DEPARTEMEN INFORMATIKA

FAKULTAS TEKNOLOGI INFORMASI DAN KOMUNIKASI

INSTITUT TEKNLOGI SEPULUH NOPEMBER

USULAN TUGAS AKHIR

1. IDENTITAS PENGUSUL

NAMA	**:**	**Setyassida Novian Putra Damara**
NRP	**:**	**05111440000024**
DOSEN WALI	**:**	**Adhatus Solichah Ahmadiyah S.Kom, M.Sc**
DOSEN PEMBIMBING	**:**	**1. Bagus Jati Santoso, S.Kom., Ph.D**
		2. Royyana Muslim Ijtihadie S.Kom, M.Kom., Ph.D

2. JUDUL TUGAS AKHIR

"Desain dan Implementasi Aplikasi Pengolahan *Top-k Dominating Queries* Pada Data *Streaming* Terdistribusi"

3. LATAR BELAKANG

Selama beberapa tahun terakhir ini kueri berbasis preferensi yang hanya mengambil *preferable* objek dari *multidimensional* dataset mendapat perhatian lebih dari para peneliti yang mempunyai fokusan pada pengolahan basis data. Jenis kueri ini dapat memberikan hasil dengan berbagai kriteria sehingga dapat dimanfaatkan dalam banyak aplikasi seperti, *multimedia retrieval* [1], *web search* [2], *analisis pasar* [3], dan e-*commerce* [3]. Dua metode yang paling banyak digunakan dalam kueri berbasis preferensi adalah *top-k* dan *skyline queries*.

Misalnya, diberikan fungsi ranking (penilaian) $f : \mathbb{R}^d \rightarrow \mathbb{R}$, di mana d adalah jumlah atribut (dimensi), kueri dengan metode *top-k* mengambil objek data k dengan nilai terbaik. Keuntungan dari metode *top-k queries* ini adalah pengguna dapat mengontrol jumlah objek data hasil dengan cara menetapkan parameter k, sedangkan kelemahan dari metode ini dirasa sulit bagi pengguna untuk menentukan peringkat yang sesuai fungsi.

Metode *skyline queries* mengatasi kelemahan pada metode *top-k queries* karena pada metode *skyline queries* tidak memerlukan fungsi peringkat apapun. Hasil kueri dari metode *skyline* terdiri dari objek data yang tidak didominasi oleh objek data lain dalam dataset tertentu. Hubungan dominasi didefinisikan sebagai berikut: misalnya diberi dua buah data o_i dan o_j, o_i mendominasi o_j jika o_i tidak lebih buruk daripada semua atribut o_j dan lebih baik dari pada o_j setidaknya pada satu atribut. Dengan metode *skyline queries*, pengguna bisa mendapatkan objek data yang tidak lebih buruk dari data yang lain. Namun, pengguna tidak bisa mengendalikannya ukuran dari hasil, yang mungkin dapat mengembalikan hasil data yang sangat banyak.

Metode *top-k dominating queries* adalah metode kueri yang menggabungkan kelebihan dari metode *top-k* dan *skyline queries*. Lebih spesifik lagi, metode *top-k dominating queries* tidak memerlukan fungsi peringkat yang didefinisikan oleh pengguna dan dapat mengontrol ukuran hasil. Metode *top-k queries* yang mendominasi mengambil objek data *k* yang mendominasi jumlah tertinggi objek data dalam dataset tertentu. Artinya, nilai data objek *o* adalah jumlah objek data yang didominasi oleh *o*. Kueri teratas mendominasi identifikasi yang paling penting objek data secara intuitif. Hal ini dapat membantu banyak aplikasi seperti contoh di atas.

Pada akhrinya kueri berbasis preferensi berperan penting dalam berbagai macam aplikasi dengan melibatkan basis data terdistribusi, seperti aplikasi pemantauan jaringan, aplikasi berbasis website, dan aplikasi untuk analisa pasar. Dalam aplikasi-aplikasi tersebut akan sering menghasilkan data dengan skala yang besar [4] yang menimbulkan tantangan untuk memberikan solusi atas permasalahan dari pengolahan *top-k dominating queries* pada data *streaming* terdistribusi.

4. RUMUSAN MASALAH

Rumusan masalah yang diangkat dalam tugas akhir ini dapat dipaparkan sebagai berikut:

1. Bagaimana cara mengolah *top-k dominating queries* pada data *streaming* terdistribusi?
2. Bagaimana merumuskan stuktur data yang terbaik untuk memecahkan permasalahan *top-k dominating queries* pada data *streaming* terdistribusi?
3. Bagaimana cara mengurangi biaya komputasi dan penyimpanan pengolahan t*op-k dominating queries* pada data *streaming* terdistribusi?

