Artificial Intelligence and Social Computing, Vol. 163, 2025, 141–151
https://doi.org/10.54941/ahfe1006045

Web-Based Human-centred Explainability of NLP Tasks With Rationale Mapping Theory

Andrea Tocchetti, Valentina Naldi, and Marco Brambilla

Politecnico di Milano, Dipartimento di Elettronica, Informazione e Bioingegneria, Milano, MI 20133, Italy

ABSTRACT

Recently, human-generated data has been used to explain machine learning and NLP models. Such methods usually focus only on labelling results with relevant human-generated tags, explicitly identifying objects, actions, or other elements in the output. Therefore, potential explanations only refer to the data elements and the model parts that produce them. The cognitive process applied by the human to perform the task is completely neglected. We claim the latter is essential to provide complete and human-understandable explanations of results, models, and processes. Some existing approaches studied in linguistics, such as rationale mappings, aim to achieve this objective by formalizing tree-based data structures to collect human rationale applied to NLP tasks. This work presents a web-based, human-centred approach to collect rationale mappings for various NLP tasks. Our contribution includes the formalization of the Rationale Mapping theory, the design of the human-computer interaction paradigm implementing the theory, the specification of the data collection process, its implementation as a crowdsourcing web application, and its validation with experimental studies showing its reliability and effectiveness.

Keywords: Human-centred approach, Crowdsourcing, Web application, NLP, Explainability

INTRODUCTION

Due to the increasing complexity of Machine Learning (ML) models, recent research focused on methods to explain their results. This research stream has tackled models addressing problems spanning image analysis, text understanding, content generation, and more (Danilevsky, 2020). Various techniques have been designed depending on the kind of models and the type of problem or scenario. In many cases, crowdsourcing techniques were involved to collect human-generated data to explain models (Tocchetti, 2022). Such approaches usually label model outputs with human-generated tags, explicitly identifying objects, actions, or other elements. Therefore, potential explanations only refer to the data features and the model parts that produce them. The cognitive process applied by the human to perform the task is completely neglected, providing no alignment or reconciliation between how a person would solve a task and how a ML model generates the results.

We claim this missing link between cognitive processes and AI/ML model behavior specification is a crucial weakness of most explainability approaches. In this paper, we propose to bridge this gap for the specific case of NLP task explanations. Some existing approaches studied in linguistics, such as Rationale Mappings (Tocchetti, 2023), aim to achieve this objective by formalizing tree-based data structures to collect human rationale applied to NLP tasks. Building on this research, this work presents a web-based, human-centred approach to collect rationale mappings for various NLP tasks. The paper's contribution includes the design of the human-computer interaction paradigm implementing the theory, the specification of the data collection process, its implementation as a crowdsourcing web application, and its validation with experimental studies showing its reliability and effectiveness. The usability and workload of the application are assessed through standardized user questionnaires. In this paper, we cover three NLP tasks of interest: Sentiment Analysis, Text Summarization, and Question answering, as we consider them to cover a sufficient variety among the most important NLP tasks.

BACKGROUND & RELATED WORKS

Explainable AI & Natural Language Processing (NLP)

Explainable AI (XAI) is a research field striving *to develop inherently explainable systems and explainability techniques that faithfully explicit the behavior of complex machine learning models tailoring their explanation in an understandable way for humans* (Tocchetti, 2022). In recent years, the relevance and popularity of XAI have dramatically strengthened due to the increasingly widespread usage of black-box applications, *i.e.*, systems with observable input(s) and output(s) and hard-to-understand internals. However, grasping the system behavior is relevant to the system's developers, final users, and the people affected by the decisions taken by such a system (Xu, 2019). A fundamental challenge when developing XAI methods is providing human-understandable explanations that faithfully represent the model behavior (Tocchetti, 2022), highlighting the need for explanations aligned with human reasoning and tailored for the intended audience.

This article contextualizes explainability in Natural Language Processing (NLP). While several XAI techniques exist in NLP, *e.g.*, saliency-based approaches, declarative representations, and natural language, they mostly rely on the (sometimes improper) assumption the human receiving the explanation will interpret it as intended (Danilevsky, 2020). This article focuses on three NLP tasks: Sentiment Analysis, Text Summarization, and Question Answering.

