Sistema recomendador basado en tópicos latentes

María Emilia Charnelli[1,3], Laura Lanzarini[2], Javier Díaz[1]

[1]LINTI - Laboratorio de Investigación en Nuevas Tecnologías Informáticas
[2]III LIDI - Instituto de Investigación en Informática LIDI
Facultad de Informática, Universidad Nacional de La Plata
[3] CONICET - Consejo Nacional de Investigaciones Científicas y Técnicas

mcharnelli@linti.unlp.edu.ar, laural@lidi.info.unlp.edu.ar,
jdiaz@unlp.edu.ar

Resumen. El filtrado colaborativo es una de las técnicas más utilizadas en los sistemas de recomendación. El objetivo del presente artículo es proponer un nuevo método que utiliza tópicos latentes para modelar los ítems a recomendar. De esta forma se incorpora la capacidad para establecer una semejanza entre estos elementos mejorando el rendimiento de la recomendación realizada. La performance del método propuesto ha sido medida en dos contextos muy diferentes arrojando resultados satisfactorios. Finalmente se incluyen las conclusiones y algunas líneas de trabajo futuras.

Palabras clave: Sistemas de Recomendación, Filtrado Colaborativo, Modelado de Tópicos Latentes.

1. Introducción

Los sistemas de recomendación analizan patrones de interés del usuario como artículos o productos, para proporcionar recomendaciones personalizadas que satisfagan sus preferencias [1]. Las sugerencias intervienen en varios procesos de toma de decisiones, tales como qué artículos comprar, qué películas mirar, o qué libros leer. El término ítem es utilizado para indicar lo que el sistema recomienda a los usuarios [2]. Para poder llevar esto adelante, es necesario modelar a los ítems que se quieren recomendar. La generación de un modelo a partir de la información textual y no estructurada de un conjunto de ítems representa un gran desafío. El análisis de tópicos latentes ha emergido como uno de los métodos más eficientes para clasificar, agrupar y recuperar datos textuales. Descubrir los tópicos en textos cortos es crucial para un amplio rango de tareas que analizan tópicos, como caracterizar contenido, modelar perfiles de intereses de usuarios, detectar tópicos latentes o emergentes. El modelo de tópicos bitérmino BTM [3] permite extraer de forma eficaz los tópicos que caracterizan a un conjunto de textos cortos. Con BTM se pueden obtener los temas subyacentes en un conjunto de documentos y una distribución global de cada tópico sobre cada uno de ellos, a través del análisis de la generación de bitérminos.

El enfoque más común para un sistema recomendador es la técnica de filtrado colaborativo basada en modelos de vecindad. Su forma original está basada en las similitudes entre usuarios [4]. Dichos métodos de usuario-usuario estiman puntajes desconocidos basados en puntajes registrados de usuarios con ideas afines. Posteriormente, se hizo popular el enfoque análogo pero ahora teniendo en cuenta las similitudes entre ítems [5] [6]. En estos métodos, se calcula un puntaje utilizando valoraciones realizadas por el mismo usuario en ítems similares. Una mejor escalabilidad y una precisión mejorada hacen que el enfoque por ítems sea más favorable en muchos casos [7] [8]. Además, los métodos de ítem-ítem son más susceptibles de explicar el razonamiento detrás de las predicciones. Esto se debe a que los usuarios están familiarizados con los elementos previamente preferidos por ellos, pero no conocen a los usuarios supuestamente parecidos. La mayoría de los enfoques de ítem-ítem utilizan una medida de similitud entre los ratings que tienen los mismos.

En este trabajo se propone un método basado en el enfoque ítem-ítem que utiliza un modelo de tópicos latentes para modelar a los ítems que se requieren recomendar y establece una semejanza entre estos elementos que mejoran el rendimiento de la recomendación. La evaluación del método propuesto se realiza mediante un conjunto de materiales educativos del repositorio digital Merlot[9] y un dataset de películas de MovieLens [10]. El presente artículo está organizado de la siguiente forma: la segunda sección describe el preprocesamiento efectuado sobre los datasets, la tercera sección muestra la extracción y modelado de tópicos latentes, la cuarta sección describe el método propuesto, en la quinta sección se muestran los resultados experimentales. Finalmente, en la sexta sección se presentan las conclusiones y las líneas de trabajo futuras.