5. BATASAN MASALAH

Permasalahan yang dibahas dalam tugas akhir ini memiliki batasan antara lain:

1. Nilai atribut yang akan diproses dalam algoritma ini bertipe numerik.
2. *Tool* dan bahasa pemrograman yang digunakan adalah Matlab.
3. *Dataset* yang digunakan berupa data *real-life* dan sintetis.
4. Konsep data *streaming* yang digunakan adalah *sliding windows* dengan tipe *count-based*.

6. TUJUAN PEMBUATAN TUGAS AKHIR

Tujuan pembuatan tugas akhir ini antara lain:

1. Menemukan algoritma untuk mengolah *top-k dominating queries* pada data *streaming* terdistribusi.
2. Merumuskan struktur data yang terbaik untuk mendukung pemecahan permasalahan *top-k dominating queries* pada data *streaming* terdistribusi.
3. Merumuskan metode untuk mengurangi biaya komputasi dan penyimpanan *top-k dominating queries* pada data *streaming* terdistribusi.

7. MANFAAT TUGAS AKHIR

Manfaat dari pembuatan tugas akhir ini adalah:

1. Meneliti permasalahan dalam pengolahan *top-k dominating queries* pada data *streaming* terdistribusi.
2. Mengusulkan dua algoritma/metode yang efisien untuk pengolahan *top-k dominating queries* pada data *streaming* terdistribusi, dengan harapan kedua algoritma ini mengurangi biaya komputasi secara signifikan.
3. Mengusulkan algoritma perkiraan dalam pengolahan *top-k dominating data computation*, dengan harapan dengan algoritma perkiraan ini dapat mengurangi biaya komputasi jika dibandingkan dengan *exact algorithm*.

8. TINJAUAN PUSTAKA

a. **Top-K Queries**

Untuk saat ini ada dua metode/teknik kueri berbasis referensi yang sangat sering digunakan, diantaranya adalah: (i) metode *top-k queries* dan (ii) metode *skyline queries*. Metode *top-k queries* menggunakan fungsi peringkat yang didefinisikan sebagai berikut $f : \mathbb{R}^d \rightarrow \mathbb{R}$, yang dimana akan memberikan nilai untuk setiap *tuple/points p* (*tuple* adalah sebuah objek yang merupakan kumpulan elemen hasil kueri). Hasil dari *top-k queries* terdiri dari *k points* yang memiliki nilai tertinggi sesuai dengan fungsi $f()$. Keunggulan dari metode/teknik ini adalah jumlah jawaban/keluaran yang dapat dikontrol atau disesuaikan jumlahnya dengan meruabah parameter *k*, walaupun untuk beberapa kasus, kardinalitas dari jawaban/keluaran yang dihasilkan sama atau melebihi nilai parameter *k* (contohnya, terdapat duat atau lebih *points* yang memiliki nilai yang sama, dengan begitu keluaran yang dihasilkan dapat melebihi nilai *k* yang sudah ditetapkan.) Dalam beberapa kasus, semua nilai yang sama akan disertakan menjadi hasil atau depat juga dipangkas dengan kriteria tertentu sehingga jawaban/keluaran yang dihasilkan dapat sebanyak *k*. Salah satu keterbatasan metode ini adalah sangat bergantung pada fungsi peringkat, fungsi ini biasanya diatur oleh pengguna, yang dimana fungsi yang berbeda akan menghasikan jawaban/keluaran yang berbeda pula. Terlebih-lebih dalam beberapa kasus pemilihan fungsi yang sesuai tidak berdarkan intuitif. Contohnya dalam aplikasi *e-commerce*, tidak ada cara yang

mudah dalam mengombinasikan atribut kecepatan *CPU* dan manajemen baterai untuk memilih laptop yang paling banyak diminati. [5]

b. **Skyline Queries**

Disisi lain pada metode *skyline queries*, tidak membutuhkan fungsi peringkat seperti yang ada pada metode *top-k queries* dan memiliki karkteristik *scaling invariance*, yang berarti jika skala data dirubah pada nilai dimesi *d* nya, maka hasil pengolahan tidak akan berubah. Hasil dari metode *skyline queries* ini tersusun dari *points* yang tidak didominasi oleh *points* yang lain. Hubungan dominasi ini bergantuk pada semantik setiap atribut data yang ada. Dalam beberapa kasus, nilai yang lebih rendah akan lebih baik (contoh: harga) sedangkan dalam beberap kejadian nilai yang tinggi juga diinginkan (contoh: kualitas suatu barang). Tanpa menghilangkan aturan umum, penulis fokus dalam meminimalisir nilai dimensi (nilai yang lebih kecil akan lebih baik). Dengan begitu sebuah *points* p dikatakan mendominasi points yang lain q $(p \prec q)$, jika dan hanya jika p tidak lebih buruk dari q pada seluruh atribut dan lebih baik dibandingkan q minimal pada 1 dimensi. [5]

