1 Methods

1.1 Data

From the Sysrev data I have selected a few environment-related projects focusing on the include/exclude question.
The full titles of the projects considered are:

1. EntoGEM: a systematic map of global insect population and biodiversity trends
2. Climate change impacts on human health
3. Fragmentation effects on North American mammal species systematic review and meta-analysis.

For this initial exploration, I have only considered texts for which all manual reviewers agree on the category assignment, e.g., all reviewers categorise the article as relevant. All included titles are initially processed using gensim's (Řehůřek & Sojka, 2010) simple_preprocess function.

Table 1: **Data summary**

Project	**N texts**	**N relevant**	**N irrelevant**
1. EntoGEM	5596	2459	3137
2. Climate:health	1775	1219	556
3. Frag:mammals	1655	352	1303

1.2 Potential biases in texts

To identify potential issues with the underlying corpus, I apply logistic regression models, identifying the relationship between terms/words and article relevance. Prior to modelling, titles are further processed using gensim (Řehůřek & Sojka, 2010) to remove stopwords and words with fewer than 3 characters, with the remaining words being stemmed using a Porter stemmer. Texts are then transformed to a term-frequency matrix, to which a logistic regression model is fitted (Pedregosa et al., 2011). Term weights are extracted from the model and used to identify those terms with the strongest discriminatory effect, where positive weights are associated with relevant articles and *vice versa*.

1.3 Potential biases in models

To assess potential biases in text-classification models, I have used the LIME (Local Interpretable Model-agnostic Explanations) framework (Ribeiro, Singh, & Guestrin, 2016), applying it to some example text-classifiers. Briefly, LIME uses local, linear approximations of the model to identify the parts of the input (text) that have the strongest influence on the prediction (Ribeiro et al., 2016). Again, in the examples presented, positive weights are linked to relevance whereas negative weights are linked to irrelevance.

The classification models I've used to demonstrate LIME are fully-connected neural networks built in keras (Chollet et al., 2015) and incorporating Google's universal sentence encoder (Cer et al., 2018). The Google encoder converts texts to a numeric vector with length 512 which then passes through a neural network with two hidden layers of 1024 nodes. Each layer is followed by, 'relu' activation, 'l2' regularisation and dropout of 0.3, before a final softmax layer returns the predicted relevance. During training the loss is defined as binary crossentropy and the 'adam' optimiser is used.
A fairly standard classification model.

For each project, models were trained using an equal number of relevant and irrelevant texts (obtained via undersampling the majority class). I then randomly sampled 100 relevant and 100 negative texts to pass to LIME, along with the trained model. - alternatively, selecting those texts that are predicted to be most relevant/irrelevant may give better information. Weights are averaged (where necessary) to provide a single LIME score per word.

2 Results

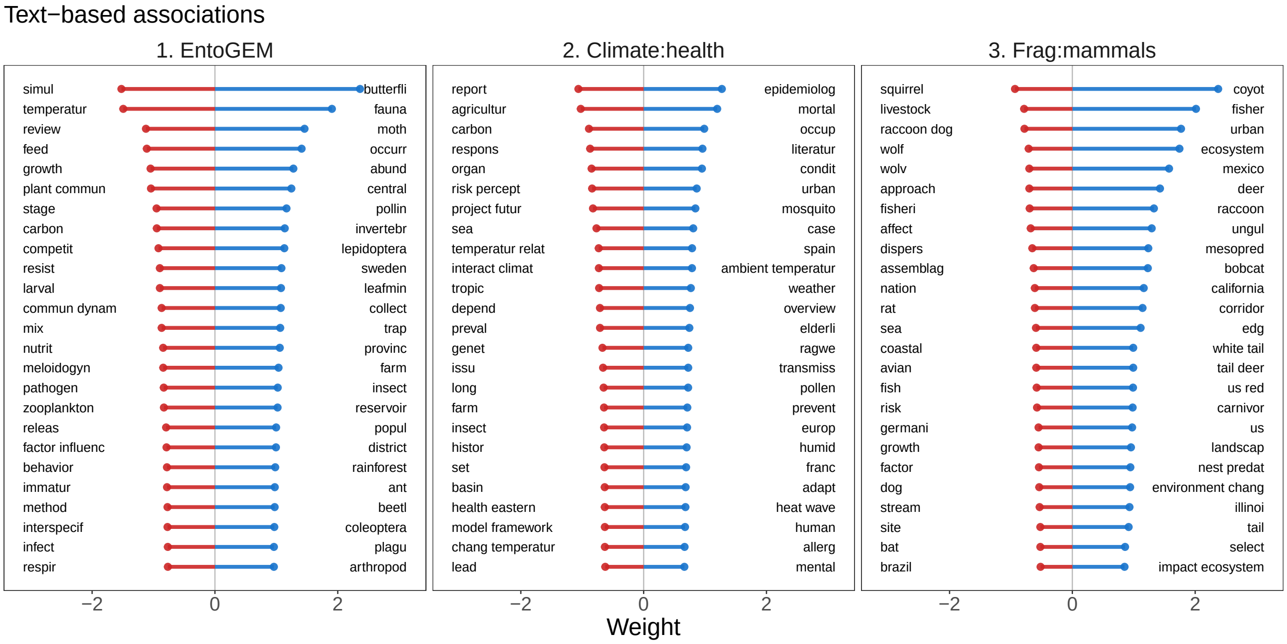

Figure 1: **Terms and their association with relevant (blue) and irrelevant (red) labels.**

