

# Using a Knowledge Graph for Zoning Ordinance Question Answering

Micah Cochran and Seth Lewis

## PROBLEM

- In the United States, laws vary from one place to another.
- A large portion of local laws deal with land development.
- Currently, the best representation is a text document.
- Understanding laws takes time and effort.
- Many just want to know how the laws impact their project.

## PROPOSAL

### 1) A method to structure and query the land development law.

- Create a Knowledge Graph (KG) for land development law with specific set of RDF tags.
- This structure to the law into specific set of tags and that can be queried with SPARQL.
- This technology alone would benefit land developers and local governments.

### 2) A chatbot that answer questions about land law.

- SPARQL queries are too complicated for someone just trying to get a sense of how land law applies to their development.
- Leverage the Knowledge Graph to store information.
- Create an interface between questions and the information stored in the Knowledge Graph.
- Potentially, this would allow end users to be able to ask the chatbot questions and get relevant answers to those questions.
- BIG picture, this promotes democracy, people who are fully informed on a topic can make better decisions. Long term, those opinions can shape how new laws are created and updated.

## ZONING ORDINANCE EXCERPT

**§3.5. Nonconforming Uses and Buildings**

**3.5.1. Statement of Intent.** Within the districts established by this ordinance or amendments that may later be adopted, there exist lots, structures, and uses of land and structures, which were lawful when established, but which would be prohibited or restricted under the terms of this ordinance or future amendment. It is the intent of this ordinance to permit these nonconformities to continue until they are removed, but not to encourage their survival.

**3.5.2. Expansion.** Nonconformities shall not be enlarged upon, expanded or extended. Lots not meeting height requirements are exempt from this rule notwithstanding the rule for restoration found in Sec. 3.5.3.A below. In such instances, the building may be added onto at the same height of the nonconforming structure so long as the addition does not exceed 25% of the GFA of the nonconforming structure. Nonconformities shall not be used as grounds for adding other structures or uses prohibited elsewhere in the same district.

**3.5.3. Continuance.** A nonconforming use or building lawfully existing at the effective date of this ordinance may be continued, except as hereafter provided, although such use does not conform to the provisions of this ordinance.

A. Restoration after damages: Other than detached dwellings, no nonconforming building or structure which has been damaged by fire or other causes to the extent of more than fifty (50) percent of its replacement value prior to the time of such damage, shall be repaired or restored except in conformity with the provisions of this ordinance. However, for buildings or structures not damaged or destroyed beyond fifty (50) percent of its replacement value prior to the time of such damage, they may be repaired and used as before if repairs are initiated in twelve (12) months and completed within two (2) years of such damage.

Source: City of Athens, Alabama Zoning Ordinance a portion of Section 3.5.

## QUESTION TEMPLATES & GENERATION

- Training for Natural Language Processing needs questions, the more the better.
- The Knowledge Graph is as the source for the question generation.
- Templates were generated by the domain expert.
- This process generates about 1,000 questions.

## SPARQL

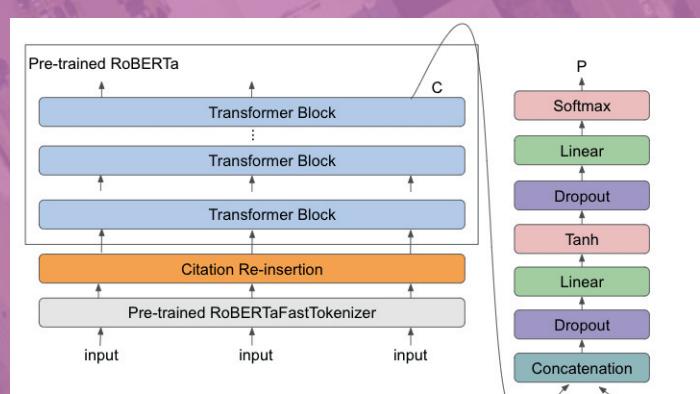
```
SELECT ?property_name ?property_label
WHERE {
  ?property_name a rdf:Property ;
    rdfs:label ?property_label .
}
Result:
?property_name = ':minLotWidth'
?property_label = 'minimum lot width'
...and more results...
```

SPARQL queries supply the variables to fill in the templates.

## PYTHON TEMPLATE CODE

```
tmplt_vars = {'regulation_text': 'minimum lot width',
              'zoning': 'R3a'}
q_template = string.Template(
    "What is the $regulation_text in the $zoning zoning district?")
q_template.substitute(tmplt_vars)
>>> "What is the minimum lot width in the R3a zoning district?"
```

