2021 EAGE ANNUAL
82ND CONFERENCE & EXHIBITION
AMSTERDAM | THE NETHERLANDS

EXTRACTING KNOWLEDGE WITH NLP FROM MASSIVE GEOLOGICAL DOCUMENTS

C.H. Lun[1], T. Hewitt[1], S. Hou[1]

[1] CGG

Summary

There have been many advances in natural language processing in recent years but most of the work have been focused on texts from a general domain or medicine and so datasets in the geology domain are sadly lacking. We demonstrate how existing taxonomy and geological texts can be used to address this issue and also show how named entity recognition and object detection can be used to retrieve information from a large number of documents.

Extracting Knowledge with NLP from Massive Geological Documents

Introduction

A lot of geological data is trapped inside documents such as reports and papers in unstructured prose. Integrating all data both new and legacy will allow us to build a holistic model of the subsurface and therefore reduce the risks in exploration and development. However, the volume of geological text accumulated over decades is huge and it is infeasible for domain experts to extract relevant information or even to classify each document manually in a reasonable time frame. In recent years there have been huge advancements in natural language processing (NLP). In this paper we show how the NLP task of named entity recognition (NER) can be used to automatically identify geological terms which powers information retrieval and thus aids geologists to explore the vast document landscape. It is also a first step towards transforming unstructured data into structured data to feed into a holistic subsurface model. While there are public datasets available for NER for domains like business and medicine, the same cannot be said for geology, but we shall show how existing taxonomy and geological texts can be used to automatically create a dataset without the need for time consuming manual annotation.

Named Entity Recognition and Language Models

NER is a task concerned with identifying spans of text which constitute a named entity. A named entity or entity is roughly speaking anything that can be referred to with a proper name like a person or location but can also include dates, times and numerical expressions such as measurements. In this work we are concerned with genus species names (e.g. Striatricolporites tenuissimus) and well names (e.g. Bok-1). We frame NER as a token classification task which involves predicting the label (in our case either genus species, well or none) of each token in the input text. What constitutes a token depends on the tokeniser used but they are often individual words or parts of a word.

State of the art NLP models are nowadays dominated by deep learning architectures. These models require each token of the input text to be represented as vectors. Early models uses for the vector fixed word embeddings such as word2vec (Mikolov et al., 2013) however these fail to express the difference in meaning of words under different contexts. With the introduction of transformer (Vaswani et al., 2017) based language models the role of word embeddings have been replaced by the contextual embeddings computed by such language models.

A language model is a model which estimates the probability distribution of a sequence of tokens. Depending on the model this probability can be factorised as products of conditional probability of the next token given the previous tokens or conditional probability of the masked token given the unmasked token in the sequence. Large transformer based language models like BERT (Devlin et al., 2018), XLNet (Yang et al., 2019) and GPT-3 (Brown et al., 2020) have demonstrated remarkable results on NER and many other NLP tasks such as question answering, text classification and text generation etc. The training of a language model on these tasks involves two stages: 1) unsupervised pretraining of the model on a large corpus. The exact task which the model is trained to perform varies with different models but all involve some form of text prediction given a context. 2) Training of the model on the downstream task like NER on an annotated but much smaller dataset.

The reason why transformer based language models work so well are two folds; one is that a transformer enables long distance parsing for example the processing of the first input token can take into account the last token in the input thus allow it to more effective learn features of the language such as grammar. Secondly, transformers enjoys nice theoretical properties; they are able to compute contextual embeddings of the input where each unique context is mapped to a unique vector (Lemma 6 of Yun et al., 2020) and they are also a universal approximator of continuous sequence to sequence functions on a compact domain (Theorem 3 of Yun et al., 2020). The latter is analogous to the classical universal approximation theorem for feed forward networks, see e.g. Cybenko, 1989.

NER Dataset Creation

Our method of creating a dataset for NER requires two input sources, a text corpus and a taxonomy. In this work we demonstrate with a taxonomy of genus species names and well names. The corpus we used consists of PDFs of well reports from different time periods. Depending on the PDF we extract the text from it using either OCR or retrieve its embedded text using a PDF parser.

The next step is to string match the taxonomy against the extracted text after some manual clean-up to remove erroneous entries. In addition to string matching we also perform rule based pattern matching to tag anything that are not in the taxonomy. For example, for genus species we tag any word appearing before "spp." and "spp." itself as an occurrence of a genus species and for well names an example is using regular expression to tag any occurrences of the pattern NAME-NUMBER like "Bok-1". While string matching and pattern matching does manage to find many entities in the text, it is however not complete since there may be spelling mistakes in the taxonomy or error in the OCR which prevents entities from being matched and due to many variations in naming conventions it is difficult to implement all possible patterns for matching. More importantly, a taxonomy and any rule-based system are static and cannot adapt automatically to new names and new naming conventions. All of this necessitates using NLP techniques which takes into account of the meaning of the input text to recognise entities.