Sentiment Analysis (SA) determines whether a subjective text (*e.g.*, people's opinions, thoughts, etc.) conveys a positive, negative, or neutral view. When performing this task, human interpreters would identify the portions of the text expressing subjective opinions and subsequently assess and combine their views to derive the overall sentiment of the text. Human rationale describing the sentiment attribution can be represented by associating the output sentiment with the parts of the input text that most influenced the output label.

Text Summarization (**TS**) generates a summarized version of a given text, either by reporting (extractive approach) or rephrasing (abstractive approach) parts of the input. When performing the task, human interpreters would identify and summarize the most important information in the text. Rationale can be represented by mapping its informative content to where it is extracted from the input text.

Question Answering (**QA**) provides a relevant answer to a question given a paragraph or a set of documents containing relevant information for answering it. When performing the task, human interpreters would understand the type of information they must find and developing the answer by inspecting the provided paragraph. Rationale can be represented by extracting paragraph(s), sentence(s), or sub-sentence(s) meaningful to the question or containing the answer.

RATIONALE MAPPINGS AND RATIONALE TREES

Rationale Mappings and Rationale Trees are the fundamental blocks of the formalization introduced by Tocchetti et al. (Tocchetti, 2023) for structuring the thought process humans apply when performing NLP tasks. These structures are based on Argumentation Mining and the Pragma-Dialect Theory (Palau, 2009). This section provides a summary of such a formalization for the chosen tasks. We advise the reader to inspect the full article (Tocchetti, 2023) for a complete understanding of the individual tasks' definitions and constraints.

Rationale Mappings. Given any of the considered NLP tasks, a Rationale Mapping is a triple ⟨ text, text, label ⟩ where text is a word or a set of consecutive words from a given text, and label is a term defining the relationship between texts involved. Rationale mappings organize individual humans' analytical reasoning steps applied to NLP tasks. Three types of Rationale Mappings are common to the considered tasks.

- **External Mappings** (**EM**) represent human reasoning applied to two words or sequences of words belonging to different texts. In external mappings, the label can be either specific to the NLP task when it involves a discrete output or when specific terms describe the applied approach or a generic linguistic label (*i.e.*, semantic and syntactic). Furthermore, the latter can be extended with textual descriptions to detail the thought process further. External mappings can be subject to simplifications when one of the texts and the label coincide, therefore appearing in the form ⟨ text, label ⟩.
- **Internal Mappings** (**IM**) represent human reasoning applied to two words or sequences of words belonging to the same text.
- **Resolution Mappings** (**RM**) represent internal mappings applied for anaphora/coreference resolution between two words or sequences of words belonging to the same text. In resolution mappings, the label expresses the type of resolution applied to the two texts involved. Resolution mappings cannot be subject to simplifications.

Our methodology focuses on detailing External Mappings and Resolution Mappings only, hence excluding Internal Mappings since such a level of detail is typically not covered by explanations.

Rationale Trees. Rationale Mappings can be hierarchically organized in tree structures to detail the human reasoning applied to generate the output. The root node represents the task, *i.e.*, its input(s) and output. The other nodes are Rationale Mappings, each further detailing the relationship between the texts in their parent node. In such structures, multiple child nodes coordinatively detail their parent's rationale (Palau, 2009). The deeper the node, the more specific the rationale it describes. In Rationale Trees, External Mappings can be inner or leaf nodes, while Resolution Mappings can only be leaf nodes. Moreover, External Mappings can have any mapping as child node. Building a proper, non-redundant Rationale Tree requires enforcing some conditions between a child node and its parent node, *i.e.*, either one of the texts in the child node must be a sub-text of its corresponding text in its parent node. Regarding sibling nodes, they can't detail relationships between the same (sub-)texts or words.

