Literature Review

*Intro*

As previously noted, a Bayesian optimiser is a common strategy used to minimise the amount of feedback required whilst refining summaries. To combat this lack of user data, Simpson, Gao and Gurevych explore an interactive text ranking approach whereby the user compares two candidate summaries and selects their preferred option \cite{Simpson19}.

The common approach to minimise the number of selections made by the user is through active learning \cite{Simpson19}. Active learning learns a model by obtaining labels iteratively: the model is trained on acquired labels and uses an acquisition function to quantify the value of asking the user about a propositioned pair of candidates \cite{Simpson19}. Simpson, Gao and Gurevych introduce Gaussian process preference learning (GPPL) as a central learner with which Bayesian optimisation strategies can be used for interactive text ranking \cite{Simpson19}. GPPL is the preferred preference learner since alternative approaches require costly sampling, limiting its usage in an interactive setting.

The paper documents the experimentation process undergone by four different strategies for GPPL \cite{Simpson19}:

NON-BO

BO

(bullet each AF).

With regards to the two Bayesian optimisation strategies, the paper highlights the added considerations that these alternatives allow. Since IMP uses a greedy approach to choose the pair, it promotes the exploitation of promising candidates over the exploration of unknowns \cite{Simpson19}. Latter experimentation finds that this preference for exploiting top candidates allows for better learner performance \cite{Simpson19}. Instead, TP is less greedy than IMP as it allows more exploration; however, there is still a greater emphasis on summaries with potentially high scores in comparison to non-Bayesian alternatives \cite{Simpson19}.

It was found that the Bayesian optimisation strategies outperformed the alternative strategies. Considering the two Bayesian optimisation strategies, the expected improvement (IMP) strategy for GPPL performed better than Thompson sampling (TP) when tasked to optimise the proposed best solution \cite{Simpson19}. It is thought that TP was limited by the number of user decisions required since random sampling is a large element of the algorithm.

With a large emphasis on the experimental study, this paper has little focus on the limiting properties of each strategy. Discussions of the differences between the proposed approaches and existing solutions are only in the context of how these new strategies differ from the existing ones. There is no deeper comment on the implications or assumptions that go along with these changes.

Moreover, there is little reflection given to the data that will be summarised. There are a lot of variabilities when it comes to data quality and some commentary this is missing from the strategy comparison is a discussion on the robustness of each approach.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

However, as presented by Xu, Ma, Nallapati and Xiang, deep learning techniques have become popular to solve passage ranking tasks \cite{Xu19}. A weak supervision framework is proposed such that a weak learner replaces human annotators with a BERT-based supervised learning model trained on these labels \cite{Xu19}. The weak supervision technique is used since there are high costs attached to hand-labelled ranking \cite{Xu19}.

The proposed BERT scoring model estimates the relevance of a set of passages ${p\_i}$ to a given query $q\_i$ \cite{Xu19}, quantified using \emph{hinge loss}. The relevance score is generated by taking the final hidden state for the first token of the input and feeding it into a Multi-layer Perceptron (MLP) module \cite{Xu19}.

The weak supervision aspect to the proposed framework consists of three steps \cite{Xu19}. We first want to define labelling functions to generate noisy labels. This is done by reducing the labelling problem to a simpler task; labelling whether the candidate passage is strongly related to the query or not. We build a collection of query-passage pairs labelled as positive, negative or neutral \cite{Xu19}. The passages are ranked for each query based on similarity scores which form our noisy labels. Examples of such functions, discussed in this paper, include the BM25 Score, the TF-IDF Score, the cosine similarity of universal embedding representation and the cosine similarity of the last hidden layer activation of the pre-trained BERT model \cite{Xu19}.

The final major step that is

\item Aggregate noisy labels

\item Generate triplet training instances.

\end{enumerate}

* Are these labels ranks? Or not?
* Why hinge loss?
* Why four scoring functions? Is more better?
* Why is GM approach chosen over majority voting?
* This paper provides a deeper understanding of how the framework works and fits together, but provides very little with regards to