Risk Factor Categorization from SEC Filings

- Richard Albright
- CSE-6748
- Summer 2021
- ralbright7@gatech.edu
- GTID: 903548616

Risk Factor Categorization from SEC Filings

- Employer Website https://www.infilings.com

- Companies disclose risks to their business on a quarterly basis

- These risks are not required to be categorized.

- How do we compare one company’s risks to another?

Risk Factor Disclosure History

- In 2005, the Securities and Exchange Commission required significant risk factors to be disclosed in a Company’s financial filings.
- Companies must make a full risk disclosure annually as part of their 10-K filings (Forms 10-K, 10-KT).
- Companies must report new and changed risk disclosures quarterly in their 10-Q filings (Forms 10-Q, 10-QT).
- Companies making an Initial Public Offering must disclose risks in their registration statement (Forms S-1, S-1/A, 424B4).

Size and Scope of the Data Set

- Analyze 10-K, 10-KT, 10-Q, 10-QT, S-1, S-1/A, 424B4 filings
- Filings span from 2012 until 2020
- 254,837 Filings
- 3,764,242 Risk Disclosures
- 12.3 Gigabytes of Data (and that’s only Risk Disclosures!)

Processing the Data

- We don’t have categories yet.
- Analysts would need to manually review and tag every filing
- There’s just too much data to reasonably construct history with the resources available
- We can automate using Topic Modeling.
- How do we get the data?
- We have a REST API!
- But does it work?
- Where can it be improved?

Topic Modeling

- Topic Modeling is a form of Natural Language Processing
- What Model?
- Latent Dirichlet allocation (LDA)
- Word2Vec
- Hybrid?

LDA

- uses TF-IDF (term frequency – inverse document frequency)
- Reduces arbitrary lengths of word lists to fixed numeric arrays
- Only recognizes words it has seen before

Word2Vec

- Neural Network Algorithm to learn word associations
- Can detect synonyms of words
- Can suggest words from a fragment of text
- Each word is mapped to a number in a vector
- 2 Types: continuous bag of words or skip-gram

Hybrid Model

- We want a fixed set of risk categories defined by subject matter experts that can be defined using LDA.
- We also want the model to be flexible enough to guess at unknown words.
- Enter the Fasttext Algorithm (modified word2vec algorithm)
- Use LDA to train risk factor categories
- Use the Fasttext Algorithm to create word vectors on the risk factor text and the risk factor categories
- Many to one relationships allowed, a risk factor text can be assigned to multiple categories
- Use cosine similarity to assign each risk factor text to its closest category
- Not all risk factors need a category, set a cutoff on the probability prediction of unclear categories.

The Approach

- Optimally we’d like somewhere between 50 and 100 risk categories that can be reduced to a broader set of topics.
- Optimally run the LDA algorithm over a range of K topics and use the elbow method on the model’s Log Perplexity.
- Hand the results of this model off to subject matter experts to tag a random sample of risk disclosures using the preset risk factor categories.
- Run the LDA algorithm over each tagged category to generate a vector for that category.
- Run the Fasttext algorithm on the category word vectors and the risk factor text
- Use cosine similarity to determine which risk factor category the risk factor text is then assigned.

What can we do with the results?

- We can create a risk profile based on the broader risk categories
- Use page rank algorithm on those risk profiles to recommend companies with similar risk factors
- We can weight the graph using each company’s Sharpe Ratio (measure of a company’s risk based on stock volatility)
- Return the top X results to a client researching a particular company.
- Does the risk profile match insider activity (SEC Form 4 Disclosures)?
- Given enough time can we create an insider activity profile for each company as well and combine the 2 models?

Whats been done?

- The external api has been tested and modified to make it easier for a client to use.
- The api returns a filing in json format to make it easier for clients to process.
- The text returned is in raw format and must be preprocessed
- All filings have been preprocessed using the Spark-NLP library
- Text has been split into sentences.
- Dates have been normalized into a consistent format.
- Sentences were then tokenized into words, normalized, lemmatized, then had their stop words removed.
- Tokenized words were then used to form 2 and 3 word N-grams.
- Certain N-grams were removed based on part of speech tagging.

