CiTIUS at the TREC 2021 Health Misinformation Track

Marcos Fernández-Pichel, Manuel Prada-Corral, David E. Losada, Juan C. Pichel and Pablo Gamallo
Centro Singular de Investigación en Tecnoloxías Intelixentes (CiTIUS), Universidade de Santiago de Compostela
15782 Santiago de Compostela, Spain
{marcosfernandez.pichel,manuel.deprada.corral,david.losada,juancarlos.pichel,pablo.gamallo}@usc.es

ABSTRACT

The TREC Health Misinformation Track pursues the development of retrieval methods that promote credible and correct information over misinformation for health-related information needs. In this year, only the AdHoc Web Retrieval task was carried out. Its main goal was developing search technologies that promote credible and correct information over incorrect information. In these working notes, we present the CiTIUS team's multistage retrieval system for addressing this task.

CCS CONCEPTS

• **Health misinformation, Multistage retrieval, NLP**;

1 INTRODUCTION

Search engines are widely used to find health advice online [1]. However, the web is plagued with misinformation about diseases and treatments [2]. It has been demonstrated that interacting with incorrect search results leads to poor health-related decisions being made [3].

The TREC Health Misinformation Track pursues the development of retrieval methods that promote credible and correct information over misinformation for health-related information needs. In these working notes, we present a multistage retrieval system that our team, from CiTIUS at the University of Santiago de Compostela (Spain), has developed for addressing this task.

Nowadays, modern architectures are often formed by a multistage pipeline, with an initial (document) retrieval phase, followed by one or more re-rankers and different combination stages [4–7]. We designed an architecture that allowed us to test different signals and combinations methods to better detect misinformation, and promote correct and credible documents. The different runs submitted to the TREC track were obtained from different configurations of our processing pipeline.

These working notes are organised as follows: Section 2 briefly explains the task and its objectives, Section 2.1 presents the data provided by the organisers, Section 3 presents a high-level view of our retrieval system, Section 4 introduces the solutions that we presented to the task, and, finally, Sections 5 and 6 expose the obtained results and some conclusions.

2 ADHOC RETRIEVAL TASK

The main goal of the 2021 AdHoc Web Retrieval was to develop search technologies that promote credible and correct information over incorrect information, assuming that interacting with incorrect health search results, leads to poor decisions being made.


```
<topic>
  <number>1234</number>
  <query>dexamethasone croup</query>
  <description>Is dexamethasone a good treatment for croup?</description>
  <narrative>Croup is an infection of the upper airway and causes swelling,
    which obstructs breathing and leads to a barking cough. As one kind of
    corticosteroids, dexamethasone can weaken the immune response and
    therefore mitigate symptoms such as swelling. A very useful document
    would discuss the effectiveness of dexamethasone for croup, i.e. a very
    useful document specifically addresses or answers the search topic's
    question. A useful document would provide information that would help
    a user make a decision about treating croup with dexamethasone, and
    may discuss either separately or jointly: croup, recommended treatments
    for croup, the pros and cons of dexamethasone, etc.</narrative>
  <disclaimer>We do not claim to be providing medical advice, and medical
    decisions should never be made based on the stance we have chosen.
    Consult a medical doctor for professional advice.</disclaimer>
  <stance>helpful</stance>
  <evidence>https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5804741/</evidence>
</topic>
```

Figure 1: A TREC 2021 Health Misinformation Track topic (Topic 101).

2.1 Dataset

The organisers of the track opted for the no-clean version of the C4 dataset. This corpus was created by Google to train their sequence-to-sequence T5 model [8]. The collection is formed of text extracts from the April 2019 snapshot of Common Crawl[1], and it contains approximately 1 billion English documents.

Each topic provided by the organisers consists of a health-related query. The topic represents a user trying to determine whether or not a treatment is useful for a given disease or condition. All topics have a fixed structure (see Figure 1), containing a stance field that states if the treatment is actually helpful or not for the disease.

If a treatment is considered helpful then correct documents will be those supportive of the treatment. If the treatment is considered unhelpful then correct documents would dissuade users from applying the treatment for the disease.

3 MULTISTAGE RETRIEVAL SYSTEM

Figure 2 presents our multistage retrieval system whose goal is to identify and promote correct and credible documents. Combining different signals and fusion strategies, we produced the runs that we submitted for the competition (see Section 4).