c. **Top-K Dominating Queries**

Top-k dominating queries merupakan gabungan antara dua kueri berbasis preferensi **[5]** yang paling banyak digunakan yaitu: 1) *Top-k query* dan 2) *skyline query*. *Top-k dominating queries* mengunakan fungsi ranking untuk merangking object (metode *Top-k queries*) dan mengunakan *dominating relationship* (metode *Skyline queries*). Tujuan dari metode *Top-k dominating queries* adalah mempertahankan keuntungan dan mengeliminasi keterbatasan dari metode *top-k queries* dan *skyline queries*. Maka dari itu *Top-k dominating queries* memiliki beberapa sifat khusus antara lain:

1. jumlah hasilnya terkontrol,
2. hasil tidak berubah,
3. tidak perlunya definisi penilaian dari user,
4. pada setiap objek akan memiki nilai yang akan menentukan rankingnya.

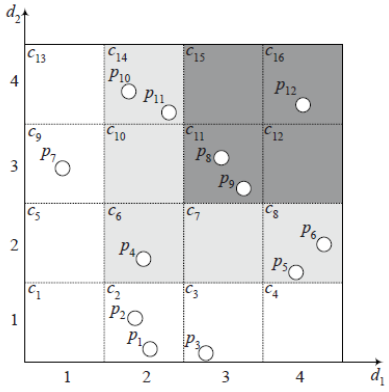
Top-k dominating queries dapat didefinisikan sebagai sebuah kueri yang akan mengembalikan sebanyak *k* objek yang memiliki nilai dominasi tertinggi pada suatu dataset yang diberikan. Penentuan nilai skor pada object dilakukan dengan cara menghitung jumlah objek lain yang didominasi oleh objek tersebut. Dalam hal ini yang dimaksut dengan dominasi adalah sebagai berikut. Diberikan dua buah objek yaitu *o* dan *o'*. *o* dapat dikatan mendominasi *o'* yang disimbolkan dengan $o \prec o'$ jika dua kondisi ini terpenuhi, kondisi yang pertama pada setiap dimensi *i* pada objek *o* tidak ada yang lebih besar dari dimensi *i* pada objek *o'*, kondisi yang kedua terdapat minimal satu dimensi pada object *o* yang lebih tinggi dari pada objek *o'*.

d. **Sliding Windows**

Sliding Windows adalah sebuah teknik dalam dunia pemrograman yang memodelkan data seperti sudut pandang kaca di dalam sebuah bis [6]. Terdapat *n*

objek yang akan diamati dan *k* yang merupakan panjang sebuah kaca mengamatan atau dapat disebut juga panjang objek yang sedang diamati. Biasanya objek yang diamati disebut dengan objek aktif. Terdapat dua tipe *sliding window* [5] yang pertama adalah *count-based sliding window* dimana nilai *k* selalu konstan. Jika ada sebanyak *r* objek masuk menjadi objek aktif maka akan ada juga sebanyak *r* objek aktif yang akan kadaluarsa. Tipe yang kedua adalah *time-based sliding widow* dimana jumlah *k* tidak konstan. Pada tipe ini setiap objek memiliki waktu aktif masing-masing yang tidak saling terkaitan. Sehingga jumlah objek yang aktif bisa berbeda pada waktu yang berbeda.

e. **Grid Based Indexing**



Gambar 1. Sel yang didominasi secara menyeluruh dan sebagian. **[5]**

Skema *grid-based indexing* digunakan dengan tujuan memberikan indeks dan menjaga kemudahan dalam proses penyajian algoritma yang digunakan. Hal ini telah mendapatkan penelitian sebelumnya bahwa implementasi struktur indeks sederhana ini mempengaruhi performa pada lingkungan yang sangat dinamis [7], [8], [9]. Pada setiap sel *grid* mengandung identitas *ID* dari setiap *point* yang berada pada tiap sel. Sel c_i mendominasi secara penuh seluruh sel yang terletak pada daerah atas-kanan hingga ujung pojok kanan atas c_i. Sebagai contoh, sel c_6 pada Gambar 1 mendominasi sel-sel c_{11}, c_{12}, c_{15}, dan c_{16}. Sel-sel yang didominasi c_6 ini disebut sebagai sel yang didominasi secara keseluruhan (*fully dominated cells* atau *simply dominated cells*).