Whats been done (continued)?

- Overruled by management on finding the optimal number of topics.
- An 80 topic LDA model was constructed as a first pass and handed off to the analysts (subject matter experts) to help them with category definitions.
- 1/3 of risk disclosure topics were financial in nature.
- The initial results are in the hands of analysts to create the final risk factor categories and to tag a sample set with the categories they are developing.
- Fasttext algorithm using cosine similarity to match to the LDA vectors was tested on a small sample from the initial cluster to verify the approach.
- Working on company insider activity profiling while waiting for subject matter experts.
- Insider Activity “Events” are broken into 5 categories – buybacks, management changes, positive events, negative events, research commentary
- A profile is created over a lookback period based on counts of events in each category.
- Tested Jaccard Similarity, cosine similarity, and Euclidean Distance as measures of similarity.
- Euclidean Distance was determined to be the best measure of similarity for company insider activity profiles.

Timing Considerations

- The time waiting for analysts to tag data is a HUGE variable.
- The model may be built out using the original proposal of finding the optimal number of risk factors.
- Use the unsupervised clusters as the model and build the rest of the project so the code is complete
- The model can always be rerun when the final risk factor categories are determined.
- The proof of concept would be complete, only the results would change.
- The insider activity model is not the main focus of this project and may not be completed given the time considerations.
- A combined model of risk factors and insider activity may not be completed given the time considerations.

What needs to be done?

- Training the risk categories on a predetermined set of risk categories as determined by the analysts.
- Create risk profiles based on the counts of risk factors across the yet to be determined broad risk factor categories.
- Create a page rank algorithm using the risk factor categories to return companies that are most similar to a given company.
- Create company insider activity profiles
- Create a page rank algorithm for companies with similar insider activity.

Recommendations Made

- The Rest API had some bugs that have since been addressed
- Bulk mode for risk factors was non-existent, even though it is a wrapper function around another method that has a bulk mode, each filing was downloaded individually.
- The Rest API has a bag of words mode that is not sufficient for the many different types of NLP that can be run on the data set. Recommend that we return results of multiple stages of the text preprocessing pipeline to clients.
- The bag of words mode in the API is done strictly via Perl Regex. Possibly store the results of this project’s preprocessing and return those results instead.