3.1 Document Retrieval

In this first stage, we indexed the corpus using the Anserini library [9]. Afterwards, we applied a BM25 search whose parameters were set to $k_1 = 0.9$ and $b = 0.4$, which is a setting in the range of their recommended values. We employed Pyserini's[2] implementation

[1] https://commoncrawl.org/
[2] https://github.com/castorini/pyserini

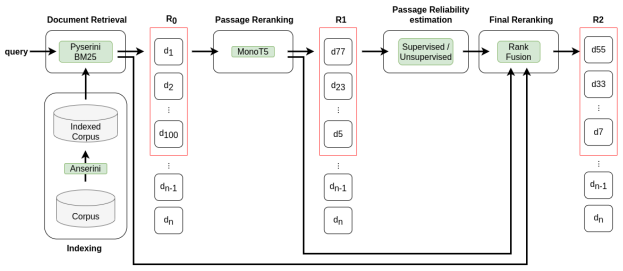

Figure 2: Multistage retrieval system for misinformation detection.

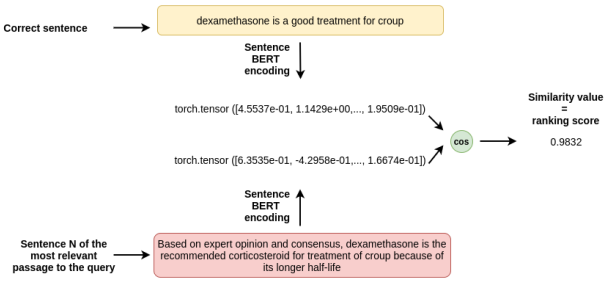

Figure 3: Unsupervised strategy for passage reliability prediction.

of BM25. This outputs an initial ranking of documents ordered by decreasing estimation of relevance to a query.

3.2 Passage Re-ranking

The main goal of this stage was to skip the noisy or offtopic parts of the texts and focus on the most relevant passages. To that end, we utilised state-of-the-art neural re-ranking technology. More specifically, our method is based on Nogueira's work [10] which used the T5 model [8] for predicting document relevance and ranking. In our case, we took advantage of the Python library Pygaggle[3] that provides models already fine-tuned for a passage ranking task. We opted for a monot5-base model trained on the Med-MARCO, a medical subset of MS MARCO passage ranking dataset.

At prediction time, a sliding window over the documents was applied (window size was equal to 6 sentences and the stride was equal to 3), and the passage selected was that producing the highest score. The ranking of documents was then reordered based on this score. Following the lessons learnt from our experiments in 2020 [11], we opted for reordering the top 100 documents based on their high scoring passages, while the documents ranked from the 101th position remained in their original positions.

3.3 Passage Reliability Prediction

Two alternative approaches were tested for passage reliability estimation, namely: a supervised classifier and a sentence similarity unsupervised strategy.

3.3.1 Supervised approach. A T5-base model was fine-tuned to classify passages as "reliable" or "unreliable" to a given query:

$$Query : q\ Passage : p\ Reliability :$$

The model was trained using the 2019 Decision Track data and the 2020 TREC Health Misinformation Track data. It should be noticed that we kept only the topics that contained at least one harmful document. Moreover, two different labelling approaches were tested: first, categorising as "reliable" the documents labelled as $correct = 1$ in the qrels and as "unreliable", the ones labelled as $correct = 0$; and second, considering as "reliable" the ones labelled as $correct = 1$ and $credible = 1$, and the remaining ones as "unreliable". The latter labelling strategy yielded the best results in preliminary experiments (e.g., training with 2019 decision track data and testing with 2020 TREC health misinformation data) and, thus, we adopted this method for the submitted runs.

We also experimented with the following two alternatives: i) feeding the model with the unmodified "description" field upfront the passage (run citius.r8) or ii) feeding the model with a derived correct sentence obtained from the "description" and "stance" fields (run citius.r7). See Section 3.5.1 for more details.

A T5-base model was fine-tuned with a constant learning rate of 3×10^{-4} for a variable number of iterations depending on the dataset size and with batches of size 8. We performed 2 training epochs and selected a max length of 512 input tokens.

At prediction time, a softmax function was applied over the "reliable" and "unreliable" tokens. Then, documents were re-ranked according to the probabilities assigned to the "reliable" token.

3.3.2 Unsupervised approach. An unsupervised approach based on sentence similarity was also included into our system. To that end, a derived correct sentence obtained from the "description" and "stance" fields (see Section 3.5.1) was encoded and compared with each sentence in the most relevant passage of the document.

This encoding was carried out with Sentence BERT models [12]. Previous studies have shown that these models perform much better than traditional BERT models for sentence similarity tasks [13].

The cosine similarity measure is applied on the obtained embeddings and used as ranking value (its mean over all passage sentences). An example is shown in Figure 3.