2. Preparación de los datos

En este trabajo se utilizaron dos bases de datos, una de materiales educativos y otra de películas. La primera, brinda información de usuarios y materiales educativos del área de la Ciencias de la Computación publicados en el repositorio digital Merlot [9]. Los datos involucran más de 984 materiales y más de 260 usuarios que subieron, evaluaron o comentaron a cada una de las publicaciones. Así también, se dispone de información pública de las publicaciones y de los usuarios. De cada una de las publicaciones se obtuvo: título, tipo de material, fecha de creación, fecha de actualización, usuario que la realizó, valoración de revisores de 1 a 5, valoración de usuarios de 1 a 5, comentarios, y la descripción textual no estructurada. Mientras que el segundo dataset es sobre ratings de películas de MovieLens. Este dataset contiene 100.000 puntajes de 1 a 5 de 943 usuarios en 1682 películas, donde cada usuario evaluó al menos 20 películas; de las películas se conoce el título y la fecha; y además, se recolectaron los argumentos de cada una de ellas.

Cuando se trata de operar con información textual es preciso recurrir a técnicas de Minería de Texto (Text Mining) a fin de poder representar a cada descripción en un vector de términos. Esto fue llevado a cabo a través de un proceso

compuesto por varias etapas. En una primera etapa, se unificaron los contenidos en un único idioma. Luego se aplicó un filtro de stopwords, que se encarga de filtrar las palabras que coincidan con cualquier stopword indicado. Se filtraron stopwords del idioma inglés; palabras propias del contexto. También se eliminaron direcciones de páginas web, caracteres no textuales. Luego, cada palabra en el texto fue reducida a su raíz aplicando el algoritmo de stemming Snowball [11]. La importancia de este proceso radica en que elimina las variaciones sintácticas relacionadas con el género, número y tiempo verbal. Una vez que se obtienen las raíces de cada una de las palabras se calculó la frecuencia de aparición de cada una de ellas en las publicaciones y se escogieron las palabras que aparecen más de una vez.

3. Extracción de tópicos latentes

Para la extracción de los tópicos en las descripciones de los ítems se utilizó BTM (Biterm Topic Model) que es una técnica de aprendizaje no supervisado que descubre los tópicos que caracterizan a un conjunto de documentos breves.

Sea un conjunto de N_D documentos denominado corpus y sea W el conjunto de todas las palabras del corpus, un tópico se define como una distribución de probabilidad sobre W. Por lo tanto, un tópico puede ser caracterizado por sus T palabras mas probables. Dado un número K de tópicos, el objetivo de BTM consiste en obtener las K distribuciones sobre cada una de las palabras. Un "bitérmino" denota a un par de palabras sin orden que co-ocurren en un documento corto. En este caso, dos palabras diferentes en un documento construyen un bitérmino. Dado un corpus con N_D documentos y un vocabulario W de palabras únicas, se supone que contiene N_B bitérminos $\boldsymbol{B} = \{b_i\}_{i=1}^{N_B}$ con $b_i = (w_{i,1} \in W, w_{i,2} \in W)$, y K tópicos expresados sobre W. Sea $z \in [1, K]$ una variable para indicar un tópico. La probabilidad $P(z)$ de que un documento en el corpus sea de un tópico z, se define como una distribución multinomial K-dimensional $\boldsymbol{\theta} = \{\theta_k\}_{k=1}^{K}$ con $\theta_k = P(z = k)$ y $\sum_{k=1}^{K} \theta_k = 1$. La distribución de palabras por tópico $P(w|z)$ puede ser representada como una matriz $\Phi \in R^{K \times W}$ donde la k-ésima fila ϕ_k es una distribución multinomial W-dimensional con entrada $\phi_{k,w} = P(w|z = k)$ y $\sum_{w=1}^{W} \boldsymbol{\phi}_{k,w} = 1$. Dados los parámetros α y β, la suposición principal del modelo consiste en asumir que cada uno de los documentos del corpus fueron generados de la siguiente manera:

1. Se elige una distribución de tópicos $\boldsymbol{\theta} \sim Dirichlet(\alpha)$ para todo el corpus
2. Por cada tópico $k \in [1, K]$
   - Se extrae una distribución de palabras para el tópico $\phi_k \sim Dirichlet(\beta)$
3. Por cada bitérmino $b_i \in \boldsymbol{B}$
   - Se extrae una asignación de tópicos $z_i \sim Multimonial(\boldsymbol{\theta})$
   - Se extraen dos palabras $w_{i,1}, w_{i,2} \sim Multimonial(\phi_{z_i})$

Teniendo en cuenta el mecanismo de generación supuesto por BTM, se puede obtener la verosimilitud para todo el corpus dado los parámetros α y β a partir de la probabilidad de cada uno de los bitérminos:

$$P(\boldsymbol{B}|\alpha,\beta) = \prod_{i=1}^{N_B} \int\int \sum_{k=1}^{K} P(w_{i,1}, w_{i,2}, z_i = k|\boldsymbol{\theta}, \boldsymbol{\Phi}) d\boldsymbol{\theta} d\boldsymbol{\Phi} \quad (1)$$

$$= \prod_{i=1}^{N_B} \int\int \sum_{k=1}^{K} \theta_k \phi_{k,w_{i,1}} \phi_{k,w_{i,2}} d\boldsymbol{\theta} d\boldsymbol{\Phi} \quad (2)$$

Obtener exactamente los parámetros $\boldsymbol{\theta}$ y $\boldsymbol{\Phi}$ que maximizan la verosimilitud de la ecuación 2 es un problema intratable. Siguiendo lo propuesto en [12], los parámetros $\boldsymbol{\theta}$ y $\boldsymbol{\Phi}$ pueden ser aproximados utilizando muestreo de Gibbs [13].

Para inferir los temas de un documento, es decir, evaluar $P(z|d)$ para el documento d, se deriva la proporción de tópicos de un documento a través de los tópicos de los bitérminos. Si d contiene N_d bitérminos, $\{b_i^{(d)}\}_{i=1}^{N_d}$,

$$P(z|d) = \sum_{i=1}^{N_d} P(z|b_i^{(d)}) P(b_i^{(d)}|d) \quad (3)$$

3.1. Criterio de evaluación

Para evaluar la calidad de los tópicos obtenidos se utiliza la métrica de coherencia propuesta por Mimno et al. [14]. Dado un tópico z y sus T palabras más probables $V^{(z)} = (v_1^{(z)}, ..., v_T^{(z)})$ donde $v_i^{(z)} \in W$ para $i = 1...T$, el puntaje de coherencia es definido como:

$$C(z; V^{(z)}) = \sum_{t=2}^{T} \sum_{l=1}^{t} \log \frac{D(v_t{}^{(z)}, v_l{}^{(z)}) + 1}{D(v_l{}^{(z)})}$$

donde $D(v)$ es la frecuencia de la palabra v en todos los documentos, $D(v, v')$ es el número de documentos en donde las palabras v y v' co-ocurren. La métrica de coherencia está basada en la idea de que las palabras que pertenecen a un mismo concepto tenderán a co-ocurrir dentro de los mismos documentos. Esto es empíricamente demostrable porque el puntaje de coherencia está altamente correlacionado con el criterio humano. Para evaluar la calidad en general de un conjunto de tópicos, se calcula el promedio de la métrica de coherencia para cada uno de los tópicos obtenidos $\frac{1}{k}\sum_k C(z_k; V^{(z_k)})$. Estos resultados nos permiten determinar la cantidad de tópicos que mejor representan a todo el corpus.

4. Método propuesto

Sea K el número de tópicos que representan a un conjunto de ítems, se modelan cada uno de los mismos según la distribución de probabilidad mostrada en la ecuación 3.

Sea una lista de m usuarios $U = u_1, u_2, .., u_m$ y una lista de n ítems $I = i_1, i_2, ..., i_n$. Cada usuario tiene una lista de ítems I_u, con un puntaje asociado a cada ítem r_{ui}. Cada ítem tiene asignado un puntaje de 1 a 5.