In machine learning it is best practice to split a dataset into three parts: a training set to train the model on, a validation set for tuning the hyperparameters of the model and for determining when to stop training and finally a test to evaluate the performance of the model on unseen data. To ensure that the test set contains unseen data, which in our case means that the test set should contain predominately genus species names and well names that are not in the training or validation set, we create a graph of entity co-occurrences where two entities are said to co-occur if they appear in the same passage of text. The graph has each unique entity found in the text as nodes and there is an edge between two nodes if their corresponding entities co-occur.

With the graph constructed, its connected components are then computed. As nodes in one connected component does not have an edge with nodes in another connected component we assign connected components randomly to one of train, validation or test set to ensure that there are no overlaps in entities between sets. Furthermore, the splits are made in such a way that roughly 80% of the unique entities are in the training set, 10% in the validation set and 10% in the test set. In reality, many entities co-occur with one another so it is not possible to achieve the 8:1:1 ratio without overlaps therefore we minimize the overlap by selecting nodes with the fewest edges to be in the test and validation set.

Experiments

The language model we used in our experiments is DistilBERT (Sanh et al., 2020), a distilled version of BERT, which is faster to train and to make inference. Finding the best language model to perform NER is not the focus of this work. In fact with the dataset and training pipeline in place it is trivial to experiment with different language models.

We initialise DistilBERT with publicly available weights trained on a general domain corpus and then finetune it on our geological corpus. The reason for doing this is that the usage of words in geology are often very different from texts from other domains. For example, technical terms like "biostratigraphy" appears more often in geological texts than in other domains. Therefore, the language model needs to be trained further for text prediction on the geological corpus so that the probability distribution it estimates reflects the input text in the downstream tasks. After finetuning we train the model on the generated training dataset described in the previous section to perform NER and evaluate it on the test set. See Figure 1 for examples of predictions.

Storing the Predictions

We keep track of the document, the page and which title, paragraph, list, table or figure (referred to as page objects from now on) each predicted entity is found since co-occurrence of entities in the same page object may indicate that its text expresses a fact involving the entities. Keeping track of where

entities appear enables a search functionality where a user can retrieve all segments of text containing the entities being searched for and thus aiding the user to understand the documents.

. Some Geli Khana beds show an abundance of the macrofossils Claraia sp . GENUSSPECIES , Costatoria sp . GENUSSPECIES and ? Bakevellia sp . GENUSSPECIES_NEW (found at K5 - 133) , indicative of a typical Induan assemblage . The medium to well rounded , moderately sorted sandstone clasts at K5 - 134 could have the same origin as those described in dolomite filled vugs occur at K5 - 107 . The topmost part of this calcareous section features a medium to dark grey bioclastic packstone , containing a characteristic Lower Triassic Tethyan fauna , including Costatoria ? costata GENUSSPECIES_NEW and Claraia sp . GENUSSPECIES (K5 - 109) . This member was not seen in the Khabour River Valley due to

breached traps include UK 29/10a- 2 WELL_NEW , 22/30a-1 WELL and 22/14b-3 WELL (Gaarenstroom et al . , 1993) . The hydrocarbons are likely to have migrated into the Chalk in these areas . Hydrocarbon shows throughout long intervals of the Chalk section in well 21/20a-1 WELL (dry hole)

Figure 1 *Example of predictions on the test set. Labels with the suffix "_NEW" indicates that the entity is neither in the taxonomy nor found by pattern matching.*

The bounding boxes of every page object is obtained using an object detection model called Faster RCNN (Ren et al., 2015) trained on a public dataset and then finetuned on our own manually annotated dataset which better reflects the style and layout of geological documents. At inference time each page of a PDF is rasterised into an image which is then fed into Faster RCNN, see Figure 2.