METHOD

This section introduces an approach to collecting human rationale to build Rationale Trees (see Figure 1). The data collection step involves human actors in creating Rationale Mappings. These will be then organized into Individual Rationale Trees (IRT), *i.e.*, data structures built by a single participant. Ultimately, the collected trees are merged into Complete Rationale Trees (CRT), *i.e.*, data structures built by combining multiple IRTs. Multiple Rationale Mappings and Individual Rationale Trees are collected for each data point. On the other hand, only one Complete Rationale Tree is provided for each. Such structures are collected for Sentiment Analysis, Text Summarization, and Question Answering, based on their characteristics and intended human meaning of the tasks themselves.

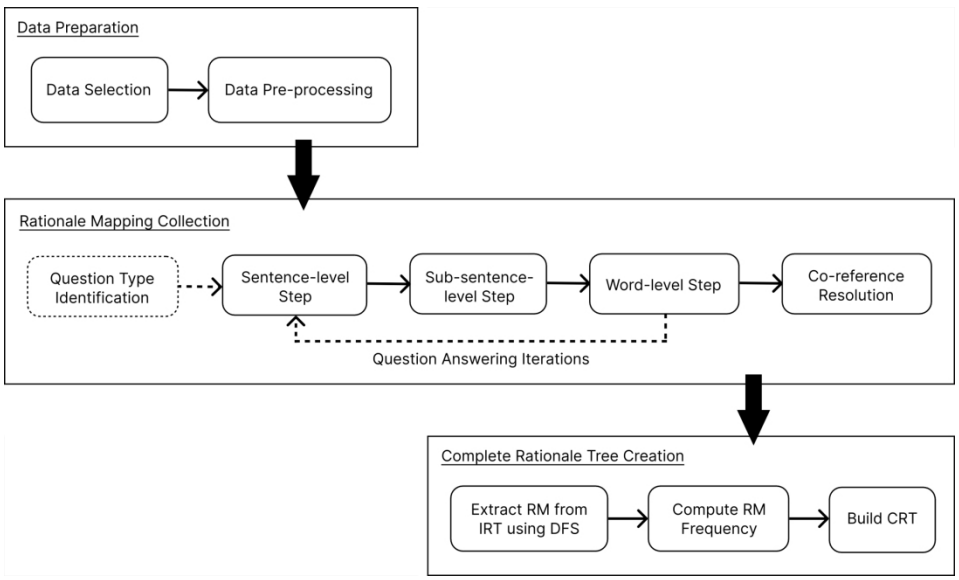

Figure 1: A schematic representation of the process to generate rationale trees.

Sentiment analysis is a text classification task that accepts free text as input and defines a discrete output (*i.e.*, Positive, Negative, or Neutral). External Mappings allow for a simplified representation and are defined between input and output, establishing the slice of the input text contributing towards the final sentiment. Labels describe the sentiment between the texts involved in a mapping, *i.e.*, Positive or Negative.