Con el objetivo de evaluar la semejanza entre dos ítems a partir de las distribuciones de probabilidad obtenidas con BTM, el método propuesto utiliza la divergencia de Kullback-Leibler [15]. Dadas dos distribuciones de probabilidad P y Q la función de divergencia se define como:

$$D_{KL}(P, Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}$$

A partir de esta divergencia, es posible definir la semejanza entre dos ítems p y q de la siguiente manera:

$$sim(p, q) = \exp(-D_{KL}(p, q))$$

Para estimar el rating de un nuevo ítem m dado un usuario u, se propone el siguiente método para predecir $\hat{r}_{um}$:

1. Se obtienen las distribuciones de probabilidad para cada uno de los ítems que el usuario evaluó I_u, según se mostró en la sección 3.
2. Se obtiene la distribución de probabilidad del material m.
3. Se calcula la semejanza sim de m con cada I_{u_j}.
4. Se ordenan las semejanzas, y se eligen las primeras t, donde t es un parámetro que define el tamaño de vecindad a considerar.
5. A partir de los t más semejantes, se calcula la predicción:

$$\hat{r}_{um} = \mu_m + \frac{\sum_{j=1}^{t} sim(m, j)(r_{uj} - \mu_j)}{\sum_{j=1}^{t} sim(m, j)}$$

donde r_{uj} es el puntaje del ítem j dado por el usuario u, y μ_j y μ_m son los puntajes promedios de j y m respectivamente.

5. Resultados experimentales

Se evaluó el modelo obtenido por BTM en el conjunto de los materiales educativos y de películas. Por cada número de tópicos entre 2 y 30 se promedió la coherencia obtenida, muestreando aleatoriamente el conjunto de prueba y entrenamiento en 1000 iteraciones. La figura 1 muestra el promedio de la coherencia del modelo con respecto a la cantidad de tópicos extraída en el dataset de materiales. Interesa el número de tópicos en el que se produce un quiebre en el crecimiento de la función de la coherencia promedio. En este caso, el valor óptimo se encuentra entre 5 y 7 tópicos latentes.

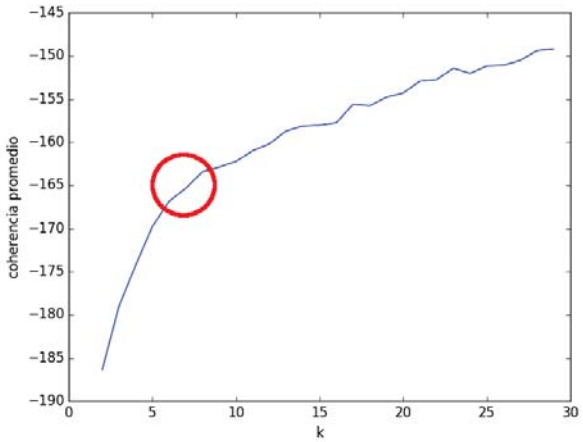

Figura 1. Dataset materiales. Coherencia promedio para distintos K

En la tabla 1 se muestran los tópicos obtenidos con K=7 para el dataset de materiales. Para cada uno de los tópicos se muestran las seis palabras más importantes, es decir, aquellas que tienen más probabilidad de pertenecer a dicho tópico.

Tópico	Palabras más importantes del tópico				
1	programming	software	data	algorithms	design
2	information	technology	computing	internet	systems
3	programming	java	language	tutorial	software
4	resources	design	systems	development	security
5	design	information	programming	interaction	human
6	binary	fractions	codes	numbers	tutorial
7	numbers	stars	interactivate	graph	simulation

Tabla 1. Dataset materiales. Modelo de tópicos obtenidos con BTM

La metodología de evaluación para el método propuesto aplica validación cruzada 10-fold. Este proceso de evaluación se repitió 50 veces para obtener una muestra significativa sobre la cual se promedian los resultados. Este proceso se aplicó para el método propuesto, identificado como KNN Topic Model, y sobre los métodos de filtrado colaborativo KNN, KNN Mean [7] basados en el enfoque ítem-ítem y usuario-usuario, SlopeOne [16] y sobre el método basado en modelos de factores latentes (SVD) [17].