3.4 Rank Fusion

As a final step, our system allows to combine different scores from the previous stages (document ranking, passage re-ranking and passage reliability estimation) to generate a final ranking. We followed two unsupervised rank fusion strategies:

- CombSUM [14], which is a score-based technique that sums the scores that the document has in the ranked lists. In our case, the scores were first normalised[4].
- Borda Count [15, 16], which is a rank-based technique that implements a voting scheme. Each document gets votes from each ranked list, and these votes are added. The number of votes depends on the document's position in the list.

[3] https://github.com/castorini/pygaggle

[4] We normalise the scores by dividing by the maximum value for each topic

3.5 NLP improvements

3.5.1 NLP syntactic parser. The initial re-ranking approach explained in Section 3.2 comes with the power of modern Transformer-based architectures but also with their inherent caveats. In this task, it is particularly relevant their lack of consistency: slight changes in the input query's formulation may greatly affect the result of the model, as opposed to traditional bag-of-words models.

The model should rank taking into account the "description" and "stance" fields present in the topics, as seen in Figure 1. Making these inputs Transformer-friendly may be tricky and two configurations were tested: using the query as it is (without the stance) or paraphrasing the description into an affirmative or negative sentence depending on the stance (equivalent to generating a correct statement). We observed that the second method performed significantly better and boosted MonoT5's performance (in experiments with 2020 data). This was confirmed in the results of our participation this year (see Section 5).

To make such transformation, an algorithm built upon a syntactic constituency parser was used. The chosen parser (CRF Constituency Parser based on RoBERTa [17]) is state-of-the-art for this task. Faster models yielding lower performance may be desirable depending on the application. Using this parser, we were able to automatically transform the question into an affirmative or negative sentence, inserting a negation depending on the stance. For example, the generated correct sentence from Figure 1 would be "*Dexamethasone is a good treatment for croup*".

The same strategy was applied to generate correct sentences for the unsupervised approach of the passage reliability prediction explained in Section 3.3.2.

3.5.2 Paraphrasing technology. The unsupervised method from Section 3.3.2 was observed to have substantial performance improvements by manually making small variations in the queries. For example, instead of "*Dexamethasone is a good treatment for croup*", equivalent inputs may be "*Dexamethasone can be used to treat croup*" or "*Dexamethasone may cure croup*".

To introduce this approach into the pipeline, a T5-based automatic paraphraser was built, based on Ceshine Lee's available models on Huggingface[5]. However, these models performed badly due to the medical vocabulary present in the majority of queries. To address this issue, a vanilla T5-base was first pretrained using a standard unsupervised masking procedure with COVID-19 related research papers (CORD-19 dataset [18]). Code is available for this pretraining[6] as well as the fine-tuned model[7].

The model had now seen lots of medical terms during pretraining. Hence, applying the general non-medical paraphrasing fine-tuning resulted in excellent paraphrasing performance. It also shows the importance of pretraining in large Transformer models, allowing us to accomplish a task for which there was no specific data (medical sentence paraphrasing).

Since this technology is introduced into the unsupervised approach in Section 3.3.2, which already uses SentenceBERT for passage-query alignment, we only use the two most dissimilar paraphrases to the query. Note that under the cosine metric for sentence similarity, comparing a similar paraphrase with a sentence of a passage instead of the query will not make any difference.

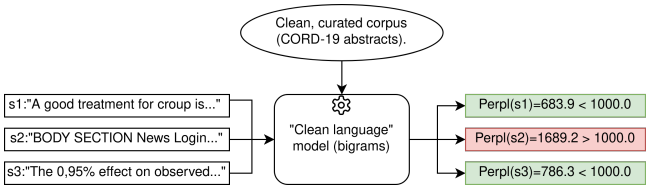

Figure 4: Perplexity model for boilerplate removal.

3.5.3 Boilerplate removal. A cleaning step was also introduced to improve MonoT5's performance. Passages are already cleaner than the original documents, but some of them have HTML boilerplate and other forms of noise included.

Language models use perplexity as a measure of how well the underlying probability distribution fits the observed sentence. Perplexity is computed as:

$$Perpl(w_1, w_2, \ldots, w_n) := 2^{\frac{-\sum_{i=1}^{n} \log_2 p(w_i)}{n}} \quad (1)$$

It is often used as an evaluation metric for a model, but it can also be used to measure the likelihood of a given sentence under the model. By setting the perplexity limit to consider a sentence "unlikely" really high, the model can be used to discard the noisiest and less plausible sentences without affecting the rest of the text (Figure 4).

For this task, there is no need for a complex model, since it is meant to just distinguish between correctly formed phrases and notorious boilerplate. We chose a simple bigram probabilistic model, which is both fast and adequate for perplexity computation.