API Definitions

FilingList

Get a list of filings

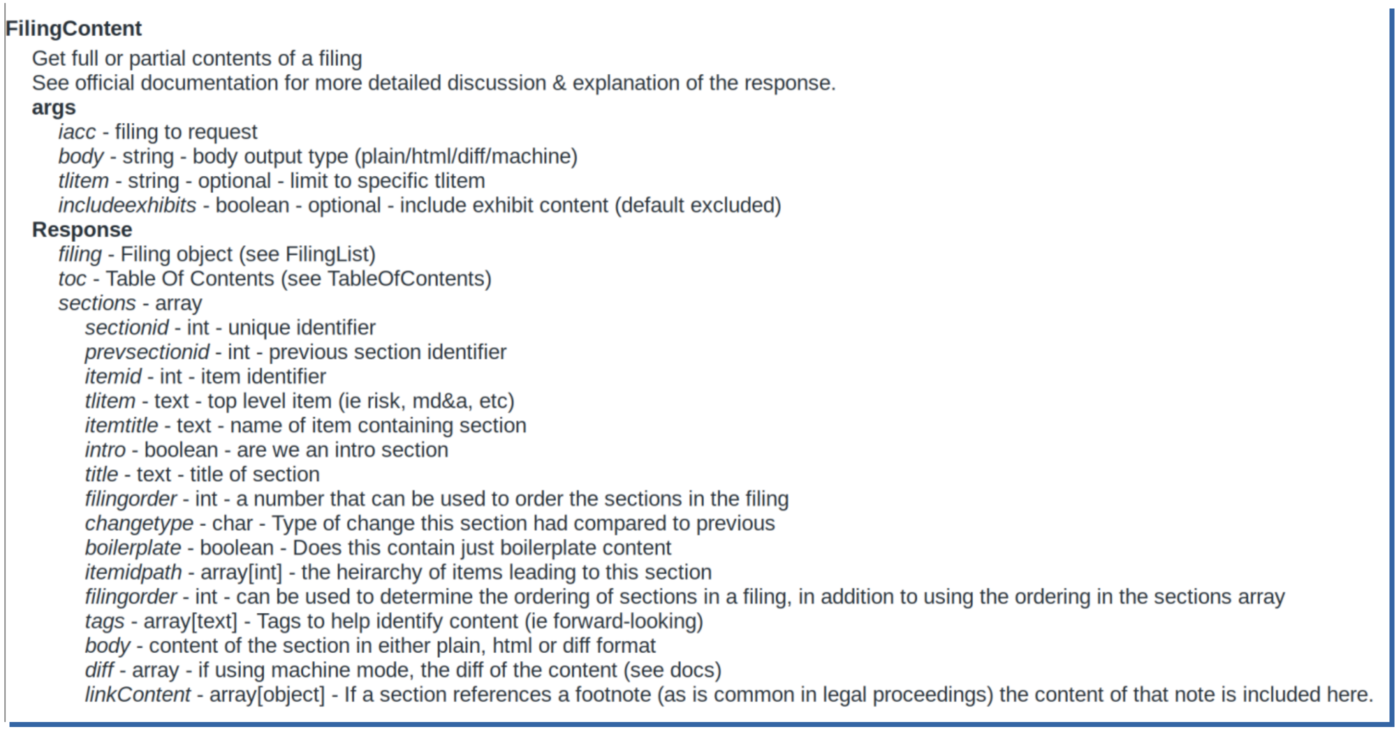
Args

- *<companyid> - optional*
- *newerSince - optional - (date YYYYMMDD HH:MM:SS)*
- *mcapLow - int - optional - lower mcap range*
- *mcapHigh - int - optional - upper mcap range*
- *formTypes - string[] - optional - array of formtypes to include*
- *limit - int - optional - default 100*
- *offset - int - optional - offset into search results to start from*

Response

- *filings* - array
  - *ticker*
  - *CUSIP*
  - *companyname*
  - *iacc*
  - *cik*
  - *accessionnum* - string - EDGAR accession number
  - *formtype* - string
  - *datefiled* - date filing was made
  - *filedtimestamp* - date w/time when filed
  - *spotapproved* - boolean - if we reviewed & approved the filing
  - *spotapproveddate* - date w/time - when filing was approved.
  - *seclink* - link to view filing on EDGAR (sec.gov)

API Definitions (continued)



FilingContent

Get full or partial contents of a filing
See official documentation for more detailed discussion & explanation of the response.

args

- *iacc* - filing to request
- *body* - string - body output type (plain/html/diff/machine)
- *tlitem* - string - optional - limit to specific tlitem
- *includeexhibits* - boolean - optional - include exhibit content (default excluded)

Response

- *filing* - Filing object (see FilingList)
- *toc* - Table Of Contents (see TableOfContents)
- *sections* - array
  - *sectionid* - int - unique identifier
  - *prevsectionid* - int - previous section identifier
  - *itemid* - int - item identifier
  - *tlitem* - text - top level item (ie risk, md&a, etc)
  - *itemtitle* - text - name of item containing section
  - *intro* - boolean - are we an intro section
  - *title* - text - title of section
  - *filingorder* - int - a number that can be used to order the sections in the filing
  - *changetype* - char - Type of change this section had compared to previous
  - *boilerplate* - boolean - Does this contain just boilerplate content
  - *itemidpath* - array[int] - the heirarchy of items leading to this section
  - *filingorder* - int - can be used to determine the ordering of sections in a filing, in addition to using the ordering in the sections array
  - *tags* - array[text] - Tags to help identify content (ie forward-looking)
  - *body* - content of the section in either plain, html or diff format
  - *diff* - array - if using machine mode, the diff of the content (see docs)
  - *linkContent* - array[object] - If a section references a footnote (as is common in legal proceedings) the content of that note is included here.

API Definitions (continued)

MultiFilingContent

Get full or portial contents of multiple filings - a combination of FilingList and FilingContent

args

See arguments for FilingList and FilingContent.