El método propuesto y los métodos KNN reciben como parámetro la cantidad de vecinos a considerar. El tamaño del vecindario tiene un impacto significativo en la calidad de la predicción [4]. En la figura 2 se muestra el error RMSE (Root Mean Squared Error) para diferentes números de vecinos en los distintos algoritmos. El error decrece a medida que la cantidad de vecinos crece. El error para el método propuesto KNN Topic Model siempre está por debajo para distintos valores de vecindad. Además, se observa que luego de 40 vecinos el RMSE decrece lentamente para cada uno de los algoritmos.

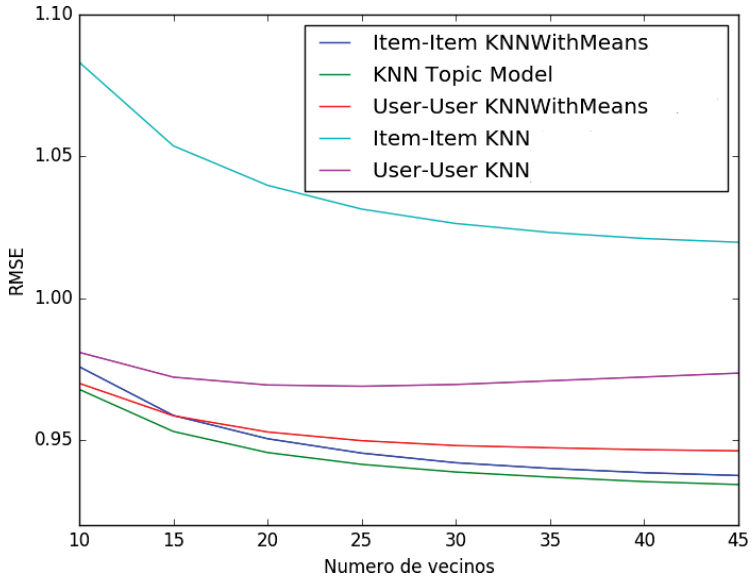

Figura 2. Dataset películas. Influencia del tamaño de vecindad

Los resultados de las 50 ejecuciones de la validación cruzada para cada algoritmo utilizando el dataset de materiales se muestran en la tabla 2 y los resultados al utilizar el dataset de películas se muestran en la tabla 3. Se estableció el número de vecinos $t = 40$ para todos los modelos basados en vecindad. Los ítems del dataset de materiales fueron representados como una distribución multinomial 7-dimensional y los ítems del dataset de películas como una distribución multinomial 10-dimensional. Para evaluar las predicciones del método propuesto frente a los resultados de los otros algoritmos, se calcularon las métricas de precisión RMSE (Root Mean Squared Error), MAE (Mean Absolute Error) y FCP (Fraction of Concordant Pairs), que mide la proporción de pares de ítems bien clasificados [18]. A diferencia de RMSE y MAE, el valor de FCP es mejor cuanto más alto es, porque mide una proporción.

Se observa que el método propuesto es competitivo frente a dos conjuntos de datasets diferentes. Para el dataset de materiales educativos el método KNN Topic Model obtiene un error RMSE más bajo y una proporción FCP más alta. Sin embargo, la métrica MAE es menor para SVD. Se destaca que con la poca información de los materiales que se disponen, a través de la utilización del modelado de tópicos es posible mejorar el FCP. En el dataset de películas se dispone de mayor información de los intereses de los usuarios, por lo que el método propuesto, si bien tiene un resultado competitivo, no supera el valor de FCP con respecto al enfoque KNN Mean usuario-usuario.

	KNN Model Topic	KNN item-item	KNN Mean item-item	KNN user-user	KNN Mean user-user	SlopeOne	SVD
Mean RMSE	**0,6047**	0,6848	0,8412	0,7757	0,6339	0,6575	0,6420
Mean MAE	0,4403	0,4566	0,5544	0,5126	0,4333	0,4336	**0,3443**
Mean FCP	**0,6517**	0,2075	0,4820	0,1400	0,3333	0,4133	0,4329

Tabla 2. Dataset materiales educativos. Resultados obtenidos

	KNN Model Topic	KNN item-item	KNN Mean item-item	KNN user-user	KNN Mean user-user	SlopeOne	SVD
Mean RMSE	**0,9340**	1,0203	0,9385	0,9732	0,9466	0,9426	0,9402
Mean MAE	**0,7359**	0,8044	0,7375	0,7680	0,7462	0,7409	0,7396
Mean FCP	0,6879	0,5990	0,6867	0,6948	**0,6946**	0,6865	0,6889