The training corpus for the perplexity models needs to be a clean, well-formed set of text. To this end, we selected the set of abstracts from the CORD-19 research papers, as a curated source of training data for the model.

4 RUNS

Taking all of this into account, we generated and submitted 10 different runs using our multistage retrieval system:

- **citius.r1**: this run consisted of an initial document level BM25 search plus a MonoT5 passage re-ranking of the top 100 retrieved documents using the correct sentence derived from the "description" and "stance" fields for each topic (see Section 3.5.1).
- **citius.r2**: initial doc-level BM25 search plus a passage re-ranking of the top 100 retrieved documents based on the CombSUM fusion of three scores: BM25 score, MonoT5 most relevant passage score using the "description" field, and passage reliability score obtained with the unsupervised approach (RoBERTa Large STSB model [8]).
- **citius.r3**: same as citius.r2, but using RoBERTa Base STSB v2 model instead [9].

[5]https://huggingface.co/ceshine/t5-paraphrase-paws-msrp-opinosis
[6]https://github.com/manueldeprada/Pretraining-T5-PyTorch-Lightning
[7]https://huggingface.co/manueldeprada/t5-cord19-paraphrase-paws-msrp-opinosis
[8]https://huggingface.co/sentence-transformers/roberta-large-nli-stsb-mean-tokens
[9]https://huggingface.co/sentence-transformers/stsb-roberta-base-v2

- **citius.r4**: same as citius.r2, but including paraphrases for the unsupervised strategy of estimating passage reliability (see Section 3.5.2).
- **citius.r5**: same as citius.r3, but including paraphrases for the unsupervised strategy of estimating passage reliability (see Section 3.5.2).
- **citius.r6**: same as citius.r2, but removing passage sentences with a *perplexity* $> 15K$ prior to the sentence similarity comparison (see Section 3.5.3).
- **citius.r7**: initial doc-level BM25 search plus a passage re-ranking of the top 100 retrieved documents based on the Borda fusion of three scores: BM25 score, MonoT5 most relevant passage score, and passage reliability score obtained with the supervised approach (section 3.3.1) using the correct sentence as input along with the passage.
- **citius.r8**: same as citius.r7, but using the topic's description unmodified instead of the derived correct sentence along with the passage.
- **citius.r9**: same as citius.r2, but MonoT5 uses the correct sentence instead of the "description" field alone to estimate the most relevant passage.
- **citius.r10**: same as citius.r3, but MonoT5 uses the correct sentence instead of the "description" field alone to estimate the most relevant passage.

5 RESULTS

The track organisers distinguished between systems that only made use of the query (automatic runs) and those that utilised other fields of the topics (manual runs). In our case, most solutions were considered as manual. In Table 1, the results for our manual runs are reported, meanwhile, in Table 2 the results for the automatic runs are shown. The track organisers also provided us with a baseline BM25 run (first row). As expected, manual runs, which employ the stance field, perform better than automatic runs.

citius.r1 is the most promising automatic run. In general, it appears that using the correct sentence in the passage re-ranking stage (citius.r1, citius.r9 and citius.r10) enhances performance. However, it seems that including other signals within our ranking process worsens performance.

On the other hand, if we take a look at the runs that use MonoT5 with the description alone, two of them stand out: citius.r3 and citius.r5. These used the RoBERTa Base STSB v2 model for sentence similarity, and the second one also included paraphrasing. In terms of the competition, we ended ranked as the third-best team, being our best automatic solution citius.r1.

Runs	Help	Harm	Help - Harm
citius.r1	0,219	0,123	0,096
citius.r9	0,203	0,143	0,060
citius.r10	0,196	0,153	0,042
citius.r5	0,194	0,159	0,035
citius.r3	0,196	0,161	0,035
citius.r2	0,188	0,166	0,022
citius.r4	0,185	0,165	0,021
citius.r6	0,185	0,166	0,019
citius.r7	0,134	0,128	0,006

Table 1: Our results for the AdHoc Retrieval task (manual runs).

Runs	Help	Harm	Help - Harm
Baseline BM25	0,122	0,144	-0,022
citius.r8	0,163	0,155	0,008

Table 2: Our results for the AdHoc Retrieval task (automatic runs).

6 CONCLUSIONS AND FUTURE WORK

In these working notes, an entire multistage retrieval architecture is proposed and applied to health misinformation detection. This approach represents the foundation of our participation in the TREC 2021 Health Misinformation Track.

We have submitted ten different runs, and some interesting conclusions could be extracted. Passage re-ranking algorithms work better with a derived correct sentence rather than with the original query alone (description field). This suggests that biasing the retrieval process towards those passages that are somehow related to correct sentences is effective, and such an approach is more solid than other alternatives which focus on passages that are merely on topic.