NOTE: Be aware of FilingList's default limit of 100 filings

Response

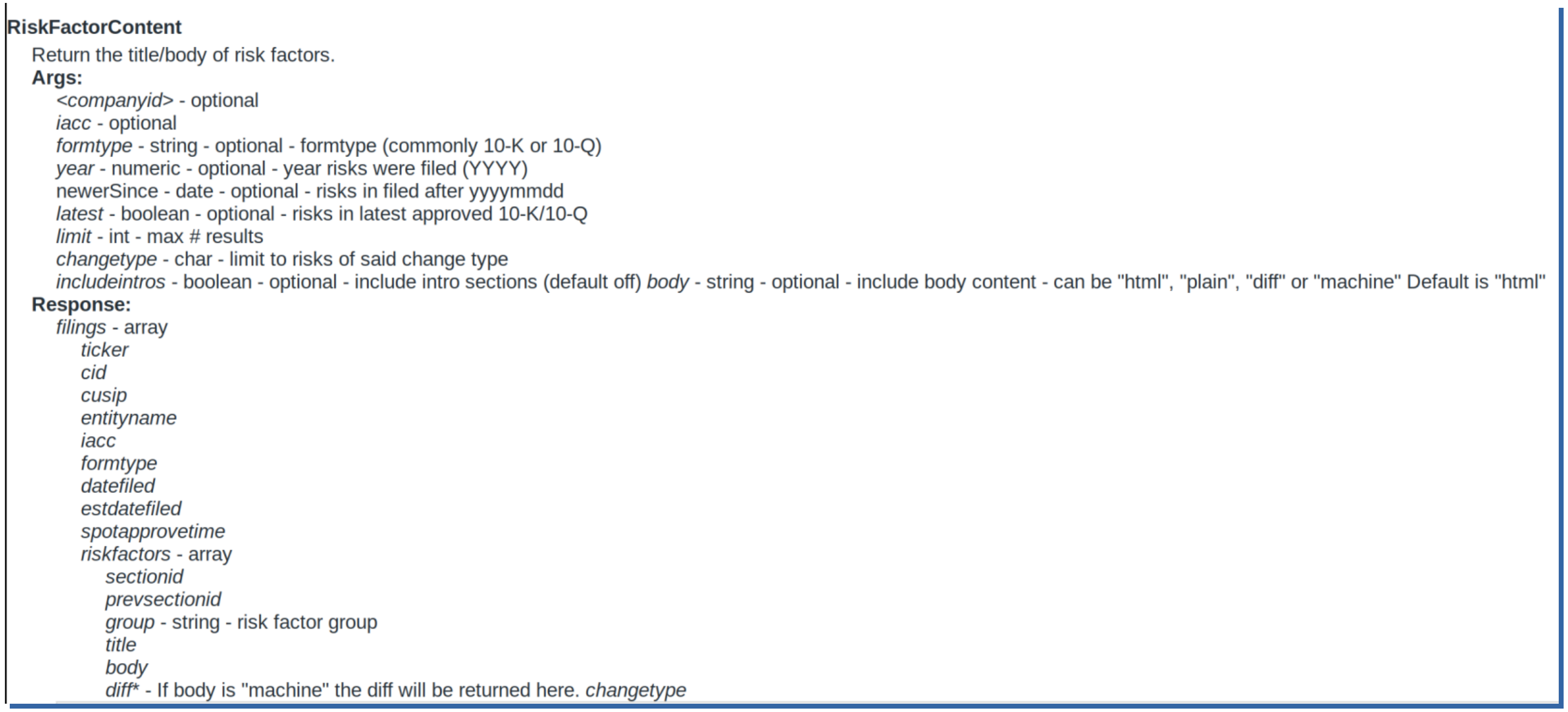
filings - array

[Same as FilingList.filings Response, with addition of sections[] array]

sections - array

[Same as FilingContent.sections]

API Definitions (continued)



RiskFactorContent

Return the title/body of risk factors.