Disisi lain sel-sel c_6, c_7, c_8, c_{10} dan c_{14} ada kemungkinan mengandung *point* yang didominasi oleh *point* yang berada pada sel c6. Sel-sel ini disebut dengan sel yang didominasi secara sebagian (*partially domimated cells*). Pada Gambar 1, sel-sel yang didominasi oleh c_6 diarsir dengan warna abu-abu gelap sedangkan sel-sel yang didominasi secara sebagian oleh c_6 ditandai dengan arsiran warna abu-abu terang.

Sebagai tambahan untuk meningkatkan efisiensi pemberian indeks, *grid* yang ada juga digunakan sebagai menghitung nilai dominasi *score*(p_i) dari sebuah point p_i. Pertama-tama, tentukan sel c_i yang mengandung p_i. Ingat, bahwa operasi

ini berjalan sangat cepat dan memerlukan waktu yang konstan. Untuk menghitung nilai *score(p_i)* membutuhkan perhitungan jumlah *point* yang didominasi oleh p_i point yang terdominasi oleh p_i berada pada sel-sel yang terdominasi secara keseluruhan atau sebagian oleh sel tamu p_i. sebagain contoh, pada Gambar 1, p_4 mendominasi p_6 dan p_{11} secara sebagian dan ada tiga *point* (p_8, p_9, dan, p_{12}) yang terdominasi secara keseluruhan. Maka dari itu, nilai/skor total dari p_4 adalah *score(p_4)* = 2 | 3 = 5. Proses ini dinamakan *exact score computation* dan merupakan proses yang paling banyak memakan waktu.

9. RINGKASAN ISI TUGAS AKHIR

Tugas Akhir ini disusun untuk menangani masalah besarnya biaya komputasi dan biaya penyimpanan dalam pengolahan *top-k dominating queries* pada *data streaming* terdistribusi. Pada data ini saya penulis mengadopsi metode *count-based sliding window* [1], dimana dihasilkan objek data dalam *W* waktu dari waktu sekarang yang akan menjadi data pantauan.

Dalam proposal Tugas Akhir ini penulis mengusulkan dua metode pendekatan untuk *exact top-k dominating data monitoring*. Pertama adalah pendekatan berdasarkan penyaringan dimana dalam metode ini menggunakan data yang meiliki nilai dominasi tinggi sebagai penyaring untuk menhindari pengiriman data yang tidak perlu. Kedua dengan metode *cache-based* dengan harapan dapat mengurangi biaya komputasi secara signifikan. Dengan memanfaatkan kedua metode ini penulis mengusulkan sebuah algoritma untuk pengolahan *top-k dominating queries* pada data *streaming* terdistribusi. Untuk mengurangi biaya komunikasi dan komputasi, sebisa mungkin kita harus menghindari perhitungan yang berulang dari data *top-k dominating*.

Pada penelitian ini, penulis mengusulkan metode *lower-* and *upper-bounding* untuk memberikan nilai untuk tiap objek data yang ada, dengan menggunakian metode ini penulis akan secara cermat untuk memilih kandidat data yang diberikan dari total keseluruhan data. Penulis juga mengusulkan metode *sampling-based approximate* untuk perhitungan data *top-k dominating* dengan harapan dapat mengurangi biaya komunikasi dan komputasi secara signifikan dengan menjaga tingkat akurasi yang tinggi.

10. METODOLOGI

a. Penyusunan proposal tugas akhir

Proposal tugas akhir ini berisi tentang penjelasan mengenai pendahuluan dari tugas akhir yang akan dibuat. Pendahuluan ini terdiri dari hal yang melatarbelakangi tugas akhir, rumusan masalah yang diangkat, batasan masalah yang ada, tujuan dan manfaat dari tugas akhir ini. Selain itu, dijabarkan pula tinjauan pustaka yang digunakan sebagai referensi pendukung dalam pembuatan tugas akhir.

b. Studi literatur

Pada studi literatur ini, akan dipelajari beberapa referensi yang akan diperlukan untuk membantu mendesain algoritma untuk mengolah *top-k dominating queries* pada

data berbasis kelompok. Secara garis besar, ada tiga metode/teknik/algoritma yang akan menjadi pilar dalam tugas akhir kali ini. Yaitu, metode *top-k queries*, *skyline queries*, dan yang terakhir adalah gebungan dari kedua metode sebelumnnya dengan nama *Top-k Dominating Queries.*

c. Analisis dan desain perangkat lunak

Pada tahap ini, penulis akan menganalisa masalah yang ada dalam pengolahan *top-k dominating query* pada data *streaming terdistribusi* dan mendesain algoritma yang dapat mengatasi permasalahan tersebut.