Tabla 3. Dataset películas. Resultados obtenidos

6. Conclusiones y líneas de trabajo futuras

En el presente artículo se logró modelar a un conjunto de ítems utilizando la detección de tópicos latentes a partir de las descripciones de los mismos. Esto permitió saber cuáles son los tópicos que describen a los ítems y cómo se relacionan entre sí. La metodología utilizada en el método propuesto y las métricas de validación aplicadas presentan resultados preliminares satisfactorios y competitivos frente a los métodos tradicionales. Como trabajo futuro se prevé la aplicación del método propuesto en otras bases de datos con información textual asociada. Resulta de interés, además, incorporar información acerca de las opiniones y gustos del usuario desde otros contextos. Los resultados de este trabajo se suman a lo previamente realizado en [19], donde se propuso un modelado

de usuarios a través de la información obtenida con BTM al identificar los tópicos de interés de los alumnos de la Facultad de Informática de la UNLP a través de sus publicaciones realizadas en grupos de Facebook. A su vez, este trabajo se relaciona con un proyecto más grande, cuyo objetivo es crear un sistema de recomendación de materiales digitales educativos.

Referencias

1. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE transactions on knowledge and data engineering **17** (2005) 734–749
2. Ricci, F., Rokach, L., Shapira, B.: Introduction to recommender systems handbook. In: Recommender systems handbook. Springer (2011) 1–35
3. Cheng, X., Yan, X., Lan, Y., Guo, J.: Btm: Topic modeling over short texts. IEEE Transactions on Knowledge and Data Engineering **26** (2014) 2928–2941
4. Herlocker, J.L., Konstan, J.A., Borchers, A., Riedl, J.: An algorithmic framework for performing collaborative filtering. In: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, ACM (1999) 230–237
5. Linden, G., Smith, B., York, J.: Amazon. com recommendations: Item-to-item collaborative filtering. IEEE Internet computing **7** (2003) 76–80
6. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th international conference on World Wide Web, ACM (2001) 285–295
7. Bell, R.M., Koren, Y.: Scalable collaborative filtering with jointly derived neighborhood interpolation weights. In: Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on, IEEE (2007) 43–52
8. Takács, G., Pilászy, I., Németh, B., Tikk, D.: Major components of the gravity recommendation system. ACM SIGKDD Explorations Newsletter **9** (2007) 80–83
9. University, C.S.: Merlot - multimedia educational resource for learning and online teaching. `https://merlot.org` (2017 (accessed June 30, 2017))
10. Researc, G.: Movielens datasets. `https://grouplens.org/datasets/movielens/` (2017 (accessed June 30, 2017))
11. Gupta, V., Lehal, G.S.: A survey of common stemming techniques and existing stemmers for indian languages. Journal of Emerging Technologies in Web Intelligence **5** (2013) 157–161
12. Griffiths, T.L., Steyvers, M.: Finding scientific topics. Proceedings of the National academy of Sciences **101** (2004) 5228–5235
13. Geman, S., Geman, D.: Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. IEEE Transactions on pattern analysis and machine intelligence (1984) 721–741
14. Mimno, D., Wallach, H.M., Talley, E., Leenders, M., McCallum, A.: Optimizing semantic coherence in topic models. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics (2011) 262–272
15. Kullback, S., Leibler, R.A.: On information and sufficiency. The annals of mathematical statistics **22** (1951) 79–86
16. Lemire, D., Maclachlan, A.: Slope one predictors for online rating-based collaborative filtering. In: Proceedings of the 2005 SIAM International Conference on Data Mining, SIAM (2005) 471–475

17. Mnih, A., Salakhutdinov, R.R.: Probabilistic matrix factorization. In: Advances in neural information processing systems. (2008) 1257–1264
18. Koren, Y., Sill, J.: Collaborative filtering on ordinal user feedback. In: IJCAI. (2013) 3022–3026
19. Charnelli, M.E., Lanzarini, L., Diaz, J.: Modeling students through analysis of social networks topics. XXII Congreso Argentino de Ciencias de la Computacion CACIC 2016 (2016) 363–371