{Team Name} at eHealth-KD 2020
{Long Title}

First Author[1][0000−1111−2222−3333], Second Author[2,3][1111−2222−3333−4444], and Third Author[3][2222−−3333−4444−5555]

[1] Princeton University, Princeton NJ 08544, USA
[2] Springer Heidelberg, Tiergartenstr. 17, 69121 Heidelberg, Germany
lncs@springer.com
http://www.springer.com/gp/computer-science/lncs
[3] ABC Institute, Rupert-Karls-University Heidelberg, Heidelberg, Germany
{abc,lncs}@uni-heidelberg.de

Abstract. The abstract should briefly summarize the contents of the paper in 15–250 words.

Keywords: First keyword · Second keyword · Another keyword.

General Details

- Articles must be written in English.
- The minimum length of the paper should be 5 (mandatory minimum) and up to 10 pages plus references.
- Please respect the title format, where *Team Name* is the officially published team identifier in the eHealth-KD 2020 website, and use *Long Title* for additional details. Contact the challenge organizers for any changes to the team names.

NOTE: This section is not meant to be part of the final version of your paper.

1 Introduction

Write down a general overview of your system. This might include:

- Motivation for choosing the selected architectures in the context of the challenge.
- Citations to any external resources or strategies used by your system.

Do not focus on describing the task and/or the corpus. Instead, include a citation to the Overview paper, and at most provide a very short introduction to the challenge if you consider it relevant. A preliminary citation is provided in this template [**?**], which will be updated in due time.

2 System Description

Describe the architecture of your system in a concise and precise manner, such that other participants might be able to reproduce your work. Make sure to include the following information:

- Your system's architecture.
  - For example, if your system is based on deep learning techniques, mention the corresponding layers and other components.
  - Hyperparameters (e.g., layer sizes, dropout rates).
- Input handling.
  - Sentence tokenization.
  - Token representation and encoding.
- Output handling.
  - How does the entity extraction task translate to your system (e.g., sequence labeling problem, takes entities overlaps into consideration)?
  - How does the relation extraction task translate to your system (e.g, pairwise queries, focus on a single entity)?
- System training.
  - In what infrastructure your system was trained?
  - Which collections (i.e. train, dev, ensembled corpora) of the dataset you used for training, validation, etc.?

Consider the following questions while writing your working notes:

- What is the general approach your system fits best into?
  - Deep learning based?
  - Classical ML algorithms?
  - Handcrafted rules?
  - NLP features?
  - Other?
- Does your system solves task A and B jointly? Or with several independent models? Do they share any layers?
- Does your system use pretrained word embeddings? Custom ones?
  - Which one?
  - Trained on a general domain corpora? Medicine related?
- Does your system use pretrained contextual embeddings such as BERT?
  - Which one?
  - How you incorporate it? Fine-tuning? Pre-computed features?
- Does your system used additional syntactic features?
  - Which ones?
  - POS-tag information?
  - Dependency parsing information?
  - char level representations?
- Does your system make use of the additional 3000 automatically annotated sentences from Medline that were provided for further training?

    - Which ones of them were used? All of them? Only the ones with the highest agreement?
- Does your system extend the training data available with any other extra resources?
    - Which one?
    - Does it use sentences from the first edition of the competition?
- Does your system use any other type of external knowledge?
    - Which one?
    - How do you think it contributes to your system?
- Does your system applies attention-based techniques to solve any task?
    - Which one?
    - How?
- Does your system uses any strategy for additional performance boosting, such as ensemble methods?
    - Which one?
    - What are the relevant parameters and overall details?
- Do you perform any type of hyperparameter tuning or architecture search?
    - Do you use an external tool (e.g., *AutoSklearn*, *AutoKeras*), or a custom solution?
    - How did you split the training data for cross-validation purposes?
    - What are the relevant parameters, execution time, resources, etc.?
- About the transfer learning evaluation scenario (scenario 4), does your system take any additional considerations into account?
    - Which ones?
    - Were they taken into account the same way for all scenarios? Or does a particular data or architecture was used to solve each particular scenario?

The purpose of these questions is to highlight the kind of details were believe the readers will be most interested. Do not take these questions literally nor organize the section as an explicit answer to these items, but rather use them to guide your overall system description, while organizing the text in a coherent narrative. If the answer to any of the previous questions is simply "no", or it is completely unrelated to your approach, then do not mention those elements. Likewise, include any additional details that you consider relevant for understanding and contextualizing your contributions.

3 Results

Report the performance achieved by your system in each run and scenario as officially published. If your team developed one or more systems that were not submitted to the challenge, feel free to include them in this section, but always noticing that there were not part of the officially evaluated runs. Include any tables and figures that you consider relevant.

You can use any officially announced evaluation statistics to compare your results with the ones of other participants, and you can design and discuss other comparison metrics to address the issues you consider most relevant with respect to your system.

4 Discussion

Though not mandatory, we encourage you to discuss the main insights that can be derived from the performance of each one of your runs. You can include additional experimentation, analysis of the impact of hyperparameters, analysis of feature relevance, etc.

Remember that the most important results of this challenge are not the F1 metrics per-se, but rather any interesting findings and insights that help advance the state-of-the-art.

5 Conclusions

Share your final conclusions on your systems and any future work recommendations.