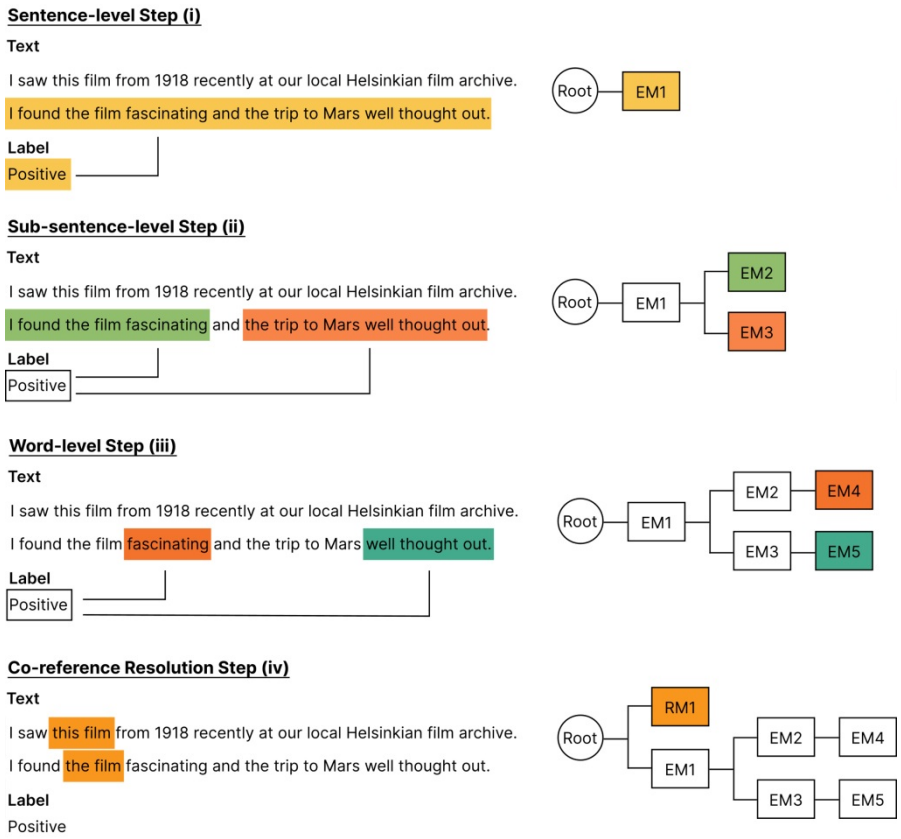

Figure 2: Rationale mapping collection process for sentiment analysis.

Text summarization is a text generation task that accepts a free-text input and generates a free-text output. As two approaches exist to perform such a task, *i.e.*, abstractive and extractive, these were chosen as labels for External Mappings. Whenever an Extractive approach is applied, a simplification can be considered.

Question answering is a text generation task that accepts multiple free text inputs and generates free-text output. A new mapping type is defined since the task involves a well-defined thought process to be performed (Calijorne Soares, 2020). Abstractive Mappings define the kind of question to be answered, *i.e.*, Yes/No Question, Wh-Question (classified even further), Choice Question, and Disjunctive Question. Given the complexity of the task, external mappings are defined between pairs of input(s) and output (*i.e.*, question-paragraph, paragraph-answer, and question-answer). In such a task,

the (generic) syntactic and semantic labels represent the interplay between the texts in the mapping. Since External Mappings are defined between couples of input(s) and output, these are given a different meaning based on the coupling, *e.g.*, an EM between the question and the paragraph represents the texts that led the human actor to understand which part of the paragraph answers the question.

Data preparation. Before collecting Rationale Mappings, a set of data points for each task, *i.e.*, the texts associated with an NLP task, must be chosen. Such data must be complete enough to describe a task instance, *i.e.*, they must include input(s) and output, or these must be derivable from the data. Question Answering data instances must satisfy an additional constraint, *i.e.*, multiple questions cannot be asked in the same text, as there can only be a single Abstractive Mapping for each Rationale Tree. Invalid instances can be dropped or split into multiple valid ones, finally leading to multiple data points with one question each. Finally, additional pre-processing and text cleaning operations may be needed based on the specific characteristics of the chosen dataset.

Rationale mappings collection. Initially, participants are provided with a theoretical description of Rationale Mappings, followed by guided exercises to strengthen their understanding of the activity. Each guided exercise is partially pre-compiled to show how the task should be performed and includes a sample solution users can use to assess the validity of their mappings. After correctly completing these exercises, participants can proceed with the actual data collection activity. The annotation process involves a sequence of four Rationale Mapping creation steps (see Figure 2): a sentence-level step (i), a sub-sentence-level step (ii), a word-level step (iii), and a final coreference resolution step (iv). While the latter allows collecting Resolution Mappings, the others guide the user into providing External Mappings with different levels of detail. Such a process is common to all tasks except for Question Answering, as an additional initial step to define the Abstract Mapping and three iterations of the first steps (i-iii), one for each couple of input(s) and output (*i.e.*, question-paragraph, paragraph-answer, and question-answer) must be performed. During these steps, participants are asked to select the texts to be included in the mappings, while the label associated with the task will be chosen for each mapping. These can be automatically inferred through task-specific strategies, *e.g.*, in Sentiment Analysis, the labels in External Mappings are directly associated with the sentence's sentiment. Then, Rationale Mappings are obtained by combining the texts provided by human actors and the automatically inferred labels, and potential duplicates are removed. Finally, the mappings provided by the same crowd-worker for each data point are organized into Individual Rationale Trees by applying Algorithm 1 (see Figure 3).