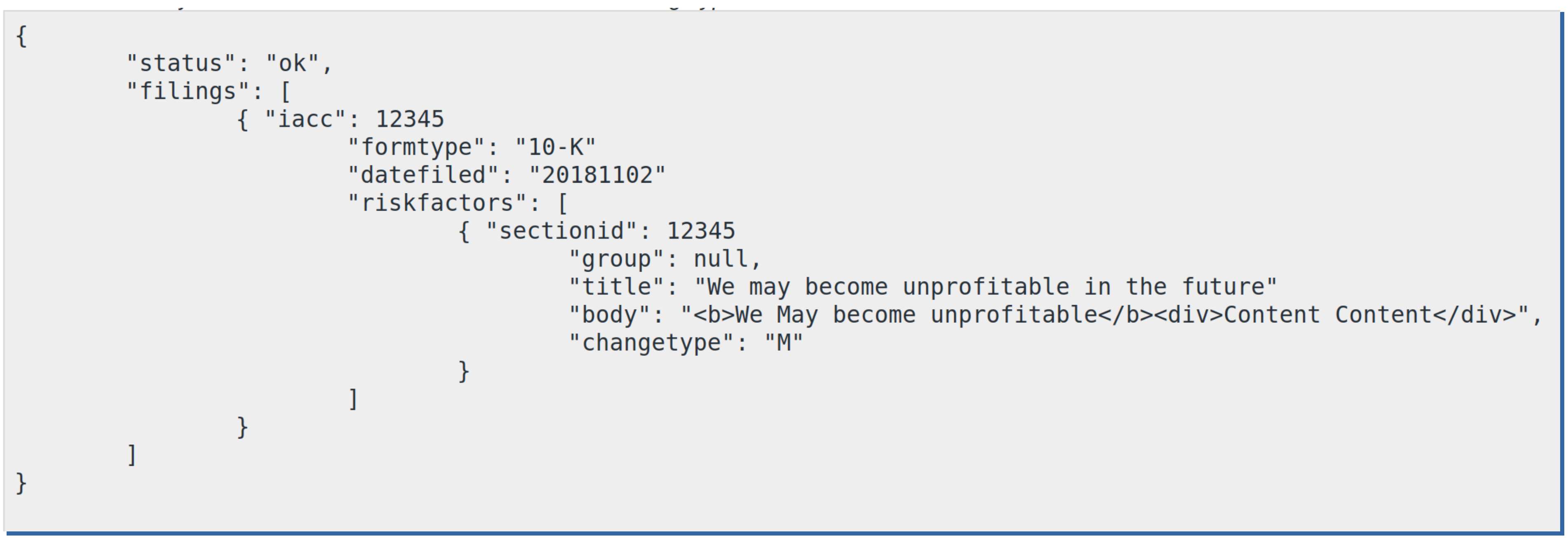
Args:

- *<companyid>* - optional
- *iacc* - optional
- *formtype* - string - optional - formtype (commonly 10-K or 10-Q)
- *year* - numeric - optional - year risks were filed (YYYY)
- newerSince - date - optional - risks in filed after yyyymmdd
- *latest* - boolean - optional - risks in latest approved 10-K/10-Q
- *limit* - int - max # results
- *changetype* - char - limit to risks of said change type
- *includeintros* - boolean - optional - include intro sections (default off) *body* - string - optional - include body content - can be "html", "plain", "diff" or "machine" Default is "html"

Response:

- *filings* - array
  - *ticker*
  - *cid*
  - *cusip*
  - *entityname*
  - *iacc*
  - *formtype*
  - *datefiled*
  - *estdatefiled*
  - *spotapprovetime*
  - *riskfactors* - array
    - *sectionid*
    - *prevsectionid*
    - *group* - string - risk factor group
    - *title*
    - *body*
    - *diff** - If body is "machine" the diff will be returned here. *changetype*

Risk Factor Json Results Sample



```
{
        "status": "ok",
        "filings": [
                { "iacc": 12345
                        "formtype": "10-K"
                        "datefiled": "20181102"
                        "riskfactors": [
                                { "sectionid": 12345
                                        "group": null,
                                        "title": "We may become unprofitable in the future"
                                        "body": "<b>We May become unprofitable</b><div>Content Content</div>",
                                        "changetype": "M"
                                }
                        ]
                }
        ]
}
```