5th International Conference on Communications and Electronics, IEEE ICCE 2014*, (July):510–515.

Jain, A., Sarma, A. D., Parameswaran, A., and Widom, J. (2017). Understanding Workers, Developing Effective Tasks, and Enhancing Marketplace Dynamics: A Study of a Large Crowdsourcing Marketplace. 10(7):829–840.

Kincaid, J. P., Fishburne, R. P., Rogers, R. L., and Chissom, B. S. (1975). Derivation of New Readability Formulas (Automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy Enlisted Personnel. *Technical Training*, Research B(February):49.

Krippendorff, K. (1980). Reliability. In *Content Analysis: An Introduction to Its Methodology*, chapter 12. Sage, London.

Landis, J. R. and Koch, G. G. (1977). The Measurement of Observer Agreement for Categorical Data. *Biometrics*, 33(1):159.

Lofi, C., Selke, J., and Balke, W.-T. (2012). Information Extraction Meets Crowdsourcing: A Promising Couple. *Datenbank-Spektrum*, 12(2):109–120.

Lu, Q., Yang, Y. S., Li, Z., Chen, W., and Zhang, M. (2019). M-CNER: A corpus for Chinese named entity recognition in multi-domains. *LREC 2018 - 11th International Conference on Language Resources and Evaluation*, pages 4457–4461.

Marrero, M., Urbano, J., Sánchez-Cuadrado, S., Morato, J., and Gómez-Berbís, J. M. (2013). Named Entity Recognition: Fallacies, challenges and opportunities. *Computer Standards and Interfaces*, 35(5):482–489.

Martin, D., Carpendale, S., Gupta, N., HoBfeld, T., Babak, N., Redi, J., Siahaan, E., and Wechsung, I. (2017). *Understanding the Crowd: Ethical and Practical Matters in the Academic Use of Crowdsourcing*, volume 10264.

Mason, W. and Watts, D. J. (2010). Financial incentives and the "performance of crowds". *ACM SIGKDD Explorations Newsletter*, 11(2):100.

Nowak, S. and Rüger, S. (2010). How reliable are annotations via crowdsourcing? A study about inter-annotator agreement for multi-label image annotation. *MIR 2010 - Proceedings of the 2010 ACM SIGMM International Conference on Multimedia Information Retrieval*, pages 557–566.

Rayner, K., Schotter, E. R., Masson, M. E., Potter, M. C., and Treiman, R. (2016). *So much to read, so little time: How do we read, and can speed reading help?*, volume 17.

Rogstadius, J., Kostakos, V., Kittur, A., Smus, B., Laredo, J., and Vukovic, M. (2011). An Assessment of Intrinsic and Extrinsic Motivation on Task Performance in Crowdsourcing Markets. *Fifth International AAAI Conference on Weblogs and Social Media*.

Sautter, G. and Böhm, K. (2013). High-throughput crowdsourcing mechanisms for complex tasks. pages 873–888.

Sweller, J. and Chandler, P. (1994). Why some material is Difficult to Learn. *Cognition and Instruction*, 12(3):185–233.

Voyer, R., Nygaard, V., Fitzgerald, W., and Copperman, H. (2010). A hybrid model for annotating Named Entity training corpora. *ACL 2010 - LAW 2010: 4th Linguistic Annotation Workshop, Proceedings*, (July):243–246.

Wilcoxon, F. (1946). Individual comparisons of grouped data by ranking methods. *Journal of economic entomology*, 39(6):269.

Yang, J., Redi, J., Demartini, G., and Bozzon, A. (2016). Modeling Task Complexity in Crowdsourcing. *The Fourth AAAI Conference on Human Computation and Crowdsourcing*, (October):249–258.

Yin, X., Wenjie Liu, Wang, Y., Yang, C., and Lu, L. (2014). What? How? Where? A Survey of Crowdsourcing. *Frontier and Future Development of Information Technology in Medicine and Education*, pages 221–232.