Algorithm 1 Rationale Tree Creation Algorithm

```
1: procedure AddNode(nodeToAdd, currentNode, siblings, parentNode)
2:     if isAncestor(currentNode, nodeToAdd) then
3:         if currentNode.children is empty then
4:             currentNode.children.push(nodeToAdd)
5:         else
6:             return AddNode(nodeToAdd,currentNode.firstChild,
                              currentNode.children.pop(),currentNode)
7:         end if
8:     else
9:         if siblings is empty then
10:            parentNode.children.push(nodeToAdd)
11:        else
12:            return AddNode(nodeToAdd,siblings.nextSibling,
                              siblings.pop(),parentNode)
13:        end if
14:    end if
15: end procedure
```

Figure 3: The algorithm describing the approach to generate rationale trees.

Complete rationale tree creation. Complete Rationale Trees are obtained by merging all the Individual Rationale Trees produced for each data point. In this process, Rationale Mappings are extracted by applying tree search algorithms to each Individual Rationale Tree. Complete Rationale Trees are created using the same algorithm for obtaining Individual Rationale Trees and considering all Rationale Mappings together. As these may appear multiple times, Complete Rationale Trees include a frequency score for each node, allowing tuning the level of detail of the final tree.

IMPLEMENTATION

Preliminary Validation. A preliminary study to validate the initial design of the approach and whether it would produce the expected outcome was performed, validating its effectiveness while collecting useful improvements.

Data structure. The final dataset to be collected describes Rationale Trees for a chosen set of dat a points from well-known datasets. Rationale Mappings include the text(s) and the indexes representing the position of their first and last word, the label, and the mapping type. In Complete Rationale Trees the frequency score is also stored. Some mappings (*e.g.*, the Abstractive Mapping) require storing additional data (*e.g.*, the question and its specialization). In Rationale Trees, each node (*i.e.*, a Rationale Mapping) additionally stores a reference to their child nodes, if any. The root node keeps the task's input(s) and output.

Requirements and design. The main requirement of the application is to enable users to provide the rationale they apply to a set of NLP tasks of interest. This involves displaying the users a data point and allowing them to perform the steps prescribed for a given task. When a task is chosen for the first time, a tutorial and three guided examples are provided to teach the user about Rationale Mappings. In the activity, users are displayed the instance to annotate and the panels to perform the required annotation steps (i-iv). Each panel displays the text(s) and the components the annotator works on. In particular, the sentence-level step (i) is implemented to allow the user to pick the sentences from a list, while the sub-sentence (ii) and the word-level (iii) steps require users to select and highlight portions of the shown

text. Whenever a mapping is created, it is added to a list visible to the user, allowing them to delete any undesired mapping. Finally, coreferences (iv) are identified by highlighting portions of the texts. In Question Answering, the three iterations are performed separately, only showing the texts involved in each specific iteration and thus emphasizing their separation. Additionally, users pick the question type from a predefined list and highlight the portion of text used to define it.

EXPERIMENT

Experiment setup. The experiment involved three collections, one for each task, each containing 20 instances sampled randomly from well-known datasets, *i.e.*, the Large Movie Review dataset (Maas, 2011) for Sentiment Analysis, the CNN/Daily Mail dataset (Nallapati, 2016) for Text Summarization, and the SQuAD 2.0 dataset (Rajpurkar, 2018) for Question Answering). The input text(s) length in Sentiment Analysis and Question Answering is lower than 1000 characters, while in Text Summarization, it is lower than 2500. The application was shared with 151 participants from our institution, 130 males and 21 females, with an average age of 23.8 and a standard deviation of 1.36. The experiment allowed collecting 1495 Individual Rationale Trees (569 for Sentiment Analysis, 473 for Text Summarization, and 453 for Question Answering). Individual Rationale Trees are organized into Complete Rationale Trees, finally leading to 20 Complete Rationale Trees per task, one for each data point. An example is reported in Figure 4. The final dataset is publicly available on GitHub. The application additionally recorded the time taken to provide each Rationale Tree. The average time to annotate an instance is 5 minutes for Sentiment Analysis, 14 minutes for Text Summarization, and almost 7 minutes for Question Answering. The length of the input texts probably causes the longer time taken for text summarization. Participants were additionally asked to fill out a form to review their experience after using the application. The form gathered basic information on the user and general feedback on the application. Moreover, users were asked questions based on the System Usability Scale (SUS) (Brooke, 1995) to evaluate usability, as well as questions inspired by the NASA-TLX method (Hart, 1988).