Clearly, many of the positively weighted terms in Fig 1 are linked to the respective topics of interest. For example, 'butterfli', 'moth' and 'abund' for EntoGEM and 'epidemiolog' and 'allerg' for the Climate:human-health project. Despite this, nuisance terms - terms associated with relevance but not likely to be of interest - also appear. Examples of these include 'sweden' and 'spain'. Furthermore, some negatively weighted words could actually be associated with topics of interest, especially for the Fragmentation:mammals project; e.g., 'squirrel' and 'wolf'.

The model-based results presented in Fig 2 are purely representative of the style of output that can be obtained and have been generated from fairly baseline, neural network classifiers. Interestingly, this analysis suggests that even greater importance is associated with geographic key-words. If this persists in other classification models it could present a potential source of bias.

By providing users with this information, they could manually assess which terms they are most concerned about and then 'mask' them from classifier construction/prediction to create a 'less biased/partially de-biased' result.

Graph design etc. could easily be changed to fit with the Sysrev theme, but hopefully this provides an overview of how these approaches could be useful.

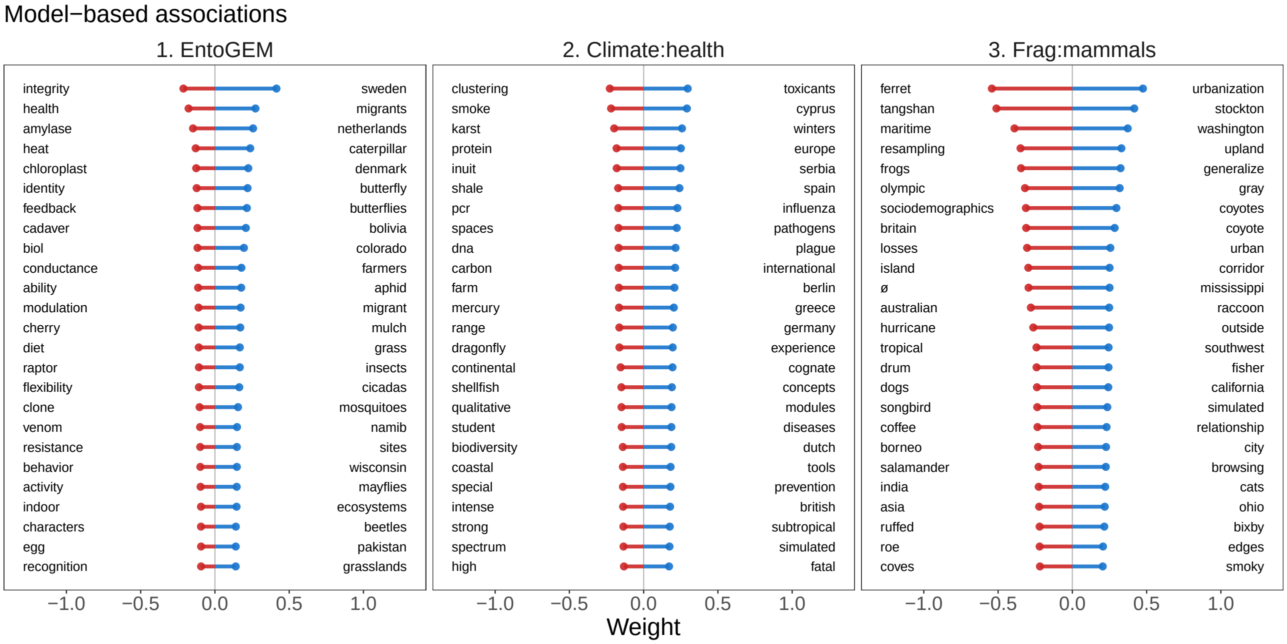

Figure 2: **Association between terms and model predicted relevance (blue) and irrelevance (red).**

References

Cer, D., Yang, Y., Kong, S.-y., Hua, N., Limtiaco, N., John, R. S., … Kurzweil, R. (2018). Universal Sentence Encoder. *arXiv preprint arXiv:1803.11175*.

Chollet, F., et al. (2015). *Keras.* `https://keras.io`.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Řehůřek, R., & Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks* (pp. 45–50). Valletta, Malta: ELRA.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1135–1144).