Another important conclusion is that unsupervised strategies for passage reliability estimation widely outperformed their supervised counterparts.

We intend to further analyse these two findings and continue improving our system. Our ultimate goal is to identify which other signals or features could help to better detect health misinformation.

For future editions of this challenge, we want to further advance in automatic solutions, which do not require knowledge of the actual stance. In our view, this instance of the task is more realistic in standard web search scenarios. Manual solutions, instead, are better suited for certain retrieval tasks where the correctness of the treatment is known by the user of the system (e.g., a social media moderator who wants to remove harmful contents from the platform).

ACKNOWLEDGMENTS

The authors thank the support obtained from: i) project RTI2018-093336-B-C21 (Ministerio de Ciencia e Innovación, Agencia Estatal de Investigación & ERDF), ii) project PLEC2021-007662 (MCIN/AEI/10.13039/501100011033, Ministerio de Ciencia e Innovación, Agencia Estatal de Investigación, Plan de Recuperación, Transformación y Resiliencia, Unión Europea-Next GenerationEU), and iii) Consellería de Educación, Universidade e Formación Profesional (accreditation 2019-2022 ED431G-2019/04, ED431C 2018/29) and the European Regional Development Fund, which acknowledges the CiTIUS-Research Center in Intelligent Technologies of the University of Santiago de Compostela as a Research Center of the Galician University System.

REFERENCES

[1] Susannah Fox. *Health topics: 80% of internet users look for health information online.* Pew Internet & American Life Project, 2011.
[2] Gunther Eysenbach. Infodemiology: The epidemiology of (mis) information. *The American journal of medicine*, 113(9):763–765, 2002.

[3] Frances A Pogacar, Amira Ghenai, Mark D Smucker, and Charles LA Clarke. The positive and negative influence of search results on people's decisions about the efficacy of medical treatments. In *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval*, pages 209–216, 2017.

[4] Nima Asadi and Jimmy Lin. Fast candidate generation for two-phase document ranking: Postings list intersection with bloom filters. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 2419–2422, 2012.

[5] Nima Asadi and Jimmy Lin. Document vector representations for feature extraction in multi-stage document ranking. *Information retrieval*, 16(6):747–768, 2013.

[6] Nima Asadi and Jimmy Lin. Effectiveness/efficiency tradeoffs for candidate generation in multi-stage retrieval architectures. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 997–1000, 2013.

[7] Ronak Pradeep, Xueguang Ma, Xinyu Zhang, Hang Cui, Ruizhou Xu, Rodrigo Nogueira, and Jimmy Lin. H2oloo at trec 2020: When all you got is a hammer... deep learning, health misinformation, and precision medicine. *Corpus*, 5(d3):d2.

[8] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv e-prints*, 2019.

[9] Peilin Yang, Hui Fang, and Jimmy Lin. Anserini: Reproducible ranking baselines using lucene. *Journal of Data and Information Quality (JDIQ)*, 10(4):1–20, 2018.

[10] Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. Document ranking with a pretrained sequence-to-sequence model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 708–718, Online, November 2020. Association for Computational Linguistics.

[11] Marcos Fernández-Pichel, David E. Losada, Juan C. Pichel, , and David Elsweiler. CiTIUS at the TREC 2020 Health Misinformation Track. In *The Twenty-Ninth Text REtrieval Conference Proceedings (TREC 2020), NIST Special Publication 1266*, 2020.

[12] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019.

[13] Pablo Gamallo, Manuel de Prada Corral, and Marcos Garcia. Comparing dependency-based compositional models with contextualized word embeddings. In *ICAART (2)*, pages 1258–1265, 2021.

[14] Edward A Fox and Joseph A Shaw. Combination of multiple searches. *NIST special publication SP*, 243, 1994.

[15] JC de Borda. Mémoire sur les élections au scrutin. *Histoire de l'Academie Royale des Sciences pour 1781 (Paris, 1784)*, 1784.

[16] Javed A Aslam and Mark Montague. Models for metasearch. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 276–284, 2001.

[17] Yu Zhang, Houquan Zhou, and Zhenghua Li. Fast and accurate neural CRF constituency parsing. In *Proceedings of IJCAI*, pages 4046–4053, 2020.

[18] Lucy Lu Wang, Kyle Lo, Yoganand Chandrasekhar, Russell Reas, Jiangjiang Yang, Darrin Eide, Kathryn Funk, Rodney Kinney, Ziyang Liu, William Merrill, et al. Cord-19: The covid-19 open research dataset. *ArXiv*, 2020.