d. Implementasi perangkat lunak

Perangkat keras yang digunakan adalah perangkat keras yang berbasis Windows demi kemudahan instalasi berbagai macam perangkat lunak yang dibutuhkan. Pada tugas akhir kali ini direncanakan penggunaan perangkat keras dan sistem operasi mempunyai spesifikasi sebagai berikut:

- Intel Core i3-2330M 2.2GHz
- NVIDIA GeForce GT 920m 1GB
- 4GB of RAM
- Windows 10 64-bit

jika kemudian ternyata digunakan perangkat keras yang mempunyai spesifikasi yang berbeda, rincian spesifikasinya akan dilaporkan kemudian pada saat penyusunan laporan akhir pada tugas akhir kali ini.

Sedangkan untuk *environment* perangkat lunak, akan digunakan bahasa pemrograman Matlab dengan berbagai macam *library* pendukung.

e. Pengujian dan evaluasi

Pengujian dilakukan untuk mengetahui apakah system yang dibangun dengan metode yang diusulkan telah bekerja dengan baik dan efisien atau belum. Pengujian dalam algoritma ini akan dilakukan dalam beberapa cara, antara lain:

1. Pengujian Akurasi
   Pengujian ini akan berfokus pada ketepatan solusi yang diberikan dalam menemukan algoritma untuk mengolah *top-k dominating queries* pada data *streaming* terdistribusi.
2. Pengujian Waktu Eksekusi
   Pengujian ini akan berfokus pada seberapa lama waktu yang dibutuhkan untuk mengeksekusi algoritma dalam pengolahan *top-k dominating queries* pada data *streaming* terdistribusi.
3. Pengujian Penggunaan Memori
   Pengujian ini akan berfokus pada pengukuran besarnya memori yang digunakan saat aplikasi dijalankan.

f. Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi

perangkat lunak yang telah dibuat. Sistematika penulisan buku tugas akhir secara garis besar antara lain:

1. Pendahuluan
   a. Latar Belakang
   b. Rumusan Masalah
   c. Batasan Tugas Akhir
   d. Tujuan
   e. Metodologi
   f. Sistematika Penulisan
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Pengujian dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka

11. JADWAL KEGIATAN

Tahapan	2017				2018																			
	Desember				Januari				Februari				Maret				April				Mei			
Penyusunan Proposal			■	■	■																			
Studi Literatur				■	■	■	■																	
Perancangan Sistem					■	■	■	■	■															
Implementasi							■	■	■	■	■	■	■	■	■	■	■	■	■	■				
Pengujian dan Evaluasi											■	■	■	■	■	■	■	■	■	■	■	■		
Penyusunan Buku															■	■	■	■	■	■	■	■	■	■

12. DAFTAR PUSTAKA

[1] E. Tiakas, G. Valkans, A. N. Papadopoulos, Y. Manolopoulos and D. Gunopoulos, "Metric-based top-k dominating queries," *EDBT*, pp. 415-426, 2014.

[2] D. Skoutas, D. Sacharidis, A. Simitsis, V. Kantere and T. Sellis, "Top-k dominant web services under multi-criteria matching," *EDBT*, pp. 898-909, 2009.

[3] A. Yu, P. Agarwal and J. Yang, "Processing a large number of continuous preference top-k queries," *SIGMOD*, pp. 397-408, 2012.

[4] D. Amagata, T. Hara and N. Shojiro, "Sliding window top-k dominating query processing over distributed data streams," *Springer Science*, pp. 535-566, 2015.

[5] M. Kontaki, A. N. Papadopoulos and Y. Manolopoulos, "Continuous Top-k Dominating Queries," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 5, pp. 840 - 853, 2012.

[6] "GeeksforGeeks," GeeksforGeeks, [Online]. Available: https://www.geeksforgeeks.org/window-sliding-technique/. [Accessed 3 January 2018].

[7] K. Mouratidis, S. Bakiras and D. Papadias, "Continuous Monitoring of Top-k Queries over Sliding Windows," *SIGMOD*, pp. 635-646, 2006.

[8] L. U, K. Mouratidis and N. Mamaoulis, "Continuous Spatial Assignment of Moving Users," *The VLDB Journal*, 2009.

[9] X. Xiong, M. Mokbel and W. Aref, "SEA-CNN: Scalable Processing of Continuous K-Nearest Neighbor Queries in Spatiotemporal Databases," *ICDE*, pp. 643-654, 2005.