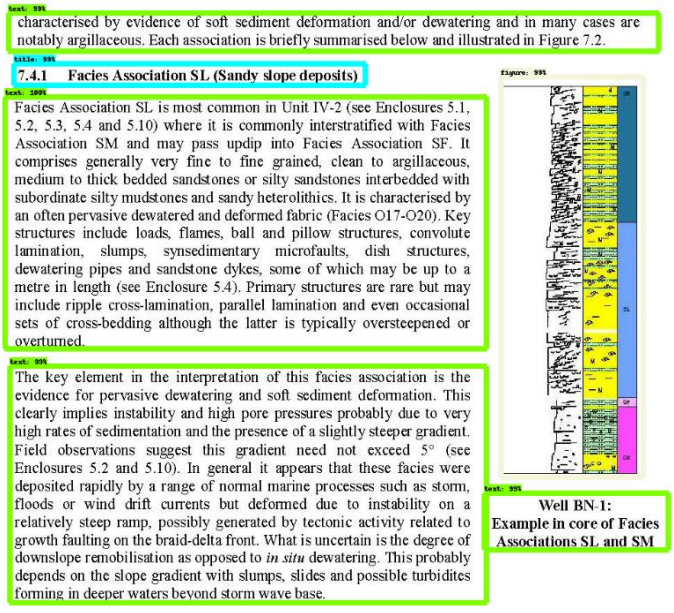

Figure 1 *Example of results of page object detection. blue – title, green – paragraphs, grey – figures.*

All results are saved in a graph database (Figure 3). Documents, pages, page objects, entity mentions and entities are represented as nodes in the graph and edges between nodes indicate relationships such as a document contains a page or a paragraph mentions a certain entity. A graph database is more suited for storing such hierarchical relationship since retrieving a node that shares an edge with another node is quick while with a relational database where documents, pages, entities etc. are stored in separate tables it would require many expensive joins to find out which document, which page and which page

object an entity is mentioned in. Another advantage is that a graph is a much more intuitive way to visualise data which allows domain experts to more easily explore the predicted results.

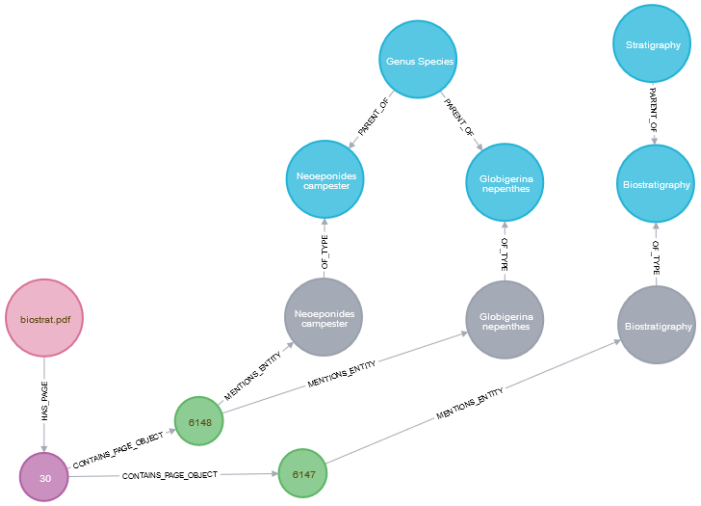

Figure 3 *Part of the graph database. Pink nodes represents documents, purple – page, green – paragraphs, grey – entity mentions and blue – entities.*

Conclusion

We have discussed how existing documents and taxonomy can be leveraged to create an annotated dataset for NER with minimal human input and therefore addressing the issue of a lack of NER dataset in the geology domain. We also showed how language models and object detection can be used to generate the results required to power an information retrieval system. Key to all of this is our comprehensive taxonomy which enables the generation of a good training dataset and our large geology corpus for effective finetuning of the language model.

Acknowledgements

The authors thank CGG for permission to publish. Special thanks to our colleagues in CGG Data Hub.

References

Brown et al. [2020] Language Models are Few-Shot Learners *arXiv preprint arXiv:2005.14165v4*
Cybenko, G. [1989] Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems* **2,** 303–314
Devlin, J., Chang, M., Lee, K., & Toutanova, K. [2019]. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *NAACL-HLT.*
Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., Dean, J. [2013] Distributed Representations of Words and Phrases and their Compositionality. *Advances in Neural Information Processing Systems*
Ren, S., He, K., Girshick, R., Sun, J. [2015] Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *Advances in Neural Information Processing Systems*
Sanh, V., Debut, L., Chaumond, J., Wolf, T. [2019] DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108v4*
Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., Polosukhin, I. [2017] Attention is All you Need. *Advances in Neural Information Processing Systems*
Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., Le, Q. [2019] XLNet: Generalized Autoregressive Pretraining for Language Understanding. *Advances in Neural Information Processing Systems*
Yun, C., Bhojanapalli, S., Rawat, A. S., Reddi, S., Kumar, S. [2020] Are Transformers universal approximators of sequence-to-sequence functions? *International Conference on Learning Representations*