Mining for spillovers in patents
Plan of attack

Jacob Calder[*], Iwo Dubaniowski [†], C. Vic Hu [‡], Subhashini Venugopalan[§]

EE380L Data Mining
University of Texas at Austin
https://github.com/cvhu/ee380l-ghosh-project

March 25, 2013

1 Approach

Here is a detailed systematic approach to make inferences from the topic model.

1. Consolidate the data consisting of - abstract, claims, date of filing/grant, geographical location, technlogy area, classification numbers, patent-holder information and other (less important) details.

2. Run the topic model (a few times) on the abstract and claims, to generate a list of consistent topics word clusters. Call this $TopicWordsCluster_{TM}$.

3. To make any inference, we first need to associate the $TopicWordsCluster_{TM}$ with the actual topic (e.g. with respect to the example in Topic Modelling web page and tutorial (http://www.programminghistorian.org/lessons/topic-modelling-and-mallet), that would be cricket, movies, wildlife etc.) And we don't know these topics, so we need to identify them based on the technology areas and classification numbers.

   (a) Identify the technology areas correlated with each $TopicWordsCluster_{TM}$.
   This can be done by first identifying the most correlated topics for each document. Then, associating the document's technology area with that $TopicWordsCluster_{TM}$. Doing this for all documents, we can get a list of technology areas corresponding to each topic, and counting the frequecies will give us a single technology area for each topic.
   This has been tested on the abstracts. The scripts and results can be found in https://github.com/cvhu/ee380l-ghosh-project/tree/master/data/pyAnalysisOutput

[*] jcalder@utexas.edu
[†] mduban@gmail.com
[‡] vic@cvhu.org
[§] vsubhashini@utexas.edu

(b) Similarly associate classification classes with every $TopicWordsCluster_{TM}$.
To accomplish this, first get the classification names and details corresponding to each of the classification numbers for each patent. Now, the algorithm is very similar to the previous step (as in technology areas). For each patent, identify the most correlated topic and associate it's classification details to that topic. Then refine by topic to identify the list of classifications for each $TopicWordsCluster_{TM}$, and pick the top 3-4 classifications.

4. From the previous step, we should have now generated a set of $Topics_{TM}$ or topic names for each $TopicWordsCluster_{TM}$ based on technology area and patent classification details. The next step is to do the actual inference with respect to geographies and time.

   (a) For geographies (and time), run the topic model again to generate a bunch of $TopicWordsCluster_{Geo}$ (correspondingly $TopicWordsCluster_{Years}$). Map $TopicWordsCluster_{Geo}$ ($TopicWordsCluster_{Years}$) with $TopicWordsCluster_{TM}$ based on the words directly (or re-doing step 3). The result of this will be $Topics_{Geo}$ (and $Topics_{Years}$).

   (b) Now we will have for each geography the set of most frequent $Topics$, and we should be able to test our hypothesis:

   - Are certain geographies (and years) associated with specific topics?
   - Is there a progression in the set of topics across years?
   - Map $Topics_{Geo}$ (and $Topics_{Years}$) to see if $(Topic_i, Geo_j, Year_k)$ has some association with $(Topic_i, Geo_l, Year_{j+1})$ etc.