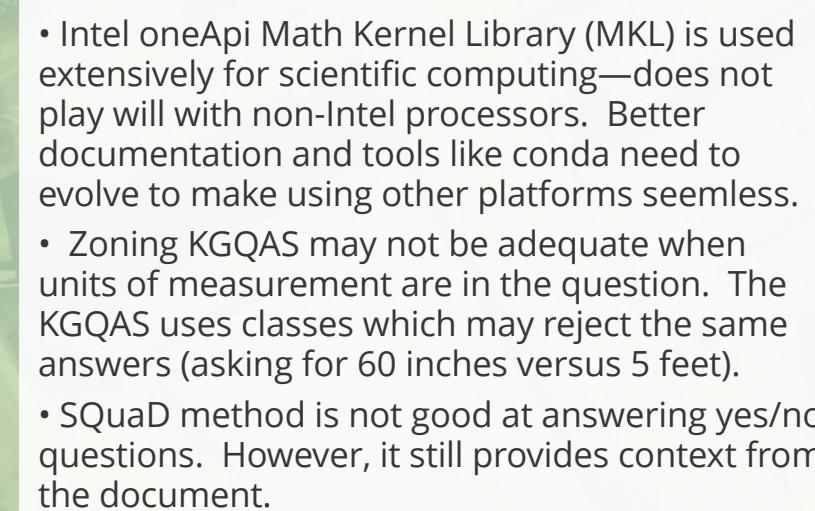
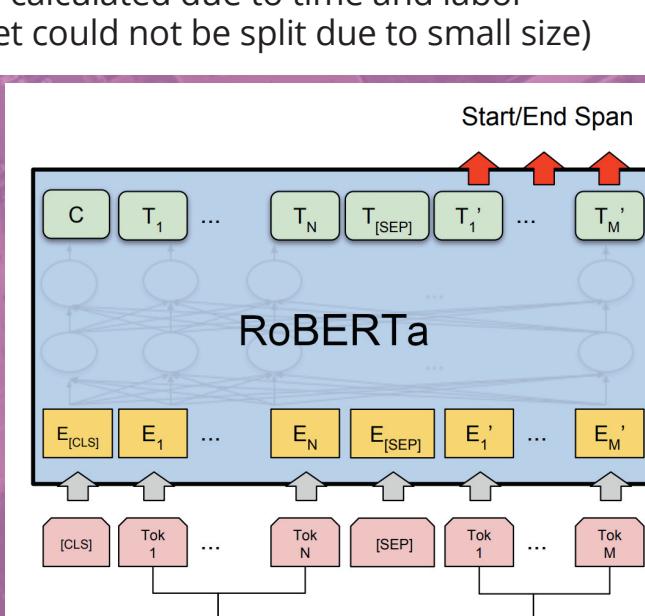
## METHOD 1: NAIVE CLASSIFICATION-BASED QUESTION ANSWERING



- Pretrained BERT and ROBERTA were selected and finetuned to contrast with SQuAD based system using pretrained ROBERTA base
- Training and testing data collectively contained only 48 unique answers but overall 1062 samples
- Answers used as classes in simplest possible QA system using transformers
- Classes were balanced due to the nature of the KG question generation system
- Naive approach, not generalizable, but gives good accuracy scores due to simplicity of answers, likely overfitting on questions and not a recommended solution

## METHOD 2: SQuAD-BASED QUESTION ANSWERING

- Custom corpus built on zoning ordinance documents and processed using custom corpus builder styled on nltk's corpus class
- Pretrained "deepset/roberta-base-squad" selected for initial model
- Corpus documents were then manually annotated with deepset's Haystack annotator and converted to SQuAD json format
- SQuAD data used to fine tune custom transformer reader and tokenize corpus document store used during model answer search
- Results were promising for future efforts and returned generally correct answers, but metrics could not be calculated due to time and labor constraints on annotation (train set could not be split due to small size)



## PROBLEMS

- Intel oneAPI Math Kernel Library (MKL) is used extensively for scientific computing—does not play well with non-Intel processors. Better documentation and tools like conda need to evolve to make using other platforms seamless.
- Zoning KGQAS may not be adequate when units of measurement are in the question. The KGQAS uses classes which may reject the same answers (asking for 60 inches versus 5 feet).
- SQuAD method is not good at answering yes/no questions. However, it still provides context from the document.

## RESULTS

Model	Model Comparisons			
	Accuracy	F1	Precision	Recall
0 Paper KGQAS	0.72	0.82	0.86	0.8
1 BERT Seq Classifier	0.897	0.513	--	--
2 ROBERTA Seq Classifier	0.934	0.648	--	--
3 Zoning KGQAS	--	--	--	--
4 ROBERTA SQuAD QA	0.0	0.0	--	--
5 ROBERTA SQuAD QA finetuned	--	--	--	--

## FUTURE DIRECTIONS

- Law is very reliant upon citation of where the information came from. Citation where in the document that information came from would be vital for a Zoning QAS. It is absent from this project.
- More properties represented by the Knowledge Graph including more relationship, metadata, and complicated relationships.
- Develop Knowledge Graphs ontologies for other related development codes such as Building Codes and Flood Ordinances.
- Creating a Knowledge Graph from a Zoning Ordinance with a permissive license allowing research would allow others to experiment with Question Answering in the Land Development Law domain. A simple contemporary model Zoning Ordinance is needed.
- Explore HayStack KG augmented Transformer QAS.
- Improvements upon the Question Answering Systems.

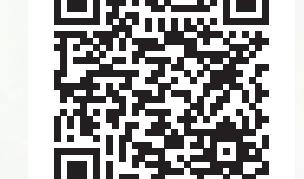
## Team Member Contributions:

Micah Cochran  
- Created Knowledge Graph, created Zoning Ordinance Corpus, Code for Training Questions, Zoning KGQAS system, poster, proposal

Seth Lewis  
- Corpus builder, Sequential Classifier QA tests, SQuAD based QA system, and proposal.

GitHub Link: <https://github.com/micahcochran/cs662-qa-land-dev-law-sys>

Repository Link



Notes: Monospace font is Source Code Pro. San serif font is Open Sans. Arial Orthographic is of Downtown Birmingham circa 2014, from USGS Earth Explorer. High Resolution Orthoimagery project. QR Code made with qrcode.