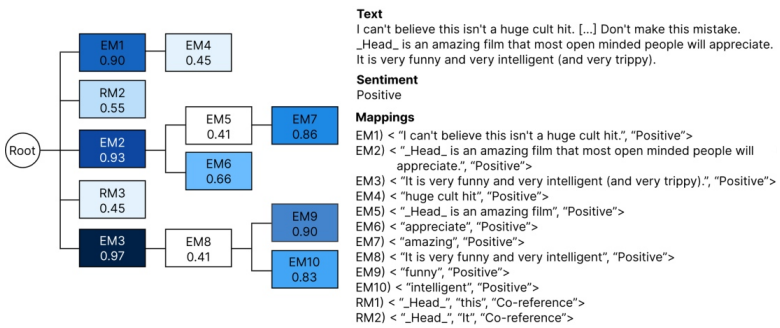

Figure 4: A complete raionale tree for sentiment analysis. nodes are coloured according to their frequency score. The higher the score, the darker the colour. Only rationale mappings with a frequency score higher than 0.4 were reported.

Discussion. Inspecting the questionnaires and the participant feedback contributed towards determining potential areas of improvement while underlining the application's practical design. The SUS score computed from the submitted questionnaires was 65.7, demonstrating that the system's usability is sufficient and falls in the 40th percentile ranking (Brooke, 2013). The question with the lowest score (*i.e.*, a value of 4.64) asks whether the users would use the system frequently. This was expected since most of the efforts of this work have been directed towards making the process understandable and smooth for users rather than making it entertaining. Moreover, the questions addressing the system's complexity and cumbersomeness resulted in a pretty low effect on the total score (*i.e.*, a value of 6.21 and 5.76, respectively). Furthermore, many users considered Question Answering the most complex and repetitive task. Striving to ease the annotation process for this task, one may remove the question-answer iteration and derive those mappings from those provided by the users in previous iterations. This may be possible by considering the texts of the question-paragraph and paragraph-answer mappings that share the exact paragraph text. Another question providing a low contribution towards the final score (*i.e.*, a value of 6.14) evaluates the participants' confidence in using the system. The comments provided by the users unveil that many doubts are related to why three iterations are needed in Question Answering. Other than removing the last iteration as suggested before, an improvement in this direction could be adding optional sections to the tutorial, further detailing the reasons behind users' annotation steps. On the other hand, questions assessing whether users think that most people would learn to use this system very quickly, whether users deemed they needed the support of a technical person to use the system, and whether they needed to learn many things before using the system strongly impacted the final score (*i.e.*, a value of 6.89, 8.61, and 7.77, respectively). An approximated NASA-TLX score of 56.9 was computed. Even though it is considered a high result (Prabaswari, 2019), it was quite expected since performing the tasks requires reading and understanding a lot of text (*e.g.*, the tutorial, the data points, etc.). Striving to reduce the workload perceived by users, it would be possible to provide them with hints on the portions of text that are likely to be involved in mappings. Such hints could be displayed to users as highlights in the text. Despite the potential reduction in the user's workload, adopting it requires evaluating the bias such an approach may introduce since users may follow such advice unthinkingly, resulting in Complete Rationale Trees lacking complexity. Another way to reduce the workload may be allowing users to perform and submit only some annotation steps (i-iv) for each data point, providing their outcome as a starting point for another user's task iteration. Similarly, task-specific changes could be applied.

CONCLUSION

This article presents a novel approach to collecting complex data structures called Rationale Trees. Such structures organize the human cognitive process applied to NLP tasks to improve explainability of NLP tasks. The proposed

methodology was designed, validated through a preliminary experiment, and implemented into a web application. Data is collected by engaging human interpreters in performing the implemented data flow, finally leading to a dataset tailored to improve model explainability while being intrinsically human understandable. Experiments revealed the approach's effectiveness in driving participants' thought processes and collecting Rationale Mappings and Trees. Future works will enhance the approach and implementation to ease the data collection process and extend the method to other NLP tasks, like Natural Language Inference (NLI) and Claim Verification.

REFERENCES

Brooke, J. (1995). "Sus: A quick and dirty usability scale". Usability Eval. Ind. 189.

Brooke, J. (2013). "Sus: A retrospective". Journal of Usability Studies 8, 29–40.

Calijorne Soares, M. A., Parreiras, F. S. (2020). "A literature review on question answering techniques, paradigms and systems". Journal of King Saud University – Computer and Information Sciences 32(6), 635–646.

Danilevsky, M., Qian, K., Aharonov, R., Katsis, Y., Kawas, B., Sen, P. (2020). "A survey of the state of explainable AI for natural language processing". In: Proc. of the 1st Conference of the Asia-Pacific Chapter of the Assoc. for Comp. Linguistics and the 10thInt.l Joint Conference on Natural Language Processing. pp. 447–459. Assoc. for Comp. Linguistics, Suzhou, China.

Hart, S. G., Staveland, L. E. (1988). "Development of nasa-tlx (task load index): Results of empirical and theoretical research". In: Human Mental Workload, Advances in Psychology, vol. 52, pp. 139–183. North-Holland.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., Potts, C. (2011). "Learning word vectors for sentiment analysis". In: Proc. of the 49th Annual Meeting of the Assoc. for Comp. Linguistics: Human Language Technologies. pp. 142–150. Assoc. for Comp. Linguistics, Portland, Oregon, USA.

Nallapati, R., Zhou, B., dos Santos, C., Gul.cehre, C.., Xiang, B. (2016). "Abstractive text summarization using sequence-to-sequence RNNs and beyond". In: Proc. of the 20th SIGNLL Conference on Computational Natural Language Learning. pp. 280–290. Assoc. for Comp. Linguistics, Berlin, Germany.

Palau, R. M., Moens, M. F. (2009). "Argumentation mining: the detection, classification and structure of arguments in text". In: Proc. of the 12thInt.l Conference on Artificial Intelligence and Law. pp. 98–107. ICAIL '09, Association for Computing Machinery, New York, NY, USA.

Prabaswari, A. D., Basumerda, C., Utomo, B. W. (2019). "The mental workload analysis of staff in study program of private educational organization". IOP Conference Series: Materials Science and Engineering 528(1), 012018.

Rajpurkar, P., Jia, R., Liang, P. (2018). "Know what you don't know: Unanswerable questions for SQuAD". In: Proc. of the 56th Annual Meeting of theAssoc. for Comp. Linguistics (Volume 2: Short Papers). pp. 784–789. Assoc. for Comp. Linguistics, Melbourne, Australia.

Tocchetti, A., Brambilla, M. (2022). "The role of human knowledge in explainable AI". Data 7(7).

Tocchetti, A., Yang, J., Brambilla, M. (2023). "Rationale trees: Towards a formalization of human knowledge for explainable natural language processing". In: Proc. of the 4th Italian Workshop on Explainable Artificial Intelligence co-located with 22ndInt.l Conference of the Italian Assoc. for Artificial Intelligence (AIxIA 2023), Roma, Italy, November 8, 2023. CEUR Workshop Proc., vol. 3518, pp. 29–46. CEUR-WS.org.

Xu, F., Uszkoreit, H., Du, Y., Fan, W., Zhao, D., Zhu, J. (2019). "Explainable AI: A brief survey on history, research areas, approaches and challenges". In: Natural Language Processing and Chinese Computing. pp. 563–574. Springer Intl. Publishing, Cham.